A hybrid linear potential flow - machine learning model for enhanced prediction of WEC performance

Claes Eskilsson, Sepideh Pashami, Anders Holst and Johannes Palm

Abstract—**Linear potential flow (LPF) models remain the tools-of-the trade in marine and ocean engineering despite their well-known assumptions of small amplitude waves and motions. As of now, nonlinear simulation tools are still too computationally demanding to be used in the entire design loop, especially when it comes to the evaluation of numerous irregular sea states. In this paper we aim to enhance the performance of the LPF models by introducing a hybrid LPF-ML (machine learning) approach, based on identification of nonlinear force corrections. The corrections are defined as the difference in hydrodynamic force (viscous and pressure-based) between high-fidelity CFD and LPF models. Using prescribed chirp motions with different amplitudes, we train a long short-term memory (LSTM) network to predict the corrections. The LSTM network is then linked to the MoodyMarine LPF model to provide the nonlinear correction force at every time-step, based on the dynamic state of the body and the corresponding forces from the LPF model. The method is illustrated for the case of a heaving sphere in decay, regular and irregular waves – including passive control. The hybrid LPF-model is shown to give significant improvements compared to the baseline LPF model, even though the training is quite generic.**

Index Terms—**Linear potential flow, machine learning, recurrent neural network, floating bodies, wave energy**

I. Introduction

WAVE to wire models (W2W) rely on linear potential flow (LPF) assumptions for computing the hydrodynamic response of wave energy converters (WECs). The W2W models are then used to make predictions of motion, loads and power production, underpinning the estimated levelized cost of energy (LCoE). A restraint of the W2W models is that they must be computationally efficient as many sea states must be computed within a reasonable time. This restraint effectively rules out higher-fidelity CFD models and is the main reason LPF models are used regardless of the restrictions of their underlying assumptions of small wave and response amplitudes. Thus there is still a great interest in enhancing the performance of LPF models.

Machine learning (ML) is getting more and more attention in the ocean energy sector, although data-driven approaches have been used for some time within system identification and control of WECs [1]–[4].

The system identification approach is well described in [1], [2]. The papers detail how training data is to be constructed, and compare linear and nonlinear identification systems. Nonlinear autoregressive with exogenous input (NARX) models and artificial neural networks (ANN) were used to construct maps of force-to-body-positions. It was shown that while the ANN model always showed the best performance during the training phase, it performed worse than NARX during the validation phase.

Many optimal control strategies require forecasting of wave excitation forces acting on WECs to obtain optimal power output. A recent study [4] employed three different ANNs to forecast the excitation force, concluding that all three models gave satisfactory results with only minor differences between them. This provides a promising indication: that the quality of the forecast is insensitive to the choice of ANN. Also, [3] developed an ANN model to forecast the wave excitation force, in 3 degrees of freedom. The model performed well, with a goodness of fit larger than 92 % in all of the 12 sea states that were tested.

While the approach in this paper draws heavily on the work on system identification and excitation force forecasting, the aim here is not to achieve a mapping for control. Instead, we use ML to alleviate some of the shortcomings of the LPF models, aiming to achieve an augmented hybrid LPF-ML model that is to be used as any LPF model today. The hybrid LPF-ML approach has the potential to be a computationally efficient alternative to nonlinear Froude-Krylov force implementations, as well as a more versatile and accurate alternative to standard Morison type drag. Other sources of error, such as the linearised forces from wave radiation and diffraction of the wave field are implicitly compensated in the formulation. ML models provide general mapping functions, which makes them very suitable for representing such nonlinear correction terms while keeping the computational overhead

Support for this work was given by the Swedish Energy Agency through Grant No. 50196-1.

C. Eskilsson is with the Hydrodynamic Research Unit, RISE – Research Institutes of Sweden, P.O. Box 24001, SE-400 22 Gothenburg, Sweden (e-mail: claes.eskilsson@ri.se); and the Department of the Built Environment, Aalborg University, Thomas Manns Vej 23, DK-9220 Aalborg Ø, Denmark (e-mail: claese@build.aau.dk).

S. Pashami is with the Data Analysis Unit, RISE – Research Institutes of Sweden, Box 857, SE-501 15 Borås, Sweden (e-mail: sepideh.pashami@ri.se).

A. Holst is with the Data Analysis Unit, RISE – Research Institutes of Sweden, Box 857, SE-501 15 Borås, Sweden (e-mail: anders.holst@ri.se).

J. Palm is with Sigma Energy & Marine AB, Ekelundsgatan 1-3, SE-411 18 Gothenburg, Sweden (e-mail: johannes.palm@sigma.se).

Digital Object Identifier:
https://doi.org/10.36688/ewtec-2023-321

within acceptable limits.

