LiftWEC

DEVELOPMENT OF A NEW CLASS OF WAVE ENERGY CONVERTER BASED ON HYDRODYNAMIC LIFT FORCES

Deliverable D5.2
Validated numerical model with real-time computational capabilities

Deliverable Lead	Maynooth University
Delivery Date	9th November 2021
Dissemination Level	Public
Status	Released
Version	1.0

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

Document Information

Project Acronym	LiftWEC
Project Title	Development of a new class of wave energy converter based on hydrodynamic lift forces
Grant Agreement Number	851885
Work Package	WP05
Related Task(s)	T5.2
Deliverable Number	D5.2
Deliverable Name	Validated numerical model with real-time computational capabilities
Due Date	30th October 2021
Date Delivered	9th November 2021
Primary Author(s)	Andrei Ermakov (MU)
Co-Author(s)	John V. Ringwood (MU), Gerrit Olbert (TUHH), Abel Arredondo-Galeana (US), Remy Pascal (INNOSEA), Louis Papillon (INNOSEA), Florent Thiebaut (ECN), Grégory Payne (ECN)
Document Number	LW-D05-02

Version Control

Revision	Date	Description	Prepared By	Checked By
1.0	09/11/2021	Release for use	AE	JR

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

EXECUTIVE SUMMARY

This deliverable is associated with Task 5.2 and is in the form of a software tool that will allow the investigation of different control strategies for the LiftWEC concept and as a basis for the model based control design. Originally it was supposed to be a software tool accompanied by a report that describes the model development process and validation. However, in this deliverable we also present a new analytical model, which is derived for a horizontal cyclorotor with two hydrofoils in the polar coordinate system. It is relatively simple, fast, and suitable for the analytical and numerical study of cyclorotor-based WECs, as well as control system development. This was achieved by using new analytical formulae, which were derived by the authors and validated with numerical and experimental results published in previous research [1-4] as well as experimental results which were obtained by LiftWEC WP04 – physical modelling [5-7]. These new formulae significantly decrease the calculation time, compared to previous models [2,4], and increase the accuracy of the results, as well as providing useful insight into the nature of the system behaviour. The model has been published in the Journal of Ocean Engineering and Marine Energy [8] and presented at the 14th European Wave and Tidal Energy Conference [9]. The published model has attracted the attention of the Atargis Energy Corporation [1], who reproduced the published results, confirmed the benefits of the new model, and provided us with useful feedback. Based on their feedback we have submitted an erratum with minor corrections of the model article text [10]. Recently, the members of LiftWEC WP03 – Numerical Modelling from Hamburg University of Technology have started the development of the software “HydraLift” which is also based on the developed mathematical model for cyclorotor-based WEC [8-10]. This software has been licensed by the End-User License Agreement ("EULA"). However, in the Appendix of this deliverable you can find an open-source Python code which represents the case described in [9].

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

TABLE OF CONTENTS

EXECUTIVE SUMMARY 3
TABLE OF CONTENTS 4
1 A CONTROL-ORIENTATED MODEL FOR LIFTWEC ..... ERROR! BOOKMARK NOT DEFINED.
1.1 The Mechanical Model ..... **Error! Bookmark not defined.**
1.2 The Hydrodynamical Model 6
1.3 Approximate Determination of Lift and Drag Forces 10
2 VALIDATION OF THE DEVELOPED MODEL 11
2.1 Validation of the Developed Model against previous research in Terms of Free Surface Perturbation 11
2.2 Validation of the Developed Model against the LiftWEC 2D experimental results in Terms of Tangential and Radial Forces 13
2.3 Conclusion 16
3 BIBLIOGRAPHY 16
4 APPENDIX 18

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

1 A CONTROL-ORIENTATED MODEL FOR LIFTWEC

1.1 THE MECHANICAL MODEL

We present a new analytical model, which can provide the basis for the control design of a cyclorotor-based wave energy converter. Our model is derived for a horizontal cyclorotor with two hydrofoils. The mechanical model is based on Newton's second law for rotation. Rotation is considered in two-dimensional potential flow, for both monochromatic and panchromatic waves, including waves generated by the rotating rotor, and viscous losses. The developed model is very convenient for modelling, analysis and control design for a cyclorotor based WEC. We consider wave propagation in the Cartesian coordinate system, and rotor rotation in polar coordinates, as shown in Fig. 1.1. The rotational centre of the cyclorotor is located at $x_0 = 0$ and submerged by y_0.

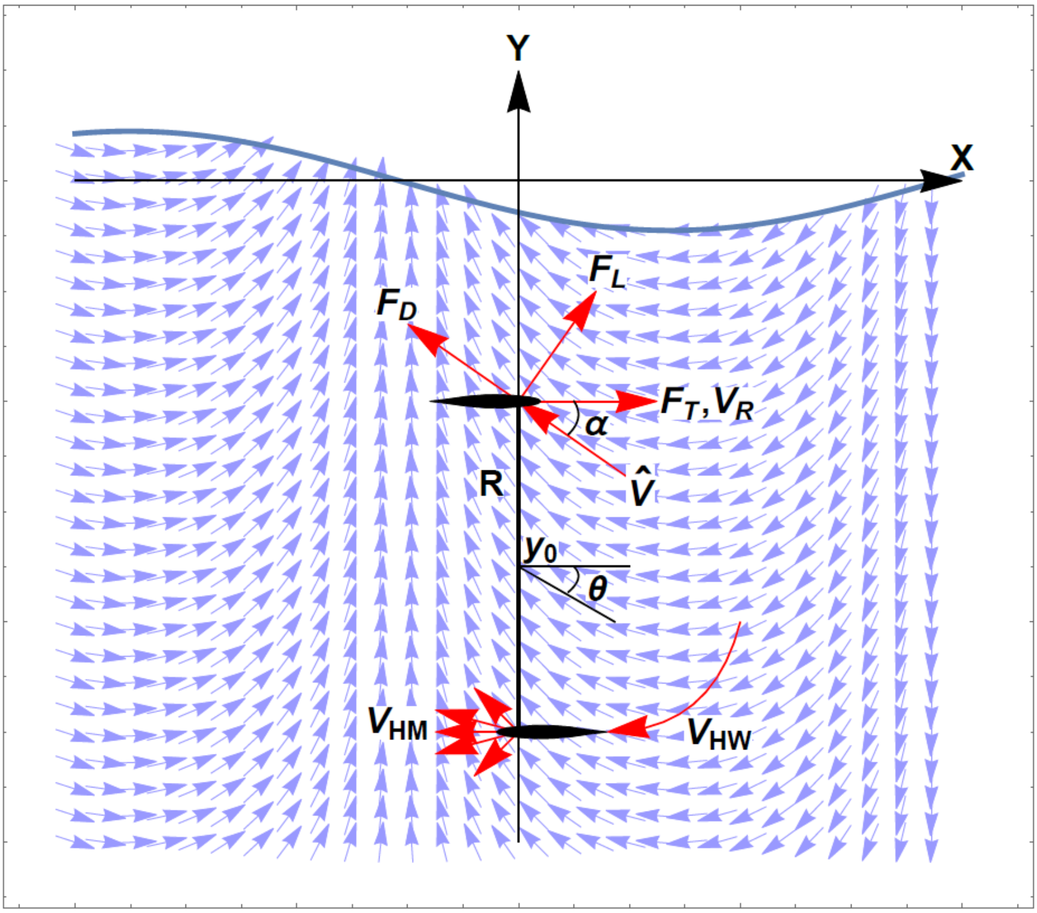

Figure 1.1: The detailed diagram of the cyclorotor with two hydrofoils: θ - polar coordinates of the hydrofoils, $\boldsymbol{V_W}$ *- wave induced fluid velocity,* $\boldsymbol{V_R}$ *- speed of the foils,* $\boldsymbol{V_{HM}}$ *- instantaneous radiation from the moving foil and* $\boldsymbol{V_{HW}}$ *- the wake which is left behind,* $\widehat{V}$ *- the overall relative to hydrofoil fluid velocity, α - the attack angle,* F_L*,* F_D*,* F_T *- lift, drag and tangential forces.*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

The position of the hydrofoils can be determined as:

$$x_i(t) = -R\cos(\theta(t) + \pi i) \tag{1}$$

$$y_i(t) = y_0 + R\sin(\theta(t) + \pi i) \tag{2}$$

where i=1,2 is the hydrofoil number, (x_i, y_i) is the position of hydrofoil i, R is the operational radius, and θ is the polar angle coordinate in the selected time moment t.

Taking the time derivatives of the positions (1,2), we can obtain the velocity vectors of the foils $\mathbf{V_R}$={$(V_R)_x$, $(V_R)_y$} as:

$$\left(V_{R_i}\right)_x = R\,\dot{\theta}(t)\sin(\theta(t) + \pi i) \tag{3}$$

$$\left(V_{R_i}\right)_y = R\,\dot{\theta}(t)\cos(\theta(t) + \pi i) \tag{4}$$

where $\dot{\theta}(t)$ is the angular velocity.

The mechanical model of the rotor is based on Newton's second law for rotation and balances the product of the rotor's acceleration $\ddot{\theta}(t)$ and inertia I with the torques caused by the tangential forces F_{T_i} generated on the hydrofoils. It is assumed that the model is directly connected with a rotational generator, which exerts an opposing torque - $\mathcal{T}$:

$$I\ddot{\theta}(t) = \left(F_{T_1} + F_{T_2}\right)R - \mathcal{T} \tag{5}$$

The tangential forces F_{T_i} generated on the hydrofoil can be manipulated by pitching the hydrofoils which, in turn, changes the attack angle α. The PTO torque $\mathcal{T}$ is used both to take rotational energy from the system, to generate electrical energy, or supply energy to increase rotational speed. In the second case, we presume the PTO generator system has the ability to switch to a motoring mode.

As an example of performance function, we use captured energy in a traditional form used in wind, tidal and wave energy metrics. It is defined as maximisation of the time integral of the product between angular velocity $\dot{\theta}(t)$ and PTO torque on the time interval [0,T].

$$Max \quad J = \int_0^T T(t)\dot{\theta}(t)dt \tag{6}$$

1.2 The Hydrodynamical Model

We consider the rotation in two-dimensional potential flow which includes incoming monochromatic or panchromatic waves, as well as radiated waves generated by the rotating rotor, and viscous losses.