Recently the authors [5] presented the first results of the hybrid LPF-ML model. By comparing the forces obtained from hierarchical numerical modelling (RANS, Euler, fully nonlinear potential flow and LPF) acting on a floating body, several nonlinear correction terms were defined. Recurrent neural networks (RNNs) were trained to predict the nonlinear corrections. The nonlinear correction forces were then included in the LPF model. The idea behind the subdivision of the correction terms was to link each correction to specific assumptions and limitations in the model theory. It also provided a correct and intuitive way to map results between model and prototype scale, as the viscous corrections are predominantly Reynolds-scaled and the pressure terms follow the Froude scaling law. Unfortunately, the hierarchical training approach was shown to be highly sensitive to numerical errors, given that these networks were very hard to train.

In the present work we thus examine a simpler approach: to compute a single step correction force using just RANS and LPF modelling. Additionally, we employ a more generic input signal (up-chirp) as suggested in [1], [2] and examine the performance of the hybrid LPF-ML model for a succession of more advanced test cases: decay test, regular wave case, irregular wave and irregular wave with a negative spring set-up.

II. Numerical models

Two models of different fidelity are tested in this paper: a high-fidelity, viscous Reynolds Averaged Navier-Stokes (RANS) model; and a model based on the standard linear potential flow (LPF) equations.

The RANS model uses the volume of fluid method (VOF-RANS) to compute the two-phase air-water interface. The model uses the `interFoam` solver, being a part of the open-source finite volume framework OpenFOAM [6], [7]. OpenFOAM is based on a cell-centered 2nd order finite volume method on unstructured polyhedral cells.

For the LPF simulations we use the MoodyMarine model [8], [9]. MoodyMarine solves the Cummins equation using pre-computed hydrodynamic coefficients.

A. Wave-body interaction

In CFD the resulting pressure force $\vec{F}_p$ and viscous force $\vec{F}_\nu$ acting on the body are given by directly integrating the total pressure p and the viscous shear stress τ over the body surface. For a body with total area $A = \sum_{i=1}^{N_c} A_i$, discretized into N_c cells, the equation of motion around the centre of gravity is described as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{M}\dot{\vec{X}}\right) = \vec{F}_p + \vec{F}_\nu + \mathbf{M}\begin{bmatrix}\vec{g}\\ 0\end{bmatrix}, \tag{1}$$

$$\vec{F}_p = -\sum_{i=1}^{N_c} A_i \begin{bmatrix} p_i\hat{n}_i \\ r_i \times (p_i\hat{n}_i)\end{bmatrix}, \tag{2}$$

$$\vec{F}_\nu = \sum_{i=1}^{N_c} A_i \begin{bmatrix} \vec{\tau}_i \\ r_i \times \vec{\tau}_i\end{bmatrix}, \tag{3}$$

where $\mathbf{M}$ is the mass matrix and $\vec{X}$ is the position of the floating body. Additionally r_i is the position vector from the centre of gravity to the cell face-centre and $\hat{\mathrm{n}}_i$ is the unit outward-pointing normal of face i.

MoodyMarine solves the Cummins equation [10] as follows:

$$\mathbf{M}\ddot{\vec{X}} = \vec{F}_a + \vec{F}_c + \vec{F}_r + \vec{F}_e + \vec{F}_{ext}\,, \tag{4}$$

$$\vec{F}_a = -\mathbf{M}_\infty \ddot{\vec{X}}\,, \tag{5}$$

$$\vec{F}_c = -\mathbf{C}\vec{X}\,, \tag{6}$$

$$\vec{F}_r = -\int_0^{T_{\mathrm{irf}}} \mathbf{K}(\tau)\,\dot{\vec{X}}\,(t-\tau)\,\mathrm{d}\tau, \tag{7}$$

where $\mathbf{M}_\infty$ is the (constant) added mass matrix at infinite frequency, and subsequently $\vec{F}_a$ is the added mass force. $\vec{F}_e$ is the hydrodynamic excitation force and $\vec{F}_c$ is the stiffness force, with the linear stiffness matrix $\mathbf{C}$. The radiation damping force is denoted $\vec{F}_r$ where $\mathbf{K}(t)$ is the impulse response function. Connected objects such as PTOs, mooring lines or applied loads are jointly denoted $\vec{F}_{ext}$.

III. Hybrid LPF-ML model

It is standard practice to include ad hoc corrections into the Cummins equation in order to improve the accuracy of model predictions. The most widespread example is the Morison-type drag force, which requires a calibration stage to validate the choice of drag coefficients. In the present work however, we choose instead to use a hybrid LPF-ML approach, in which the corrections are forecasted by a RNN. Adding a correction term $\vec{\delta}$ to the Cummins equation yields

$$\mathbf{M}\ddot{\vec{X}} = \vec{F}_p^{LPF} + \vec{F}_{ext} + \vec{\delta}\,, \tag{8}$$

$$\vec{F}_p{}^{LPF} = \vec{F}_a + \vec{F}_c + \vec{F}_r + \vec{F}_e\,. \tag{9}$$

A. Correction force

The RANS simulations provide results in terms of two generalised forces: the pressure force $\vec{F}_p$, and the viscous force $\vec{F}_\nu$. As seen from Eq. (9), LPF include forces from added inertia at infinite frequency $\vec{F}_a$, radiation damping $\vec{F}_r$, hydrodynamic excitation $\vec{F}_e$, and hydrostatic stiffness $\vec{F}_c$. We define the following correction force:

$$\vec{\delta} = \vec{F}_p{}^{RANS} + \vec{F}_\nu{}^{RANS} - \vec{F}_p{}^{FNPF}\,. \tag{10}$$

We note that the previous studies [11] and [5] employed a hierarchical modelling approach using different fidelity models. As mentioned above, that approach gives rise to more fine grained correction forces that can be linked to physical assumptions and simplifications in the LPF theory. Additionally, proper scaling of the different terms is possible [12]. However, as illustrated in [5] the influence of numerical errors makes that route problematic in practice.

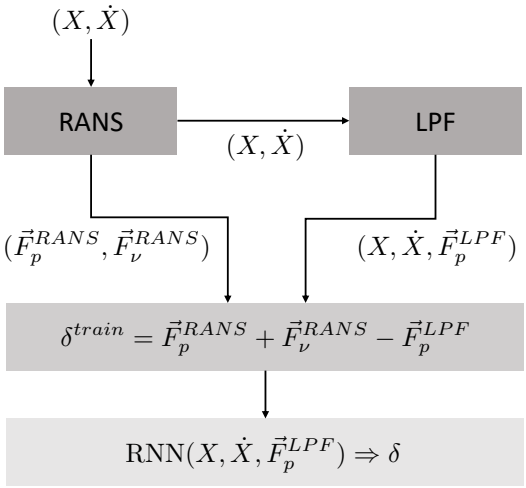

Fig. 1. Outline of the training of the RNN for correction forces.

B. RNNs

RNNs are designed to process sequential data by learning their inter-dependencies in varying lengths. Long short-term memory (LSTM) network [13] is a special kind of RNN that avoids the so-called vanishing gradient problem in RNNs for long-term dependencies [14]. LSTM has more control over which information should be kept and which should be forgotten through its internal memory nodes, and is widely used in many fields. In this work we use a standard uni-directional LSTM network.

Matlab's Deep Learning Toolbox [15] is used to create and train the LSTM network estimating the correction terms. For the present application the LSTM network is made up of a sequence input layer with six features; an LSTM layer with different sizes of hidden units; a fully connected layer, and finally a regression layer. The LSTM layer is stateless at training and uses the default settings of tanh as a state activation function and sigmoid as a gate activation function. We also test using two and three LSTM layers with dropout layers (with dropout rate 0.2) in between the LSTM layers to avoid overfitting. We will compare the results with different structures in the result section.

The adaptive moment estimation (adam) optimizer [16] is used to train the LSTM. Learning rates of 0.01–0.001 and up to 200 epochs are used depending on the time steps, layers and hidden units. The input data is z-score normalized.

C. Hybrid LPF-ML implementation

The performance of the hybrid LPF-ML model is highly dependent on the quality of the data used for training. Fig. 1 outlines the flow of data during the training. The high-fidelity RANS model is executed first: either as free floating, or with a prescribed motion or external force. The CFD model provides results of $\vec{F}_p^{RANS}$ and $\vec{F}_\nu^{RANS}$ as well as the motion $(X, \dot{X}$. Then, the LPF model is run with the prescribed motion from the CFD model. This is a vital step in order to minimize errors and achieve a representative one-to-one mapping between a given body state and its corresponding force correction. As the full time-history of the body motion is matched, the methodology also handles influences from transient effects in the two models. The output data from the LPF simulations are the four forces making up $\vec{F}_p^{LPF}$, as defined in Eq. (9). The LSTM network is then trained with six features of input data $(X, \dot{X}, \vec{F}_a, \vec{F}_c, \vec{F}_r, \vec{F}_e)$ against the correction force δ (using the Matlab command `trainNetwork`).

```
Initialize MoodyMarine;
Open portal to MATLAB and initialize workspace;
Load trained RNNs into MATLAB workspace;
while t ≤ t_end do
    Evaluate F_p^LPF;
    Transfer (X, Ẋ, F_p^LPF) to MATLAB workspace;
    Evaluate the correction force, δ, and update the
    RNN;
    Transfer δ from MATLAB workspace;
    Update the position and velocity by solving Eq. (8);
end
Close portal to MATLAB;
```

Fig. 2. Pseudo code for the hybrid LPF-ML set-up.