As an example of incoming waves, we present Airy waves which were used in the study of Siegel et al [4], which can be described by the following velocity potential:

$$\Phi_W(x, y, t) = \frac{Hg}{2\omega}e^{ky}\sin(kx - \omega t) \tag{7}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

where H is the wave height, ω is the wave frequency, k is the wave number, and g is the acceleration due to gravity.

The components of the wave induced velocity **V_W** can be found as a gradient from the potential:

$$\boldsymbol{V_W} = \nabla\Phi_W(x,y,t) = \{(V_W)_x, (V_W)_y\} \tag{8}$$

From these partial derivatives, we get the components of the wave induced velocity:

$$(V_W)_x = \frac{e^{ky} gHk}{2\omega} \cos(kx - t\omega) \tag{9}$$

$$(V_W)_y = \frac{e^{ky} gHk}{2\omega} \sin(kx - t\omega) \tag{10}$$

One of the challenges in the development of the cyclorotor-based WEC is the estimation of the waves radiated by rotating hydrofoils. In previous works [2,11,12], the hydrofoil, from the far-field, was modelled as a single moving point vortex in infinitely deep water. This vortex can be represented by a complex potential which satisfies the kinematic and dynamic boundary condition on the free surface [13]:

$$\mathcal{F}(z,t) = \frac{\Gamma(t)}{2\pi\,\mathbb{i}} Log\left[\frac{z - c(t)}{z - \tilde{c}(t)}\right] + \frac{g}{\pi\,\mathbb{i}} \int_0^t \int_0^\infty \frac{\Gamma(\tau)}{\sqrt{gk}} e^{-\,\mathbb{i}\,k(z - \tilde{c}(\tau))} \sin\left(\sqrt{gk}(t - \tau)\right) dk\, d\tau \tag{11}$$

where $\mathrm{c(t)} = \mathrm{x(t)} + \mathbb{i}\,\mathrm{y(t)}$ is the position of the hydrofoil, $\tilde{c}(t)$ is the complex conjugate of c(t), k is the wave number and Γ - is the circulation of the vortex, or the line integral of the fluid velocity along a closed path.

The potential $\mathcal{F}(z,t)$ in (11) consists of two parts. The first term on the right-hand side of (11) the instantaneous (momentary) radiation and has a singularity at the source point c(t). For this reason, it cannot be used to describe the state in close vicinity to the foil. The second term on the right-hand side of (11) describes the fluid velocity wake caused by the moving vortex. In the study by Hermans et al. [2], this term is calculated numerically, using double integration over the wave number k and the time parameter τ. A very similar approach is employed in the work of Siegel et al. [3,4,12], where it was integrated using second order k and τ marching techniques.

The authors of this work have solved the integral over k analytically in the form of the Dawson [13] function D[x]:

$$\mathcal{F}(z,t) = \frac{\Gamma(t)}{2\pi\,\mathbb{i}} Log\left[\frac{z - c(t)}{z - \tilde{c}(t)}\right] - \frac{2\mathbb{i}\sqrt{g}}{\pi} \int_0^t \frac{\Gamma(\tau)}{\sqrt{\mathbb{i}(z - \tilde{c}(\tau))}} D\left[\frac{\sqrt{g}(t - \tau)}{2\sqrt{\mathbb{i}(z - \tilde{c}(\tau))}}\right] d\tau \tag{12}$$

where

$$D(x) = e^{-x^2} \int_0^x e^{y^2} dy \tag{13}$$

This representation is valid only when $Im[z - \tilde{c}(\tau)] < 0$ or $y + y_i < 0$, which is always true for the area under consideration since y < 0 and y_i < 0. This new formula significantly decreases the calculation time and increases the accuracy of the results, since all the wave numbers k are now covered, and we only need to find one integral with defined limits.

The velocity of the waves radiated by a rotating hydrofoil can be found using the following equation:

$$\mathbf{V_H} = \frac{\partial \mathcal{F}(z,t)}{\partial z} = (V_H)_x - \mathbb{i}\,(V_H)_y \tag{14}$$

The velocity field $\mathbf{V_H}$ of the waves radiated by the hydrofoil also consists of the instantaneous (momentarily) radiated waves $\mathbf{V_{HM}}$ and wakes $\mathbf{V_{HW}}$ which were left in the hydrofoil's path:

$$\boldsymbol{V_H} = \boldsymbol{V_{HM}} + \boldsymbol{V_{HW}} \tag{15}$$

The components of the velocity field caused by the hydrofoil **i** at the point **j** are:

$$(V_{HM})_x = \frac{\Gamma_i(t)y_i\left(\left(x_j - x_i\right)^2 - \left(y_j{}^2 - y_i^2\right)\right)}{\pi\left(\left(\left(x_j - x_i\right)^2 - \left(y_j{}^2 - y_i^2\right)\right)^2 + 4\left(y_j\left(x_j - x_i\right)\right)^2\right)} \tag{16}$$

$$(V_{HM})_y = \frac{2\Gamma_i(t)y_jy_i\left(x_j - x_i\right)}{\pi\left(\left(\left(x_j - x_i\right)^2 - \left(y_j{}^2 - y_i^2\right)\right)^2 + 4\left(y_j\left(x_j - x_i\right)\right)^2\right)} \tag{17}$$

$$(V_{HW})_x \;= Re\Bigg[\int_0^t \frac{\Gamma_i[\tau]g(t-\tau)}{2\pi\left(x_j - x_i[\tau] + \mathbb{i}\left(y_j + y_i[\tau]\right)\right)^2}(1 +$$

$$\frac{2\;\mathbb{i}\;\left(\left(x_j - x_i[\tau\,]\right) + \mathbb{i}\left(y_j + y_i[\tau]\right)\right) - g(t-\tau)^2}{\sqrt{g}(t-\tau)\sqrt{\mathbb{i}\;\left(x_j - x_i[\tau] + \mathbb{i}\left(y_j + y_i[\tau]\right)\right)}} D\left[\frac{\sqrt{g}(t-\tau)}{2\sqrt{\mathbb{i}\;\left(x_j - x_i[\tau] + \mathbb{i}\left(y_j + y_i[\tau]\right)\right)}}\right]\Bigg)\,d\tau\Bigg] \tag{18}$$

$$(V_{HW})_y \;= -Im\Bigg[\int_0^t \frac{\Gamma_i[\tau]g(t-\tau)}{2\pi\left(x_j - x_i[\tau] + \mathbb{i}\left(y_j + y_i[\tau]\right)\right)^2}(1 +$$

$$\frac{2\;\mathbb{i}\;\left(\left(x_j - x_i[\tau\,]\right) + \mathbb{i}\left(y_j + y_i[\tau]\right)\right) - g(t-\tau)^2}{\sqrt{g}(t-\tau)\sqrt{\mathbb{i}\;\left(x_j - x_i[\tau] + \mathbb{i}\left(y_j + y_i[\tau]\right)\right)}} D\left[\frac{\sqrt{g}(t-\tau)}{2\sqrt{\mathbb{i}\;\left(x_j - x_i[\tau] + \mathbb{i}\left(y_j + y_i[\tau]\right)\right)}}\right]\Bigg)\,d\tau\Bigg] \tag{19}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

The complex potential $\mathcal{F}(z,t)$ can also be presented in the form of the sum of the velocity potential Φ_H and stream function Ψ_H as:

$$\mathcal{F}(z,t) = \Phi_H(x,y) + \mathbb{i}\,\Psi_H(x,y) \tag{20}$$

Thus, the new velocity potential for waves radiated by the hydrofoil, which was derived by the authors from (11), using representation (20) and the Dawson function (13), has the following form:

$$\begin{gathered}\Phi_H(x,y) = \frac{\Gamma(t)}{2\pi}\arctan\left[\frac{2y_i(x_i - x)}{(x - x_i)^2 + \left(y^2 - y_i^2\right)}\right] + \\ \int_0^t \frac{\Gamma[\tau]\sqrt{g}}{\pi}\left(\frac{(-1)^{1/4} D\left[\frac{(-1)^{3/4}\sqrt{g}(t-\tau)}{2\sqrt{(x - x_i[\tau]) + \mathbb{i}(y + y_i[\tau])}}\right]}{\sqrt{(x - x_i[\tau]) + \mathbb{i}(y + y_i[\tau])}} + \right. \\ \left. \frac{(-1)^{3/4} D\left[\frac{(-1)^{1/4}\sqrt{g}(t-\tau)}{2\sqrt{(x - x_i[\tau]) - \mathbb{i}(y + y_i[\tau])}}\right]}{\sqrt{(x - x_i[\tau]) - \mathbb{i}(y + y_i[\tau])}}\right) d\tau\end{gathered} \tag{21}$$

and, despite the presence of the complex terms, the value of the function in (21) is always real. In addition, for cases where multiple (square) roots of a variable are taken, the following development ultimately utilises only the square of the root, making it immaterial which of the roots is taken.

In the case of the potential flow, the free surface perturbation can be found from the dynamic boundary condition. For example, the elevation of the free surface caused by the Airy wave has the following form:

$$\eta_w = -\frac{1}{g}\left(\frac{\partial \Phi_W}{\partial t}\right)_{y=0} = \frac{H}{2}\cos(kx - \omega t) \tag{22}$$

Now, we can obtain the perturbation of the free surface caused by the rotating hydrofoil i using equation (21):

$$\eta_{h_i} = -\frac{1}{g}\left(\frac{\partial \Phi_{H_i}}{\partial t}\right)_{y=0} \tag{23}$$

and the overall elevation of the free surface can be presented in the form of the linear sum:

$$\eta = \eta_w + \eta_{h_1} + \eta_{h_2} \tag{24}$$

1.3 Approximate determination of lift and drag forces

Modelling of the hydrofoil interaction with the wave velocity field is a challenging problem. Accurate determination of the attack angle, circulation, lift and drag forces requires significant high-fidelity computation, such as computational fluid dynamics (CFD). All these do not make the high-fidelity models suitable for control design. That is why, we consider the approximate model for lift and drag forces which can be used for the control design.

We use a basic representation of the hydrofoils as point sources. For this case, the lift and drag coefficients should be considered not as physical values, but more as tuning parameters. These best-fit approximate coefficients can be obtained from numerical simulations or experimental tests. These parameters depend on the following system inputs, which can be measured or tracked in real time:

- The rotational velocity $\mathbf{V_R}$ and position of the rotor θ can be measured and controlled
- The wave induced fluid velocity $\mathbf{V_W}$ can be reliably predicted in real time due to the minimal upstream radiation
- The velocity of waves radiated by the hydrofoils $\mathbf{V_H}$ can be calculated relatively easily; however, we cannot define the instantaneous radiated waves $(\mathbf{V_{HM}})_i$ in the small vicinity of the point source i, due to the singularity highlighted in Section 1.2.

Thus, we consider the generation of the lift and drag forces as the result of the rotation of the hydrofoil i with an overall relative velocity $\widehat{\boldsymbol{V_\iota}}$, representing the vector difference between the wave induced fluid velocity $\boldsymbol{V_{W_i}}$ and the cyclorotor rotational velocity V_{R_i}, plus the sum of the wakes caused by the hydrofoil rotation V_{HW} and instantaneous radiation from the other foils V_{HM} as:

$$\widehat{V_\iota} = V_{W_i} - V_{R_i} + V_{HW_i} + V_{HM_j} + V_{HW_j} \tag{25}$$

The attack angle $\alpha_i(t)$ can be found as the angle between the cyclorotor rotational velocity V_{R_i} and overall relative velocity $\widehat{V_\iota}$:

$$\alpha_i(t) = \arcsin\left(\frac{\left(V_{R_i}\right)_x\left(\widehat{V_\iota}\right)_y - \left(V_{R_i}\right)_y\left(\widehat{V_\iota}\right)_x}{\left|V_{R_i}\right|\left|\widehat{V_\iota}\right|}\right) + \gamma_i \tag{26}$$

where γ_i is the hydrofoil pitch angle, which can be adjusted in real time.

For the point source representation, we use the following approximation: $\mathbf{F_L}$ lift and $\mathbf{F_D}$ drag forces which act on a particular hydrofoil depend on the lift and drag coefficients $C_L(\alpha)$ and $C_D(\alpha)$, hydrofoil chord length S, fluid density ρ and overall relative velocity $\widehat{\boldsymbol{V}}$ at a hydrofoil position (x_i,y_i):

$$F_L = \frac{1}{2}\,\rho\, C_L(\alpha)\left|\widehat{V}\right|^2 S, \tag{27}$$

$$F_D = \frac{1}{2}\,\rho\, C_D(\alpha)\left|\widehat{V}\right|^2 S \tag{28}$$

The circulation Γ can be determined using the following equation:

$$\Gamma = \frac{1}{2}\, C_L(\alpha)\left|\widehat{V}\right| S \tag{29}$$

The tangential force F_T can now be presented as a combination of the lift F_L and drag F_D forces:

$$F_T = F_L(\alpha)\sin(\alpha-\gamma) - F_D(\alpha)\cos(\alpha-\gamma) \qquad (30)$$

2 VALIDATION OF THE DEVELOPED MODEL

2.1 VALIDATION OF THE DEVELOPED MODEL AGAINST THE PREVIOUS RESEARCH IN TERMS OF FREE SURFACE PERTURBATION

In this section we validate the results obtained from (21,23) for the heights and periods of the waves generated by a single rotating hydrofoil, against results obtained experimentally and numerically by previous researchers (21).

In the research work by Hermans et al. [2] the authors derive an analytical equation that can be used to compute the heights of waves generated by a foil which rotates at a constant rate. This equation is only valid for relatively large values of t, i.e. when stable periodic wave generation is achieved. The authors solved the system (11,23) numerically and the calculated results were compared with the experimental data. The experiments were conducted in the deep water basin of MARIN.

The prototype consisted of a single hydrofoil with chord length l = 0.1 m, operating radius R = 0.14 m, submerged depth of y_0 = -0.271 m with the wave profile measured at a point located at x = 1.8 m. The circulation was defined as $\Gamma = \pi|\omega R|\, l\, tan(\alpha)$. Fig. 2.1(b) shows the published results from [2] for the free surface elevation at the measurement point for a foil rotating with ω = 6.91 rad/s and α = 0.576 rad. It can be seen that good agreement between the amplitude and period of the radiated waves was obtained Fig. 2.1.

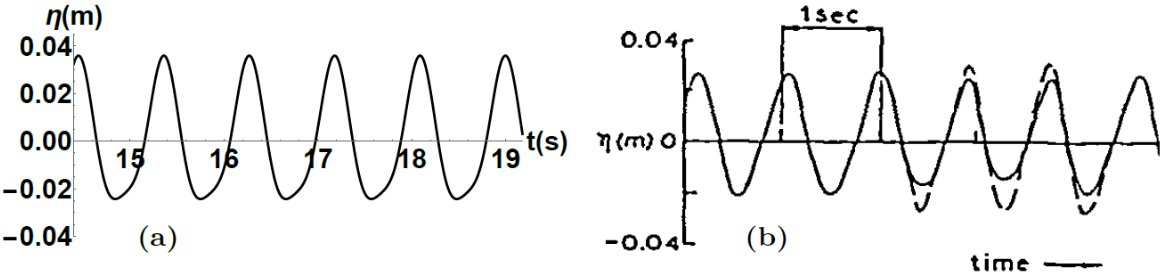

Figure 2.1: The elevation of the free surface at a measurement point located 1.8 m downstream: (a) obtained with the use of the new equation, (b) obtained in [2] experimentally (solid line) and numerically (dashed line).

In total, two validations were conducted with the results published in the works of Siegel et al. [3,4]. The first case corresponds to the results of the 1:300 scale experiment which was conducted in the 2D wave tunnel of the US Air Force Academy [3]. Fig. 2.2(b) presents the upstream (black line) and downstream (grey line) free surface elevation caused by the single rotating hydrofoil. The experimental setup has the following parameters: rotor rotation period Tr = 0.55 s, blade pitch α = -

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

7.5°, operational radius R = 6 cm, chord length S = 5 cm, submergence y_0 = -7.5 cm. We have determined the lift coefficient as C_L = -0.65 and the circulation as $\Gamma = C_L \omega RS/2$. The numerical simulation, with the use of new formulae, shows good agreement with the amplitude and period of the experimentally radiated waves, as seen in Fig. 2.2.

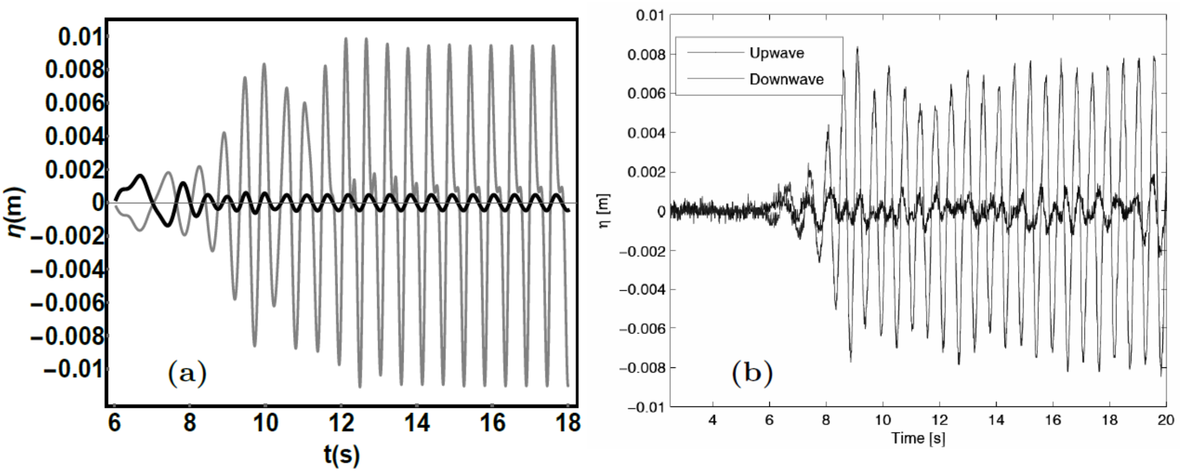

Figure 2.2: The elevation of the free surface at the measurement points located downstream (grey line) and upstream (black line): (a) obtained with the use of the new equation, (b) obtained experimentally in [3].

The last validation case is the comparison with the numerical simulation presented in [4]. The parameters were normalised by a period of T = 9 s and wavelength λ_{Airy} = 126.5 m. The single hydrofoil rotor has radius R/λ_{Airy} = 0.15, submergence depth $|y_0|/\lambda_{Airy}$ = 0.18, and circulation $\Gamma T/\lambda_{Airy}^2$= 5.6*10^{-3}. All waves are evaluated at $x = \pm 3\lambda_{Airy}$ and at time t/T = 30 after the start of the cycloidal WEC.

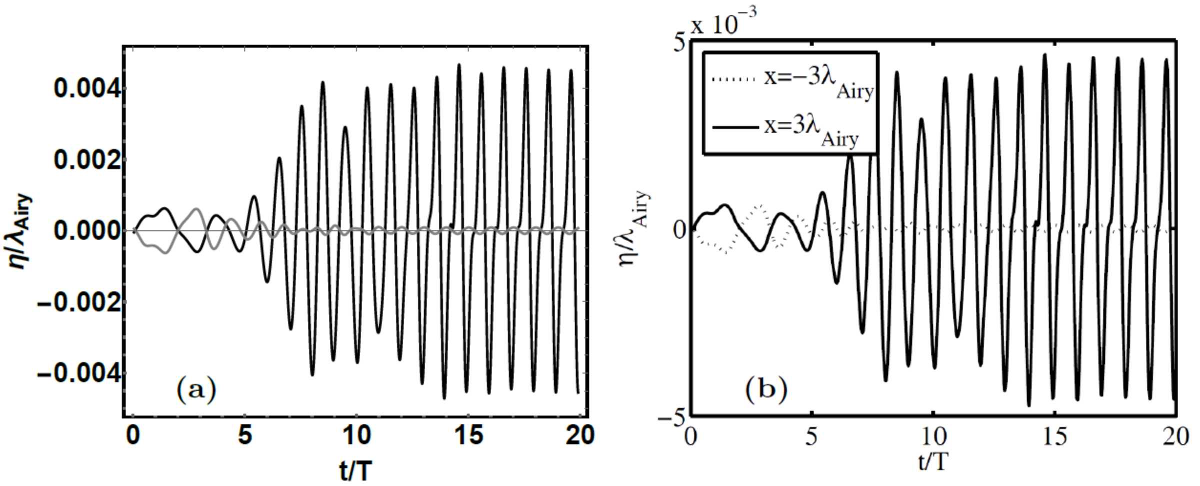

Figure 2.3: The elevation of the free surface at the measurement points located downstream (black line) and upstream (grey line): (a) obtained with the use of new equation, (b) obtained in [4] numerically.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

The numerical results obtained in [4] are presented in Fig. 3(b), where the upstream (black line) and downstream (grey line) are in very good agreement with our numerical results, shown in Fig. 2.3(a).