After the LSTM network has been trained, we use it in the hybrid LPF-ML discussed above. The MoodyMarine LPF model [9] was linked to Matlab through the Engine API for C for this particular purpose. The MoodyMarine-Matlab coupling is outlined in Fig. 2. When MoodyMarine is started it creates a portal to a Matlab workspace. We then load the trained LSTM into the workspace before the MoodyMarine simulation is started, and reset the network (using the Matlab command `resetState`). In every time-step we update the six input features and transfer them into the Matlab workspace. There the input are used by the LSTM network to produce correction force estimates and to update the state of the network (using the matlab command `predictAndUpdateState`). The correction force goes back into MoodyMarine and is used to advance the body motion in time (8).

D. Input signals

Ringwood and co-workers have, for the last decade, looked into the performance of different input signals for system identification, e.g. [1], [2], [17], [18]. In a recent study [19], three families of input signals were investigated with regard to force-to-velocity input for design of controllers: sinusoidal signals, chirp signals, and multi-sine signals. The chirp and multi-sine signals were found to show large variability and uncertainty. This is due to the relatively short time spent at each frequency. The sinusoidal signal was smooth with low variability, but many sinusoidal signals are required to span a range of frequencies. To maintain the feasibility of the hybrid LPF-ML approach, we opted to use a few multi-frequency cases as training data rather than a large number of single frequency signals. Thus, the standard linear up-chirp signal is used for position-to-force mapping. Additionally, in order to capture any geometric nonlinearity of a body we will use a sequence of chirps with different amplitudes.

TABLE I
SPHERE DATA. FROM [20].

Parameter	Unit	Value
Diameter (D)	m	0.300
Mass (m)	kg	7.056
Center of gravity (CoG)	m	(0, 0, -0.0348)
Roll moment of inertia (I_{xx})	kgm^2	0.098251
Pitch moment of inertia (I_{yy})	kgm^2	0.098254
Yaw moment of inertia (I_{zz})	kgm^2	0.073052
Water depth (d)	m	0.9
Acceleration of gravity (g)	ms^{-2}	9.82
Water density (ρ_w)	kgm^{-3}	998.2

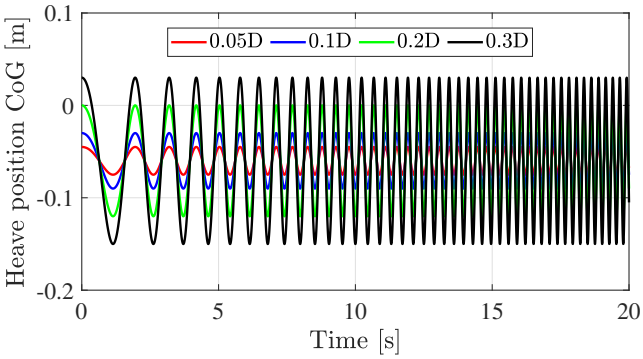

Fig. 3. Linear up-chirps with different amplitudes.

IV. TEST CASES

We examine the 1 degree-of-freedom (1DoF) problem of a heaving sphere. This is a case that has been used repeatedly within the Ocean Energy System wave energy modelling task (OES-WEMT) [20]–[22]. The heaving sphere was first simulated in full-scale (with a sphere diameter of D=10 m) using numerical models [21], [22], and was later experimentally tested in model-scale (D=0.3 m) by Kramer *et al.* [20]. The simulations in this paper have been made in model scale, see Table I for details.

Four different test cases are investigated following the simulations in [21]:

- Decay tests. Three drop heights (H_0) were considered in the experiments [20]: $H_0 = [0.1D, 0.3D, 0.5D]$, with the sphere just being lifted out of the water for the $0.5D$ case. The duration of the simulations is 3 s.
- Regular waves. An incident Stokes V wave at the resonance period $T_{e0} = 0.76$ s with wave height $H = 0.0584$ m and no PTO damping. The duration of simulations is 15 s.
- Irregular waves. An irregular sea state is given by a Pierson-Moskovitz (PM) spectra with a significant wave height $H_s = 0.060$ m and peak period $T_p = 1.30$ s. The models are run for 100 s, equivalent to 10 minutes in full-scale.
- Irregular waves with negative spring setting. Settings as in the previous case but with a negative spring PTO (spring coefficient -4.05E02 N/m and damping 11.75 Ns/m).