Figs. 2.3 and 2.4 also show that the amplitude of upstream radiated waves (blue and grey lines) are more than ten times less than the amplitude of the waves radiated downstream (green and black lines). Assuming that the downstream radiated waves have the same amplitude as incoming wave, but opposite phase, we can achieve complete wave energy absorption [12]. As a result, the interaction between the upstream radiated waves and the incoming waves can be ignored, due to the significant amplitude differences. This effect allows us to reliably forecast the wave-induced fluid velocity. Thus, the derived equations (21) can be very beneficial for the calculation of the most recent performance metrics proposed by Siegel [12], which are based on wave radiation and cancellation effects.

2.2 Validation of the developed model against LiftWEC 2D experiments in terms of tangential and radial forces

In this subsection we model the 2D experiments which were conducted by LiftWEC WP04: Physical Modelling. The parameters of the experimental setup can be found in the LiftWEC Project Deliverables [5,6] and the obtained experimental data can be found in [7].

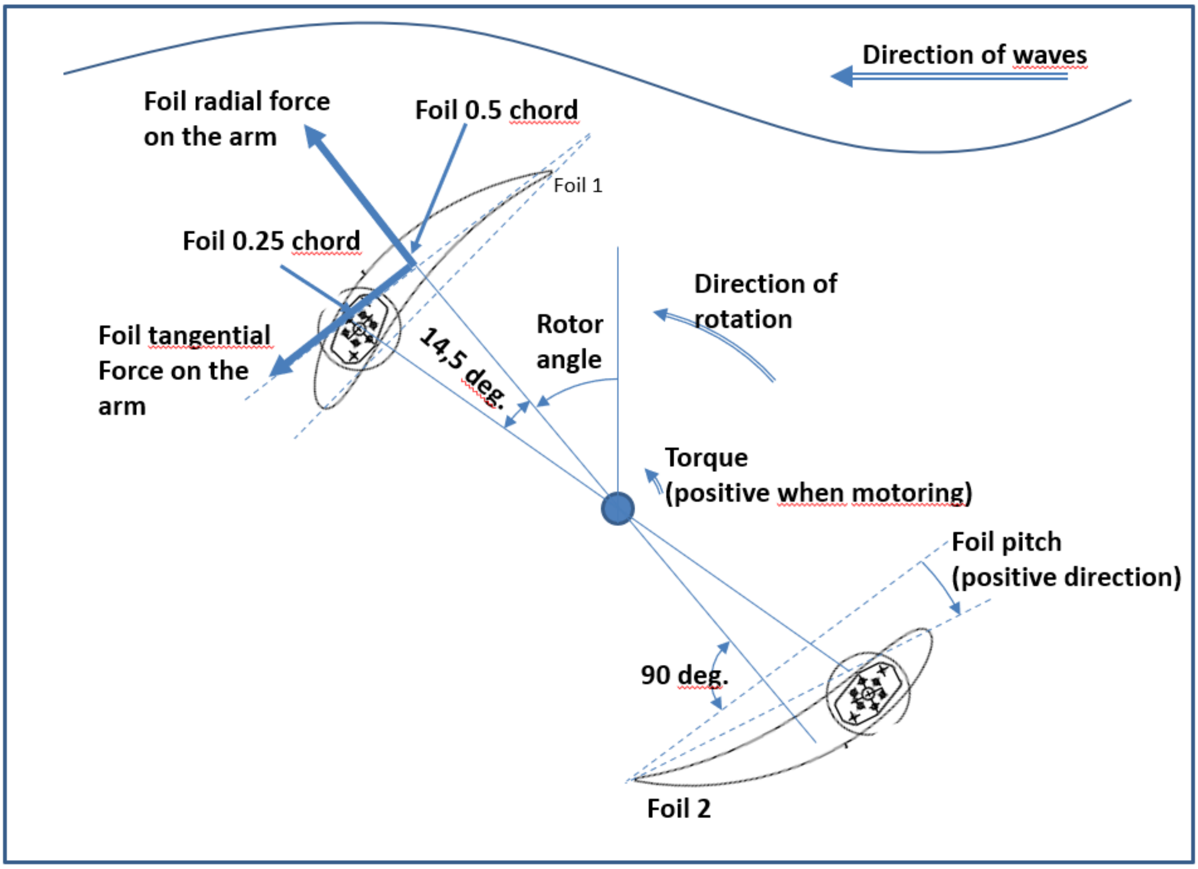
Direction of waves
Foil radial force on the arm
Foil 0.5 chord
Foil 1
Foil 0.25 chord
Direction of rotation
Rotor angle
Foil tangential Force on the arm
14,5 deg.
Torque (positive when motoring)
Foil pitch (positive direction)
90 deg.
Foil 2

Figure 2.4: The scheme of the experimental setup

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

The parameters of the laboratory facilities and rotor are presented in the table below:

Description	Unit	Dimension
Tank length	m	140
Tank width	m	5
Tank depth	m	3
Sub-channel width	m	0.49
Sub-channel length	m	17.3
Sub-channel height	m	3.3
Rotor diameters	m	0.5, 0.6 or 0.75
Pitch angles	degrees	[-12, -8, -4, 0, 4, 8, 12]
Hydrofoil profile		Curved NACA0015/NACA3515
Depth of rotor axis	m	0.755
Hydrofoil chord length	m	0.3
Hydrofoil chord curve radius	m	0.3
Number of hydrofoils		1 or 2

The validation is conducted in term of the tangential and radial forces; thus, we have derived the Lift and Drag coefficient which can describe the following forces values using a point source model:

$$\begin{bmatrix} F_{T1} \\ F_{R1} \\ F_{T2} \\ F_{R2} \end{bmatrix} = \begin{bmatrix} \sin(\alpha-\gamma) & -\cos(\alpha-\gamma) & 0 & 0 \\ \cos(\alpha-\gamma) & -\sin(\alpha-\gamma) & 0 & 0 \\ 0 & 0 & \sin(\alpha-\gamma) & -\cos(\alpha-\gamma) \\ 0 & 0 & \cos(\alpha-\gamma) & -\sin(\alpha-\gamma) \end{bmatrix} * \frac{1}{2}\rho S|\hat{V}|^2 * \begin{bmatrix} C_{L1} \\ C_{D1} \\ C_{L2} \\ C_{D2} \end{bmatrix} \qquad (31)$$

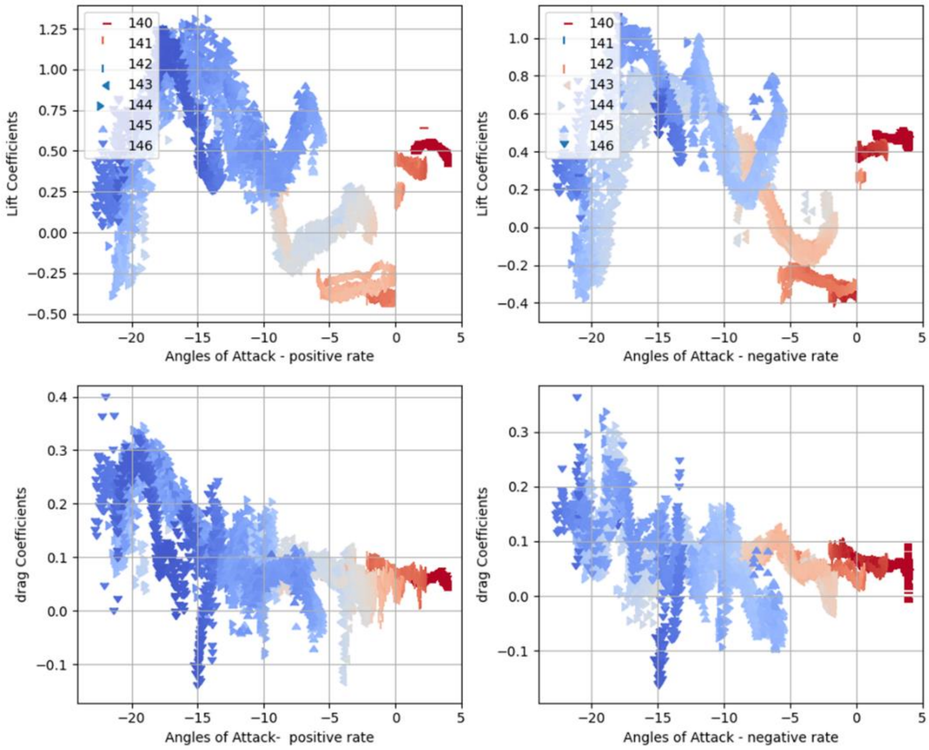

Figure 2.5: The derived values of lift and drag coefficient for the case 140-146

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

Case 143 – Single hydrofoil

Here we present the results of validation of the case 143 from the experimental data [6]. In this case the rotation of the single hydrofoil with pitch angle $\gamma = 4$ and chord length C=0.3 m in monochromatic waves has been studied. The incoming waves have the wave period T=1.8 s and wave height H=0.253 m. The rotor's radius is R=0.3m, it rotates with the wave frequency with a phase angle of 90 degrees.

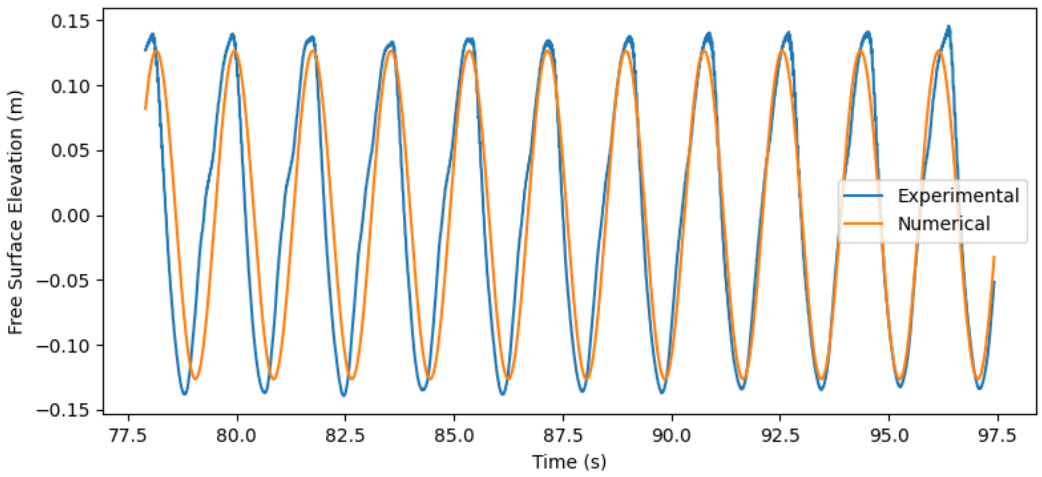

Figure 2.6: The theoretical and experimental elevation of the free surface at the gauge point 5

The time interval from 77 to 98 seconds has been considered and incoming waves were approximated by Airy functions (7,8) see Fig. 2.5. The system (31) has been solved on the each of the timesteps and the lift and drag coefficients have been derived (see Fig 2.5). Off course, the obtained coefficients play the role of the tuning parameters for specific conditions.