A. Training of LSTM

Fig. 3 shows the chirp signals used for training. The chirp signals have a duration of 20 s and a frequency range of $f \in [0.33, 4]$ Hz. The 4 signals have amplitudes of $a = [0.05, 0.1, 0.2, 0.3]D$ m respectively. We use all 4 chirps for training since there will be no data overlap as we validate against the decay test.

The four cases were modelled with `interFOAM` using mesh morphing in OpenFOAM-v2212. No symmetries were used and we used a mesh of approximately 10M cells. For spatial discretization we use second-order van Leer scheme for convection terms, second-order central differences for diffusion terms, and first-order upwind method for the turbulence equations. The time-stepping is carried out using the first-order backward Euler scheme with a CFL number of 0.5.

With regard to the setup of the LSTM network, we test to use $[25, 50, 75, 100, 150]$ hidden units in the LSTM layers, as well as employing one, two and three LSTM layers. We also test the time resolution of the training data, given by the number of time steps $N = [2000, 5000, 20000]$ in the data.

V. RESULTS

The results are presented as time-series as well as in terms of normalized root mean square error (NRMSE) and the Pearson's correlation coefficient (PCC).

The NRMSE represents the square root of the square of the differences between target values and output values and is defined as:

$$\mathrm{NRMSE} = \frac{\sqrt{\sum_{i=1}^{n} (T_i - O_i)^2}}{\sqrt{\sum_{i=1}^{n} (T_i)^2}}, \tag{11}$$

where n is the total number of instances of the studied data set, T_i is the target value and O_i is the output value.

The PCC is a measure of linear correlation between two sets of data and is defined as

$$\mathrm{PCC} = \frac{\sum_{i=1}^{n} (T_i - \bar{T})(O_i - \bar{O})}{\sqrt{\sum_{i=1}^{n} (T_i - \bar{T})^2} \sqrt{\sum_{i=1}^{n} (O_i - \bar{O})^2}}, \tag{12}$$

where $\bar{T}$ and $\bar{O}$ are the mean values of the target and output values, respectively.

A. Sphere heave decay

As a first validation stage we compare with a straightforward sphere decay test. The very same decay test was investigated in Eskilsson *et al.* [5] using a selected decay test for training the LSTM network. In [11] it was found that using the most nonlinear decay case for training typically gave satisfactory results, so the $0.5D$ drop was used as training data. We will compare the chirp-trained LSTM network to this decay trained LSTM network later.

In this work we still test on the decay dataset, but training is done on the four chirp signals, which exhibit very different multi-frequency properties, and highly different transients in the resulting signals. Needless

TABLE II
NRMSE AND PCC OF CORRECTION FORCES FOR THE DECAY TEST USING DIFFERENT LSTM SETTINGS AND TIME RESOLUTION OF THE TRAINING DATA.

Training signals	Units and layers	N	$0.1D$		$0.3D$		$0.5D$	
			NRMSE	PCC	NRMSE	PCC	NRMSE	PCC
	50	2000	3.42	0.06	1.37	0.36	0.95	0.49
	50	5000	1.15	0.26	0.97	0.63	0.84	0.63
	50	20000	0.84	0.42	0.99	0.59	0.88	0.59
	75	2000	2.01	-0.40	1.28	0.41	1.08	0.44
	75	5000	1.16	0.10	0.91	0.62	0.85	0.64
	75	20000	0.76	0.53	0.95	0.61	0.85	0.58
	100	2000	2.07	-0.28	1.51	0.49	1.03	0.49
	100	5000	0.70	0.61	0.74	0.78	0.55	0.84
	100	20000	0.97	0.30	1.00	0.59	0.86	0.63
	150	2000	1.81	-0.25	1.09	0.58	0.87	0.58
	150	5000	1.52	-0.19	0.92	0.65	0.82	0.65
Chirps	150	20000	0.96	0.19	0.87	0.62	0.85	0.60
	25/25	2000	1.13	-0.25	1.20	0.38	0.92	0.47
	25/25	5000	1.07	0.09	1.16	0.42	1.16	0.50
	25/25	20000	1.22	0.41	0.95	0.54	0.88	0.52
	50/50	2000	1.78	-0.12	1.53	0.34	1.00	0.48
	50/50	5000	0.94	0.09	1.00	0.50	0.86	0.62
	50/50	20000	1.12	0.51	0.96	0.60	0.84	0.60
	25/25/25	2000	1.37	-0.20	1.44	0.16	1.23	0.25
	25/25/25	5000	0.89	0.35	1.08	0.41	1.00	0.43
	25/25/25	20000	0.80	0.49	0.97	0.56	0.82	0.64
	50/50/50	2000	1.36	-0.28	1.33	0.34	1.13	0.38
	50/50/50	5000	1.21	-0.25	1.13	0.43	0.93	0.50
	50/50/50	20000	0.88	0.32	0.78	0.66	0.82	0.63

to say, this is a strenuous test case for the LSTM performance.

Table II presents the NRMSE and PCC values for an array of different hidden units and layers of the LSTM, as well as for the length of the in-going time series N. Fig. 4 illustrate some of the forecasted δ time series.

A bit surprising is that the $0.1D$ case is the hardest case to forecast with accuracy, the $0.1D$ case has the largest NRMSE values as well as variability. We also see that there is no immediate benefit to using a large number of hidden units in the LSTM layers. Neither gives multi-layer structures an immediate advantage in terms of low NRMSE values. The largest and deepest structure of neural networks better follows the target pattern in the case of $0.1D$. However, it clearly loose smooth prediction, which can be a sign of overfitting.

However, we can see a clear result with regard to the time resolution N. For the $0.1D$ case the $N = 2000$ case gives very bad results (Fig. 4a-d). All PCC values, but one, are negative, meaning that there is a negative correlation between the target and forecast values. For the three-layer structure there is a consistent trend of improved PCC and NRMSE values with increasing N. Even if this trend is not absolute we see a general improvement with increased time resolution.

The best results are given by a single LSTM layer with 100 units and $N = 5000$ (Fig. 4f). This result seems however to be due to chance rather than any trends. The three-layer network with $N = 20000$ also shows good results, and these are following the trend with N as mentioned above. Nevertheless, in the remainder of this work we use the LSTM network with a single LSTM layer with 100 units and trained with $N = 5000$ time steps.

In Fig. 5 we present a comparison of the forecasted correction forces using decay signals and chirp signals as training data. For the decay training, the LSTM has been trained on the $0.5D$ drop case, which is why it is marked red in Fig. 5a. The chirp training data includes the $0.05D$, $0.1D$, $0.2D$ and $0.3D$ amplitudes. It can be seen that the $0.1D$ drop is better predicted using the chirp signal. This is likely due to more small amplitude data in the chirps compared to training on $0.5D$ drop. For the $0.3D$ case the decay trained LSTM provides significantly better than the chirp trained. This is, of course, due to the similarities between the $0.3D$ and $0.5D$ drops. Clearly, only the chirp trained LSTM is able to provide a forecast for the $0.5D$ case. These findings are supported by the NRMSE data found in Table II. We see that the decay trained $0.3D$ has the lowest recorded NRMSE, but we also see that the decay trained $0.1D$ case performs quite badly. In general, we can say that, as expected, decay training data performs better for decay test cases. However, free decay will not perform well for multi-frequency cases, e.g. irregular waves, as decay is basically a single frequency signal.

Moving away from the offline training and testing of the LSTM network towards hybrid LPF-ML simulations, we first verify that the trained LSTM network provides a forecast in the MoodyMarine-Matlab coupled model that is similar to the forecast in Matlab