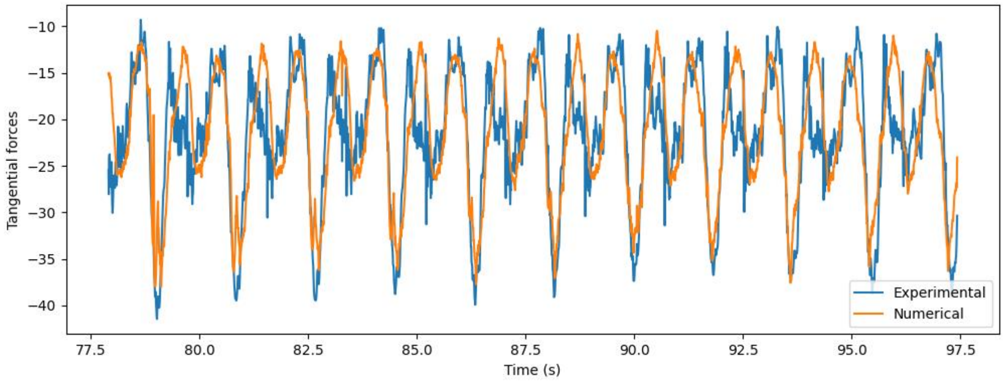

Figure 2.7: Comparison of the experimental and numerical values of the Tangential force.

However, they allow us to have acceptable agreement with the experimental data and simulate the changes of the tangential and radial for the foil which rotates in waves. The results of the simulations and their comparison with the experimental data are presented on Figs 2.7 and 2.8.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

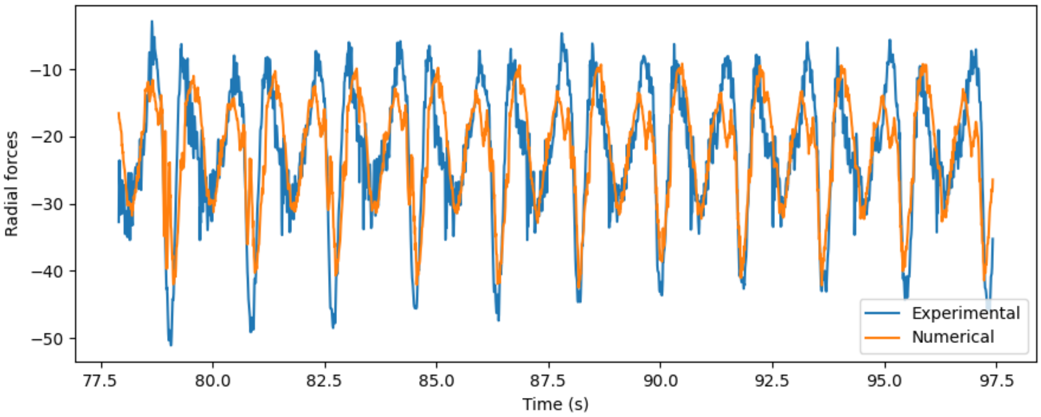

Figure 2.8: Comparison of the experimental and numerical values of the Radial force.

The more examples of the developed analytical model against the experimental results will be presented at the deliverable D3.3 Tool validation and extension report [17]

2.3 CONCLUSION

The presented model can describe a cyclorotor-based WEC with various configurations, leading to its potential use in the optimisation of the rotor design. A number of simulations, which can be conducted with the use of this model, can help to determine the potential benefits of cyclorotor-based WECs and their perspective. However, it may require derivation of new lift and drag coefficients for the selected operational conditions. These coefficients could potentially be obtained from CFD simulations, or experimental results.

The newly derived exact analytical formulae for free surface elevation, and perturbation in fluid velocity, caused by a rotating foil can also help developers of existing and new cyclorotor concepts. The presented equations can describe rotors with various numbers of foils and configurations, leading to a potential use in the optimisation of cyclorotor design.

The developed models are validated and fast, lending themselves to the control design. For example, the point source model is suitable for model predictive control, since it takes approximately 4 seconds to calculate a 1-minute forecast. Customised coding would likely reduce this computational time by an order of magnitude. The model is suitable for development of various control strategies which target different performance metrics, such as the wave cancellation proposed by Atargis [12] or the maximisation of the power coefficient (6).

3 BIBLIOGRAPHY

1. Atargis, The energy corporation website. https://atargis.com//. Accessed 10 September 2021

2. Hermans AJ, Sabben VE, Pinkster JA (1990) A device to extract energy from water waves. Applied Ocean Research, 12(4), pp 175-179

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

3. Siegel S, Romer M, Imamura J, Fagley C, McLaughlin T (2011) Experimental wave generation and cancellation with a cycloidal wave energy converter. Proceedings of 30th International Conference on Ocean, Offshore and Arctic Engineering, Rotterdam, The Nether-lands, pp 347-357

4. Siegel S, Jeans T, McLaughlin T (2011) Deep ocean wave energy conversion using a cycloidal turbine. Applied Ocean Research, 33 pp 110–119

5. LW-D04-02-1x0 2D scale model, LiftWEC Consortium https://liftwec.com/. Accessed 10 September 2021

6. LW-D04-03-1x0 Open access experimental data from Liftwec 2D scale model, LiftWEC Consortium https://liftwec.com/. Accessed 10 September 2021

7. LW-D04-03-1x0 Open access experimental data from Liftwec 2D scale model, LiftWEC Consortium https://liftwec.com/. Accessed 10 September 2021

8. Ermakov, A., Ringwood, J.V. A control-orientated analytical model for a cyclorotor wave energy device with N hydrofoils. J. Ocean Eng. Mar. Energy 7, 201–210 (2021).

9. Ermakov, A., Ringwood, J.V. Development of an analytical model for a cyclorotor wave energy device,” in Conference processing EWTEC2021, paper #1885, Plymouth, UK, 2021

10. Ermakov, A., Ringwood, J.V. Erratum to: A control-orientated analytical model for a cyclorotor wave energy device with N hydrofoils. J. Ocean Eng. Mar. Energy 7, 493–494 (2021)

11. Scharmann N, Ocean energy conversion systems: the wave hydro-mechanical rotary energy converter. PhD Thesis, Institute for Fluid Dynamics and Ship Theory, TUHH, Hamburg, Germany 2014.

12. Siegel S, Numerical benchmarking study of a cycloidal wave energy converter. Renewable Energy, 134, pp 390-405, 2019

13. Wehausen J, Laitone E, Surface Waves, Handbook of Physics, vol. 9, Springer-Verlag, 1960

14. Dawson, H.G. On the numerical value of $\int_0^h \exp(x^2)\ dx$. Proceedings of the London Math-ematical Society. s1-29 (1), pp. 519–522, 1897

15. Siegel, S. G. Cycloidal wave energy converter, Final scientific report: DOE Grant DE-EE0003635 2013

16. R. Sheldahl and P. Klimas, Aerodynamic characteristics of seven symmetrical airfoil sections through 180-degree angle of attack for use in aerodynamic analysis of vertical axis wind turbines. Sandia National Labs., Albuquerque, 1981

17. D3.3 Tool validation and extension report, LiftWEC Consortium https://liftwec.com/. To be published November 2021

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

4 APPENDIX

In the appendix, we present the Python code which describes the numerical simulations of the rotor which rotates in phase with incoming monochromatic waves with T=10s and H=2m. As an example, we consider a rotor with two symmetric hydrofoils NACA0015, which is similar to the CycWEC prototype tested by Atargis in a 3D wave tank at the Texas A&M Offshore Technology Research Centre [15]. The selected rotor has a submergence depth y_0 = 2m, operational radius R = 1m and chord length C = 0.75m. The lift and drag coefficients used for the point source method were taken from [16]. The program code's use of the new model's equations outlined in the first section and converted to the case of the rotor with two hydrofoils and polar coordinate system. A similar case is presented in the article [9].

The program provides the figures for free surface elevation and mechanical forces after synchronisation of the cyclorotor with the incoming wave.

```
from cmath import *
from math import *
import numpy as np
import matplotlib.pyplot as plt
from scipy.io import loadmat
import scipy.special as sp
from scipy.interpolate import interp1d,RegularGridInterpolator
# Environmental parameters
g = 9.81                    # gravity
rho = 1000                  # water density
h = 50                      # ocean depth
# Foil parameters
S = 5                  # chord length
R = 6                  # radius of the rotor -- OPTIMISATION PARAMETER 1
Width = 1                # width of the rotor
Inertia = 0             # inertia
y0 = -12                 # submergence depth of the rotor -- OPTIMISATION PARAMETER 2
# Timesteps of the Numerical Model
Sect = 100       # Timesteps per wave period
Num = 1200       # Number of steps
Memo = 2 * Sect    # Lifetime of the hydrofoils wakes until its dessipation
#Wave parameters
H0=2
T=10
# Lift and Drag coefficients
Data_drag = np.loadtxt('Drag_coef_matrix_AoAxRe.txt')
Data_lift = np.loadtxt('Lift_coef_matrix_AoAxRe.txt')
Re_list = Data_drag[0,1:]      # Re values are the first line
AoA_list = Data_drag[1:,0]      # AoA values are the first column
```

```
Drag_fun = RegularGridInterpolator((AoA_list,Re_list),Data_drag[1:,1:])
Lift_fun = RegularGridInterpolator((AoA_list,Re_list),Data_lift[1:,1:])

# calculation of the wakes caused by moving foils
def dynam (circ1, circ2, dtau, theta, theta_k, dt):

   M1 = -2*y0 + R*np.sin(theta) + R*np.sin(theta_k) + R*np.cos(theta)*complex(0,1) -
R*np.cos(theta_k)*complex(0,1)
   M2 = -2*y0 + R*np.sin(theta) - R*np.sin(theta_k) + R*np.cos(theta)*complex(0,1) +
R*np.cos(theta_k)*complex(0,1)
   M3 = -2*y0 - R*np.sin(theta) - R*np.sin(theta_k) - R*np.cos(theta)*complex(0,1) +
R*np.cos(theta_k)*complex(0,1)
   M4 = -2*y0 - R*np.sin(theta) + R*np.sin(theta_k) - R*np.cos(theta)*complex(0,1) -
R*np.cos(theta_k)*complex(0,1)

   phix = ( np.sqrt(g)/(2*np.pi) * circ1/M1**(5/2) * ( sp.dawsn(dtau / (2*np.sqrt(M1))) * (dtau**2 -
2*M1) - dtau*np.sqrt(M1) )
         + np.sqrt(g)/(2*np.pi) * circ2/M2**(5/2) * ( sp.dawsn(dtau / (2*np.sqrt(M2))) * (dtau**2 -
2*M2) - dtau*np.sqrt(M2) ) )*dt
   phiy = ( np.sqrt(g)/(2*np.pi) * circ2/M3**(5/2) * ( sp.dawsn(dtau / (2*np.sqrt(M3))) * (dtau**2 -
2*M3) - dtau*np.sqrt(M3) )
         + np.sqrt(g)/(2*np.pi) * circ1/M4**(5/2) * ( sp.dawsn(dtau / (2*np.sqrt(M4))) * (dtau**2 -
2*M4) - dtau*np.sqrt(M4) ) )*dt

   Id1 = phix.real
   Id2 = -phix.imag
   Id3 = phiy.real
   Id4 = -phiy.imag

   return np.array([Id1,Id2,Id3,Id4])

#Simulation of the rotation of the cyclorotoe
def Simulation(X, H, Tp):

   pitch = X
   pitch_rad = np.pi * pitch / 180

   omega = 2 * np.pi / Tp # wave frequency
   kwave = omega**2 / g   # wave number
   leng = 2 * np.pi / kwave # wave length

   dt = Tp / Sect  # timestep
   Time = np.arange(0, dt*Num, dt)

   Position = np.arange(0, omega*dt*Num, omega*dt)
```

```
  Velocity = np.ones(Num) * omega
  Acceleration = np.zeros(Num)

  Circ1 = [0] * Num        # initialisation of foil n°1 circulation gamma1
  Circ2 = [0] * Num        # initialisation of foil n°2 circulation gamma2

  Angle1 = [0] * Num        # initialisation of foil n°1 attack angle alpha1
  Angle2 = [0] * Num        # initialisation of foil n°2 attack angle alpha2

  FTan1 = np.zeros(Num)        # initialisation of foil n°1 tangential force Ft1
  FTan2 = np.zeros(Num)        # initialisation of foil n°2 tangential force Ft2

  FRad1 = np.zeros(Num)        # initialisation of foil n°1 radial force Fr1
  FRad2 = np.zeros(Num)        # initialisation of foil n°2 radial force Fr2

  FLift1 = np.zeros(Num)        # initialisation of foil n°1 lift force Fr1
  FLift2 = np.zeros(Num)        # initialisation of foil n°2 lift force Fr2

  FDrag1 = np.zeros(Num)        # initialisation of foil n°1 drag force Fr1
  FDrag2 = np.zeros(Num)        # initialisation of foil n°2 drag force Fr2

  Ix1 = 0                  # initialisation of velocities due to radiated waves (instantanous radiation Vhm
and wake Vhw) due to the rotating foils
  Iy1 = 0
  Ix2 = 0
  Iy2 = 0

  Energy = 0                  # initialisation of energy absorbed from the end of the transition part to the
end of simulation

  for i, t in enumerate(Time[0:-2]):

    theta = Position[i]
    dtheta = Velocity[i]
    ddtheta = Acceleration[i]

    # Rotational velocity hydrofoil 1
    VRx1 = - R * np.sin(theta) * dtheta
    VRy1 = - R * np.cos(theta) * dtheta
    # Rotational velocity hydrofoil 2
    VRx2 = R * np.sin(theta) * dtheta
    VRy2 = R * np.cos(theta) * dtheta

    # Wave induced velocity hydrofoil 1 (Vw)
    VAx1 = (g * H * kwave / (2 * omega)) * np.exp(kwave * (y0 - R * np.sin(theta))) * \
        np.cos(omega * t - kwave * R * np.cos(theta))
    VAy1 = - (g * H * kwave / (2 * omega)) * np.exp(kwave * (y0 - R * np.sin(theta))) * \
        np.sin(omega * t - kwave * R * np.cos(theta))

    # Wave induced velocity hydrofoil 2
    VAx2 = (g * H * kwave / (2 * omega)) * np.exp(kwave * (y0 + R * np.sin(theta))) * \
        np.cos(omega * t + kwave * R * np.cos(theta))
    VAy2 = - (g * H * kwave / (2 * omega)) * np.exp(kwave * (y0 + R * np.sin(theta))) * \
        np.sin(omega * t + kwave * R * np.cos(theta))

    # Components of the overall relative velocity
    VHx1 = VAx1 + Ix1 - VRx1
    VHy1 = VAy1 + Iy1 - VRy1
    VHx2 = VAx2 + Ix2 - VRx2
    VHy2 = VAy2 + Iy2 - VRy2

    # Angles of attack
    AoA1 = pitch + 180 * np.arcsin((VRx1 * VHy1 - VRy1 * VHx1) /(np.sqrt(VRx1 * VRx1 + VRy1 * VRy1) * np.sqrt(VHx1 * VHx1 + VHy1 * VHy1))) / np.pi
    AoA2 = - pitch + 180 * np.arcsin((VRx2 * VHy2 - VRy2 * VHx2) /(np.sqrt(VRx2 * VRx2 + VRy2 * VRy2) * np.sqrt(VHx2 * VHx2 + VHy2 * VHy2))) / np.pi

    # Interpolation of lift and drag coefficients as a function of the angle of attack
    # using the given table of data and for the calculated value of the angle of attack

    flow_speed1 = np.sqrt(VHx1**2 + VHy1**2) # in m/s
    Re1 = rho*flow_speed1*S/0.001 # dynamic viscosity = 0.001Pa*s = 0.001kg/m-s

    if Re1 > 10000000 :
      Re1 = 10000000
    if AoA1 < 0 :
      Cl1 = -Lift_fun((abs(AoA1),Re1))
      Cd1 = Drag_fun((abs(AoA1),Re1))
    else :
      Cl1 = Lift_fun((AoA1,Re1))
      Cd1 = Drag_fun((AoA1,Re1))

    flow_speed2 = np.sqrt(VHx2**2 + VHy2**2)
    Re2 = rho*flow_speed2*S/0.001

    if Re2 > 10000000 :
      Re2 = 10000000
    if AoA2 < 0 :
      Cl2 = -Lift_fun((abs(AoA2),Re2))
      Cd2 = Drag_fun((abs(AoA2),Re2))
```

```
        else :
            Cl2 = Lift_fun((AoA2,Re2))
            Cd2 = Drag_fun((AoA2,Re2))

##          Cl1 = np.interp(AoA1, AoA_list, Lift) # gives the interpolated value at the point AoA1
##          Cd1 = np.interp(AoA1, AoA_list, Drag)
##          Cl2 = np.interp(AoA2, AoA_list, Lift)
##          Cd2 = np.interp(AoA2, AoA_list, Drag)

        Angle1[i] = AoA1
        Angle2[i] = AoA2
        AoA1rad = np.pi * AoA1 / 180
        AoA2rad = np.pi * AoA2 / 180

        # Tangential force due to the lift and drag forces on the hydrofoils
        FTan1[i] = (1 / 2) * rho * S * (Cl1 * np.sin(AoA1rad-pitch_rad) - Cd1 * np.cos(AoA1rad-pitch_rad)) * (VHx1 * VHx1 + VHy1 * VHy1)
        FTan2[i] = (1 / 2) * rho * S * (Cl2 * np.sin(AoA2rad+pitch_rad) - Cd2 * np.cos(AoA2rad+pitch_rad)) * (VHx2 * VHx2 + VHy2 * VHy2)
        FRad1[i] = (1 / 2) * rho * S * (Cl1 * np.cos(AoA1rad-pitch_rad) + Cd1 * np.sin(AoA1rad-pitch_rad)) * (VHx1 * VHx1 + VHy1 * VHy1)
        FRad2[i] = (1 / 2) * rho * S * (Cl2 * np.cos(AoA2rad+pitch_rad) + Cd2 * np.sin(AoA2rad+pitch_rad)) * (VHx2 * VHx2 + VHy2 * VHy2)

        FLift1[i] = (1 / 2) * rho * S * (Cl1) * (VHx1 * VHx1 + VHy1 * VHy1)
        FLift2[i] = (1 / 2) * rho * S * (Cl2) * (VHx2 * VHx2 + VHy2 * VHy2)
        FDrag1[i] = (1 / 2) * rho * S * (Cd1) * (VHx1 * VHx1 + VHy1 * VHy1)
        FDrag2[i] = (1 / 2) * rho * S * (Cd2) * (VHx2 * VHx2 + VHy2 * VHy2)



        # Circulations
        Circ1[i] = (1 / 2) * S * Cl1  * np.sqrt(VHx1 * VHx1 + VHy1 * VHy1)
        Circ2[i] = (1 / 2) * S * Cl2  * np.sqrt(VHx2 * VHx2 + VHy2 * VHy2)

        # Calculation of the static radiation at this timestep
        gamma1 = Circ1[i]
        gamma2 = Circ2[i]

        I1s = (gamma2 / (4 * np.pi)) * ( y0 / (y0 * y0 + R * R * np.cos(theta) * np.cos(theta)) + np.sin(theta) / R)
        I2s = (gamma2 *np.cos(theta) / (4 * np.pi * R)) * ((y0 * y0 - R * R * np.sin(theta) * np.sin(theta)) /(y0 * y0 + 4 * R * R * np.cos(theta) * np.cos(theta)))
        I3s = (gamma1 / (4 * np.pi)) * (y0 / (y0 * y0 + R * R * np.cos(theta) * np.cos(theta)) - np.sin(theta) / R)
        I4s = - (gamma1 * np.cos(theta) / (4 * np.pi *R)) * ((y0 * y0  + R * R * np.sin(theta) * np.sin(theta))
/(y0 * y0 + R * R * np.cos(theta) * np.cos(theta)))

        # Calculation of the dynamic radiation at this timestep
        Id = np.array([0,0,0,0])
        # Integral members for radiated waves
        if i - Memo > 0:
            k = i - Memo # Memo is the duration of the wakes
        else:
            k = 1
        tau_k = Time[k]
        dtau_k = np.sqrt(g) * (t - tau_k)
        theta_k = Position[k]
        Id = Id + dynam (Circ1[k], Circ2[k], dtau_k, theta, theta_k, dt) /2

        for j in range(k+1, i):
            tau_j = Time[j]
            dtau_j = np.sqrt(g) * (t - tau_j)
            theta_j = Position[j]
            Id += dynam (Circ1[j], Circ2[j], dtau_j, theta, theta_j, dt)
        # end of for(j)

        tau_new = Time[i]
        dtau_new = np.sqrt(g) * (t - tau_new)
        theta_new = Position[i]
        Id = Id + dynam (Circ1[i], Circ2[i], dtau_new, theta, theta_new, dt) /2
        Ix1 = I1s + Id[0]
        Iy1 = I2s + Id[1]
        Ix2 = I3s + Id[2]
        Iy2 = I4s + Id[3]

    # wave cancelation :
    FreeSurfDownS = [0]* Num
    TransmittedDownWave = [0] * (Num-10)
    OriginalWave = [0] * (Num-10)
    timesurf=np.arange(0, dt*(Num), dt)[6:-4]
    for ii in np.arange(0,Num-10):
        A = kwave*leng - omega * timesurf[ii]
        OriginalWave[ii] = H0/2*cos(A)
        TransmittedDownWave[ii] = H0/2*cos(A)

    # Free surface peturbation downwave caused by momentary radiation from the hydrofoil
    def StatRadDownS (gamma,X,Y) :
        return ( gamma * np.arctan( 2*Y*(X-leng) / ( (leng-X)**2 - Y**2 ) ) / (2*np.pi) )
    # Free surface peturbation downwave caused by dynamic radiation from the hydrofoil
    def DynamRadDownS (gamma,X,Y,t,tau) :
        return ( np.sqrt(g)*gamma/np.pi * ( ( (-1)**(1/4) * sp.dawsn( ((-1)**(3/4)*np.sqrt(g)*(t-tau)) /
(2*np.sqrt(complex(leng-X,Y)))) / np.sqrt(complex(leng-X,Y)) ) + ( (-1)**(3/4) * sp.dawsn( ((-
1)**(1/4)*np.sqrt(g)*(t-tau)) / (2*np.sqrt(complex(leng-X,-Y)))) / np.sqrt(complex(leng-X,-Y)) ) ) )

    # Callculation of the free surface elevation downwave
    for i in range ( Num - 3 ) :
        theta_i = Position[i]
        X1_i = R * np.cos(theta_i)
        Y1_i = y0 - R * np.sin(theta_i)
        X2_i =  - R * np.cos(theta_i)
        Y2_i = y0 + R * np.sin(theta_i)
        FreeSurfDownS[i] = FreeSurfDownS[i] + StatRadDownS(Circ1[i],X1_i,Y1_i) +
StatRadDownS(Circ2[i],X2_i,Y2_i)
        theta_0 = Position[0]
        X1_0 = R * np.cos(theta_0)
        Y1_0 = y0 - R * np.sin(theta_0)
        X2_0 =  - R * np.cos(theta_0)
        Y2_0 = y0 + R * np.sin(theta_0)
        FreeSurfDownS[i] = FreeSurfDownS[i] + ( DynamRadDownS(Circ1[0],X1_0,Y1_0,Time[i],Time[0]) +
DynamRadDownS(Circ2[0],X2_0,Y2_0,Time[i],Time[0]) )*dt/2

        for j in range (1,i-1) :
            theta_j = Position[j]
            X1_j = R * np.cos(theta_j)
            Y1_j = y0 - R * np.sin(theta_j)
            X2_j =  - R * np.cos(theta_j)
            Y2_j = y0 + R * np.sin(theta_j)
            FreeSurfDownS[i] = FreeSurfDownS[i] + ( DynamRadDownS(Circ1[j],X1_j,Y1_j,Time[i],Time[j]) +
DynamRadDownS(Circ2[j],X2_j,Y2_j,Time[i],Time[j]) )*dt

        FreeSurfDownS[i] = FreeSurfDownS[i] + ( DynamRadDownS(Circ1[i],X1_i,Y1_i,Time[i],Time[i]) +
DynamRadDownS(Circ2[i],X2_i,Y2_i,Time[i],Time[i]) )*dt/2

    # end for (i)

    # radiated wave downstream
    TransmittedDownWave = TransmittedDownWave + (-1/g* (np.array(FreeSurfDownS[6:-4]) -
np.array(FreeSurfDownS[5:-5])) /dt).real

    return
(OriginalWave,TransmittedDownWave,FTan1,FTan2,FRad1,FRad2,FLift1,FLift2,FDrag1,FDrag2,Time,ti
mesurf)
```

```
# end of energy function definition
# run of the program
Answer = Simulation(0, H0, T)

# presentation of the simulation results

plt.plot((Answer[-1][-2*Sect+4:-Sect+5]-10*T)/T,Answer[0][-2*Sect+4:-Sect+5])
plt.plot((Answer[-1][-2*Sect+4:-Sect+5]-10*T)/T,Answer[1][-2*Sect+4:-Sect+5])
plt.legend(['Incoming Wave','Transmitted wave'])
plt.ylabel('Free surface elevation (m)')
plt.xlabel('Time (t/T)')
plt.show()

plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[2][-2*Sect+4:-Sect+5])
plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[3][-2*Sect+4:-Sect+5])
plt.legend(['Foil 1','Foil 2'])
plt.ylabel('Tangential Forces (N)')
plt.xlabel('Time (t/T)')
plt.show()

plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[4][-2*Sect+4:-Sect+5])
plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[5][-2*Sect+4:-Sect+5])
plt.legend(['Foil 1','Foil 2'])
plt.ylabel('Radial Forces (N)')
plt.xlabel('Time (t/T)')
plt.show()

plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[6][-2*Sect+4:-Sect+5])
plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[7][-2*Sect+4:-Sect+5])
plt.legend(['Foil 1','Foil 2'])
plt.ylabel('Lift Forces (N)')
plt.xlabel('Time (t/T)')
plt.show()

plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[8][-2*Sect+4:-Sect+5])
plt.plot((Answer[-2][-2*Sect+4:-Sect+5]-10*T)/T,Answer[9][-2*Sect+4:-Sect+5])
plt.legend(['Foil 1','Foil 2'])
plt.ylabel('Drag Forces (N)')
plt.xlabel('Time (t/T)')
plt.show()
```