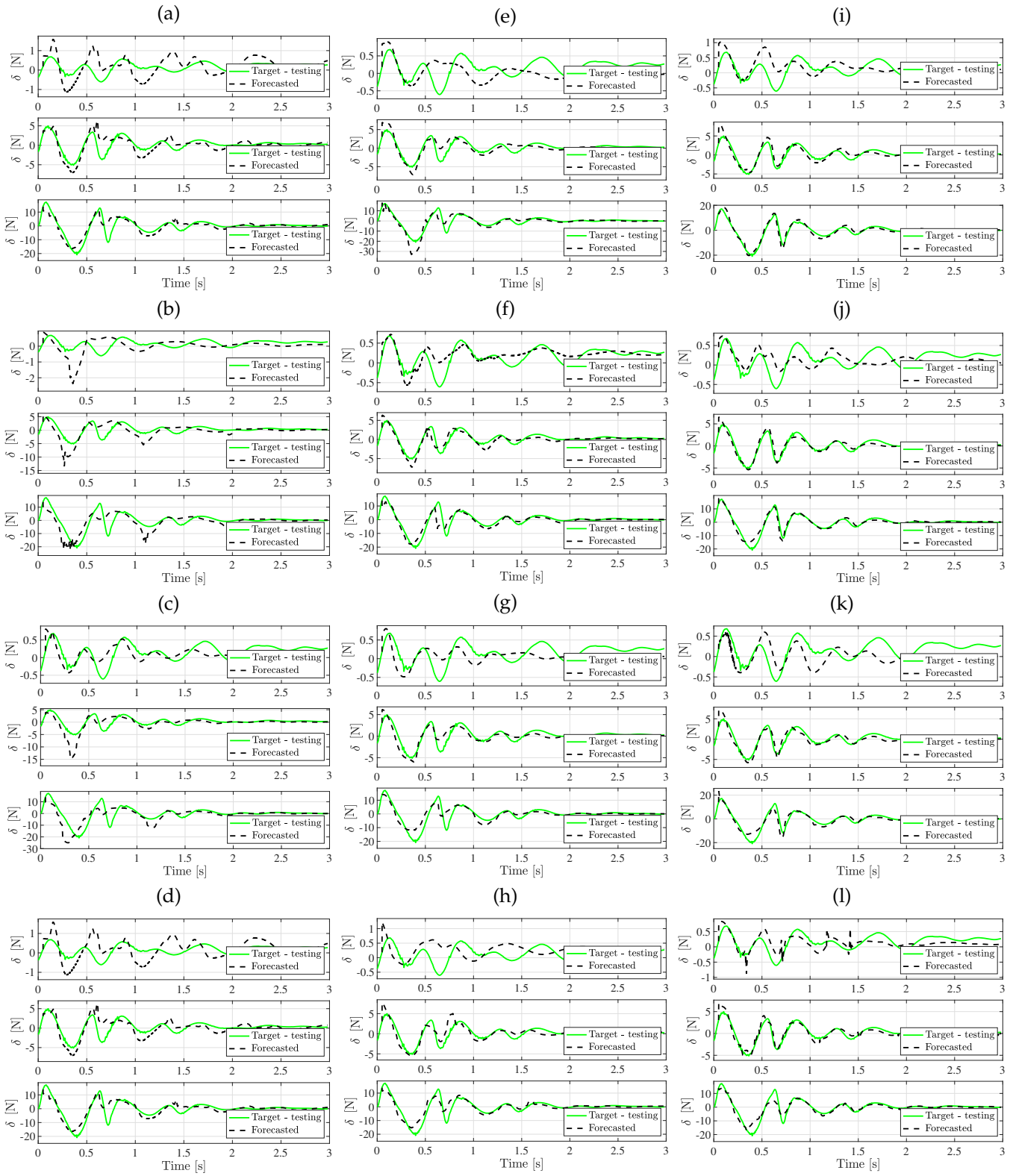

Fig. 4. Forecasted correction forces using different settings for the LSTM, top panels $0.1D$, middle panels $0.3D$ and bottom panels $0.5D$ drops cases. Left column: $N = 2000$, middle column: $N = 5000$, right column: $N = 20000$. Top row: 50 units, second from top row: 100 units, second from bottom: 2 layers with 25 units, bottom row: 2 layers with 50 units each.