Drag_coef_matrix_AoAxRe.txt

```
-1  10000  20000  40000  80000  160000  360000  700000  1000000  2000000  5000000  10000000
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

```
0 0.036 0.0265 0.0196 0.0147 0.0115 0.0091 0.0077 0.0074 0.007 0.0068 0.0068
1 0.0362 0.0267 0.0198 0.0148 0.0117 0.0092 0.0078 0.0075 0.0071 0.0069 0.0068
2 0.0366 0.0271 0.0202 0.0151 0.012 0.0094 0.008 0.0076 0.0072 0.007 0.0069
3 0.0373 0.0279 0.0209 0.0156 0.0124 0.0098 0.0083 0.0079 0.0075 0.0073 0.0071
4 0.0383 0.029 0.0219 0.0168 0.0132 0.0105 0.0089 0.0083 0.0078 0.0075 0.0074
5 0.0393 0.0303 0.0232 0.0181 0.0142 0.0114 0.0098 0.0091 0.0083 0.0077 0.0077
6 0.04 0.41 0.249 0.0197 0.016 0.0126 0.0108 0.0101 0.009 0.0081 0.0081
7 0.051 0.051 0.0267 0.0214 0.0176 0.0143 0.0122 0.0111 0.0098 0.0089 0.0086
8 0.064 0.064 0.052 0.0234 0.0193 0.0157 0.0135 0.0126 0.0108 0.0095 0.009
9 0.077 0.077 0.077 0.0255 0.0212 0.0173 0.0149 0.0138 0.0121 0.0102 0.0096
10 0.091 0.091 0.091 0.0277 0.0233 0.0191 0.0164 0.0152 0.0133 0.0113 0.0103
11 0.107 0.107 0.107 0.076 0.0256 0.0211 0.0182 0.0168 0.0146 0.0124 0.0114
12 0.123 0.123 0.123 0.123 0.0281 0.0233 0.02 0.0186 0.0161 0.0136 0.0123
13 0.14 0.14 0.14 0.14 0.0302 0.0257 0.0221 0.0205 0.0177 0.0149 0.0134
14 0.158 0.158 0.158 0.158 0.104 0.0283 0.0244 0.0225 0.0195 0.0164 0.0147
15 0.177 0.177 0.177 0.177 0.177 0.0312 0.0269 0.0249 0.0215 0.018 0.0161
16 0.196 0.196 0.196 0.196 0.197 0.124 0.0297 0.0275 0.0237 0.0198 0.0176
17 0.217 0.217 0.217 0.217 0.217 0.217 0.134 0.0303 0.0261 0.0218 0.0194
18 0.238 0.238 0.238 0.238 0.238 0.238 0.238 0.145 0.0288 0.024 0.0213
19 0.26 0.26 0.26 0.26 0.26 0.26 0.26 0.26 0.155 0.0265 0.0234
20 0.282 0.282 0.282 0.282 0.282 0.282 0.282 0.282 0.282 0.166 0.0257
21 0.305 0.305 0.305 0.305 0.305 0.305 0.305 0.305 0.305 0.305 0.177
22 0.329 0.329 0.329 0.329 0.329 0.329 0.329 0.329 0.329 0.329 0.329
23 0.354 0.354 0.354 0.354 0.354 0.354 0.354 0.354 0.354 0.354 0.354
24 0.379 0.379 0.379 0.379 0.379 0.379 0.379 0.379 0.379 0.379 0.379
25 0.405 0.405 0.405 0.405 0.405 0.405 0.405 0.405 0.405 0.405 0.405
26 0.432 0.432 0.432 0.432 0.432 0.432 0.405 0.432 0.432 0.432 0.432
27 0.46 0.46 0.46 0.46 0.46 0.46 0.46 0.46 0.46 0.46 0.46
30 0.57 0.57 0.57 0.57 0.57 0.57 0.57 0.57 0.57 0.57 0.57
35 0.745 0.745 0.745 0.745 0.745 0.745 0.745 0.745 0.745 0.745 0.745
40 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92
45 1.075 1.075 1.075 1.075 1.075 1.075 1.075 1.075 1.075 1.075 0.075
50 1.215 1.215 1.215 1.215 1.215 1.215 1.215 1.215 1.215 1.215 1.215
55 1.343 1.345 1.345 1.345 1.345 1.345 1.345 1.345 1.345 1.345 1.345
60 1.47 1.47 1.47 1.47 1.47 1.47 1.47 1.47 1.47 1.47 1.47
65 1.575 1.575 1.575 1.575 1.575 1.575 1.575 1.575 1.575 1.575 1.575
70 1.665 1.665 1.665 1.665 1.665 1.665 1.665 1.665 1.665 1.665 1.665
75 1.735 1.735 1.735 1.735 1.735 1.735 1.735 1.735 1.735 1.735 1.735
80 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78
85 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8
90 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8
95 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78 1.78
100 1.75 1.75 1.75 1.75 1.75 1.75 1.75 1.75 1.75 1.75 1.75
105 1.7 1.7 1.7 1.7 1.7 1.7 1.7 1.7 1.7 1.7 1.7
110 1.635 1.635 1.635 1.635 1.635 1.635 1.635 1.635 1.635 1.635 1.635
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.

```
115 1.555 1.555 1.555 1.555 1.555 1.555 1.555 1.555 1.555 1.555 1.555
120 1.465 1.465 1.465 1.465 1.465 1.465 1.465 1.465 1.465 1.465 1.465
125 1.35 1.35 1.35 1.35 1.35 1.35 1.35 1.35 1.35 1.35 1.35
130 1.225 1.225 1.225 1.225 1.225 1.225 1.225 1.225 1.225 1.225 1.225
135 1.085 1.085 1.085 1.085 1.085 1.085 1.085 1.085 1.085 1.085 1.085
140 0.925 0.925 0.925 0.925 0.925 0.925 0.925 0.925 0.925 0.925 0.925
145 0.755 0.755 0.755 0.755 0.755 0.755 0.755 0.755 0.755 0.755 0.755
150 0.575 0.575 0.575 0.575 0.575 0.575 0.575 0.575 0.575 0.575 0.575
155 0.42 0.42 0.42 0.42 0.42 0.42 0.42 0.42 0.42 0.42 0.42
160 0.32 0.32 0.32 0.32 0.32 0.32 0.32 0.32 0.32 0.32 0.32
165 0.23 0.23 0.23 0.23 0.23 0.23 0.23 0.23 0.23 0.23 0.23
170 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.111 0.14 0.14 0.14
175 0.055 0.055 0.055 0.055 0.055 0.055 0.055 0.055 0.055 0.055 0.055
180 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025
```

Lift_coef_matrix_AoAxRe.txt

```
-1 10000 20000 40000 80000 160000 360000 700000 1000000 2000000 5000000 10000000
 0 0 0 0 0 0 0 0 0 0 0 0
 1 0.0434 0.0891 0.1054 0.11 0.11 0.11 0.11 0.11 0.11 0.11 0.11
 2 0.0715 0.174 0.2099 0.22 0.22 0.22 0.22 0.22 0.22 0.22 0.22
 3 0.0725 0.2452 0.3078 0.33 0.33 0.33 0.33 0.33 0.33 0.33 0.33
 4 0.0581 0.3041 0.4017 0.4186 0.44 0.44 0.44 0.44 0.44 0.44 0.44
 5 0.0162 0.3359 0.4871 0.518 0.55 0.55 0.55 0.55 0.55 0.55 0.55
 6 -0.0781 0.3001 0.5551 0.5048 0.6299 0.66 0.66 0.66 0.66 0.66 0.66
 7 -0.1517 0.057 0.573 0.676 0.715 0.739 0.7483 0.77 0.77 0.77 0.77
 8 -0.1484 -0.1104 0.4663 0.7189 0.7851 0.824 0.8442 0.8504 0.88 0.88 0.88
 9 -0.1194 -0.105 0.0433 0.6969 0.8311 0.8946 0.926 0.9387 0.9574 0.99 0.99
 10 -0.0791 -0.0728 -0.0413 0.5122 0.8322 0.944 0.9937 1.0141 1.0433 1.0685 1.1
 11 -0.0348 -0.03 -0.0144 0.1642 0.7623 0.9572 1.0363 1.0686 1.1138 1.1553 1.1749
 12 0.0138 0.0173 0.0261 0.0749 0.5936 0.9285 1.0508 1.0971 1.1667 1.229 1.2591
 13 0.0649 0.0678 0.0741 0.0967 0.3548 0.8562 1.0302 1.0957 1.1948 1.2847 1.33
 14 0.1172 0.1193 0.1244 0.1382 0.2371 0.7483 0.9801 1.0656 1.1962 1.3187 1.3825
 15 0.1706 0.1721 0.1756 0.1861 0.2376 0.635 0.9119 1.0145 1.1744 1.3298 1.4136
 16 0.2242 0.2256 0.228 0.2364 0.2665 0.5384 0.8401 0.9567 1.1356 1.3186 1.4233
 17 0.278 0.2792 0.2815 0.2873 0.3098 0.4851 0.7799 0.8996 1.0921 1.2917 1.4136
 18 0.3319 0.3331 0.3351 0.3393 0.3567 0.4782 0.7305 0.8566 1.051 1.2576 1.3897
 19 0.3859 0.3869 0.3889 0.3927 0.4066 0.4908 0.7041 0.8226 1.0173 1.2242 1.3608
 20 0.4399 0.4409 0.4427 0.4463 0.4575 0.5247 0.699 0.8089 0.9954 1.1965 1.3325
 21 0.4939 0.4949 0.4966 0.5001 0.5087 0.5616 0.7097 0.8063 0.9837 1.1771 1.3077
 22 0.5479 0.5489 0.5506 0.5539 0.5611 0.6045 0.7298 0.8189 0.9827 1.1647 1.2767
 23 0.6019 0.6029 0.6045 0.6078 0.6148 0.6528 0.7593 0.8408 1.1611 1.1611 1.1981
 24 0.6559 0.6569 0.6585 0.6617 0.6685 0.7015 0.7961 0.8668 1.0078 1.1563 1.1538
 25 0.7099 0.7109 0.7125 0.7156 0.7224 0.7511 0.8353 0.9023 1.0317 1.1322 1.138
 26 0.7639 0.7649 0.7666 0.77 0.7771 0.8055 0.8353 0.9406 1.1268 1.1268 1.1374
 27 0.8174 0.8191 0.8222 0.8277 0.8382 0.8788 0.9473 0.9912 1.1397 1.1397 1.1519
```

```
30  0.855  0.855  0.855  0.855  0.855  0.855  0.855  0.855  0.855  0.855  0.855
35  0.98  0.98  0.98  0.98  0.98  0.98  0.98  0.98  0.98  0.98  0.98
40  1.035  1.035  1.035  1.035  1.035  1.035  1.035  1.035  1.035  1.035  1.035
45  1.05  1.05  1.05  1.05  1.05  1.05  1.05  1.05  1.05  1.05  1.05
50  1.02  1.02  1.02  1.02  1.02  1.02  1.02  1.02  1.02  1.02  1.02
55  0.955  0.955  0.955  0.955  0.955  0.955  0.955  0.955  0.955  0.955  0.955
60  0.875  0.875  0.875  0.875  0.875  0.875  0.875  0.875  0.875  0.875  0.875
65  0.76  0.76  0.76  0.76  0.76  0.76  0.76  0.76  0.76  0.76  0.76
70  0.63  0.63  0.63  0.63  0.63  0.63  0.63  0.63  0.63  0.63  0.63
75  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5
80  0.365  0.365  0.365  0.365  0.365  0.365  0.365  0.365  0.365  0.365  0.365
85  0.23  0.23  0.23  0.23  0.23  0.23  0.23  0.23  0.23  0.23  0.23
90  0.09  0.09  0.09  0.09  0.09  0.09  0.09  0.09  0.09  0.9  0.09
95  -0.05  -0.05  -0.05  -0.05  -0.05  -0.05  -0.05  -0.05  -0.05  -0.05  -0.05
100  -0.185  -0.185  -0.185  -0.185  -0.185  -0.185  -0.185  -0.185  -0.185  -0.185  -0.185
105  -0.32  -0.32  -0.32  -0.32  -0.32  -0.32  -0.32  -0.32  -0.32  -0.32  -0.32
110  -0.45  -0.45  -0.45  -0.45  -0.45  -0.45  -0.45  -0.45  -0.45  -0.45  -0.45
115  -0.575  -0.575  -0.575  -0.575  -0.575  -0.575  -0.575  -0.575  -0.575  -0.575  -0.575
120  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67
125  -0.76  -0.76  -0.76  -0.76  -0.76  -0.76  -0.76  -0.76  -0.76  -0.76  -0.76
130  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85
135  -0.93  -0.93  -0.93  -0.93  -0.93  -0.93  -0.93  -0.93  -0.93  -0.93  -0.93
140  -0.98  -0.98  -0.98  -0.98  -0.98  -0.98  -0.98  -0.98  -0.98  -0.98  -0.98
145  -0.9  -0.3  -0.9  -0.9  -0.9  -0.9  -0.9  -0.9  -0.9  -0.9  -0.9
150  -0.77  -0.77  -0.77  -0.77  -0.77  -0.77  -0.77  -0.77  -0.77  -0.77  -0.77
155  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67  -0.67
160  -0.635  -0.635  -0.635  -0.635  -0.635  -0.635  -0.635  -0.635  -0.635  -0.635  -0.635
165  -0.68  -0.68  -0.68  -0.68  -0.68  -0.68  -0.68  -0.68  -0.68  -0.68  -0.68
170  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85  -0.85
175  -0.66  -0.66  -0.66  -0.66  -0.66  -0.66  -0.66  -0.66  -0.66  -0.66  -0.66
180  0  0  0  0  0  0  0  0  0  0  0
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 851885. This output reflects the views only of the author(s), and the European Union cannot be held responsible for any use which may be made of the information contained therein.