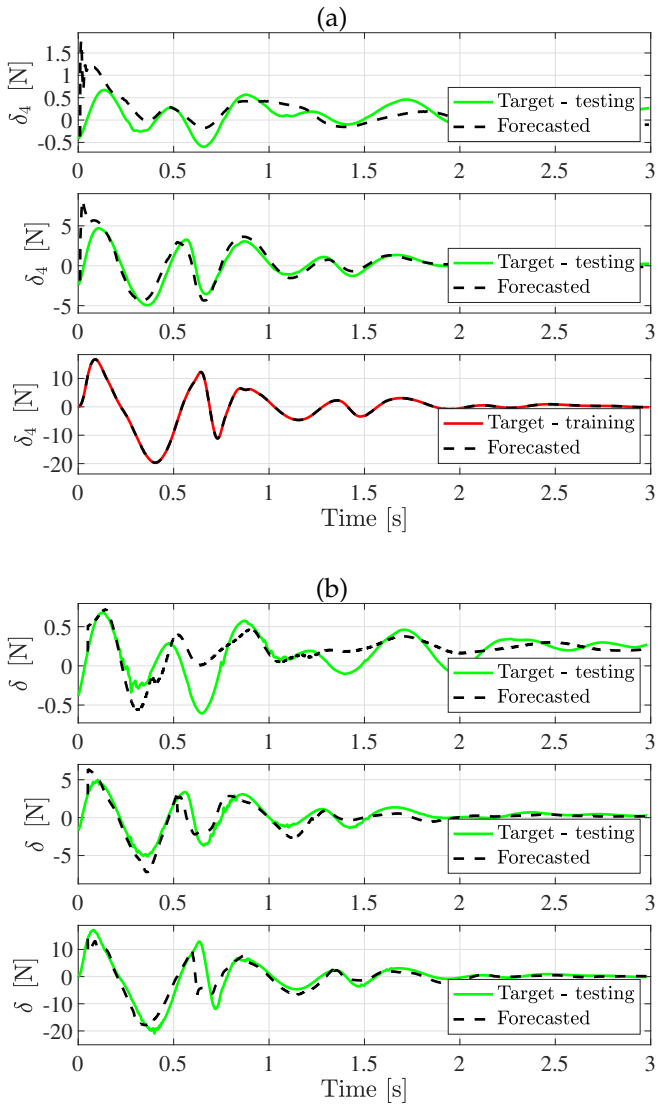

Fig. 5. Forecasted correction forces: top panels $0.1D$, middle panels $0.3D$ and bottom panels $0.5D$ drops cases. (a) decay test as training data [5] (LSTM with 100 hidden units and $N = 3000$ time steps) and (b) chirps as training data (LSTM with 100 hidden units and $N = 5000$ time steps).

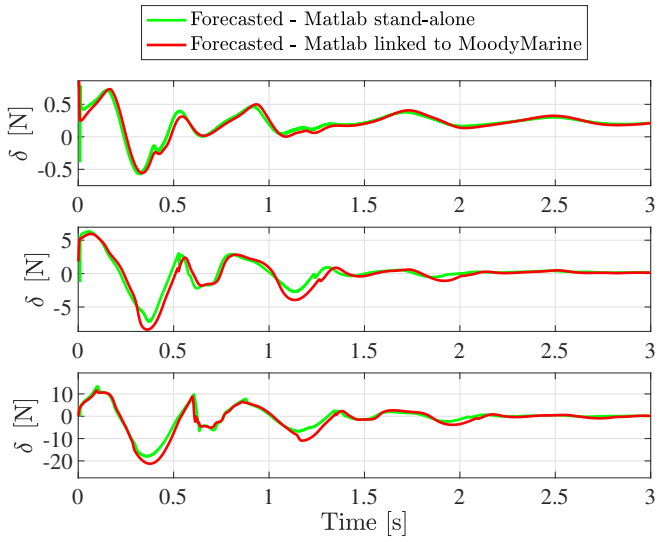

Fig. 6. Forecasted correction forces given by Matlab standalone and Matlab linked to MoodyMarine.

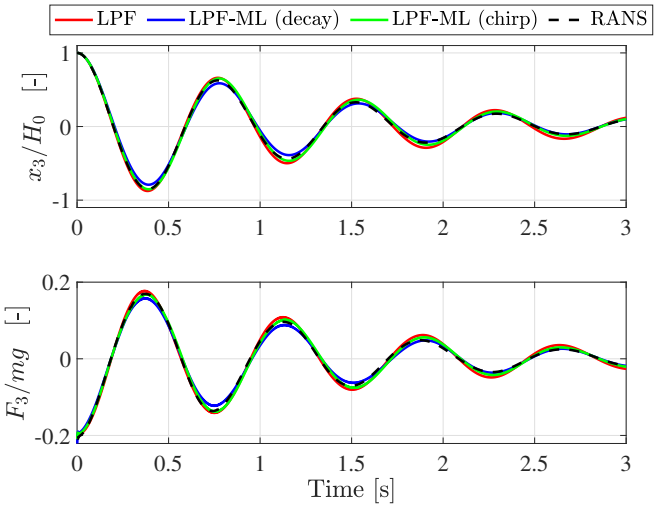

Fig. 7. Heave decay for $0.1D$ drop. Top panel: non-dimensional heave response, and bottom panel: non-dimensional heave force.

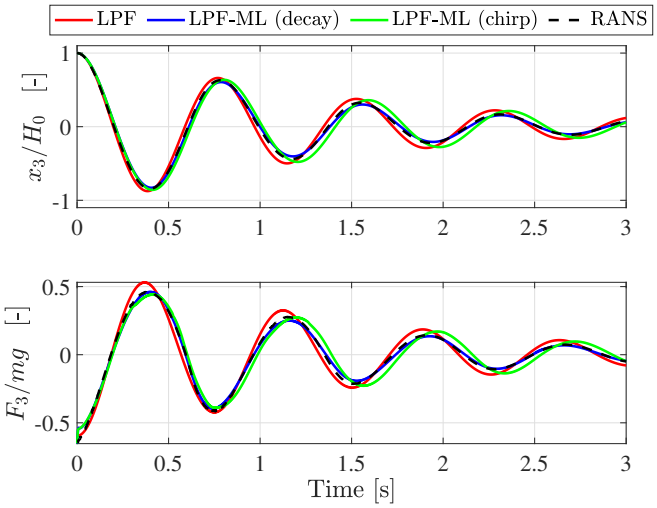

Fig. 8. Heave decay for $0.3D$ drop. Top panel: non-dimensional heave response. Bottom panel: non-dimensional heave force.

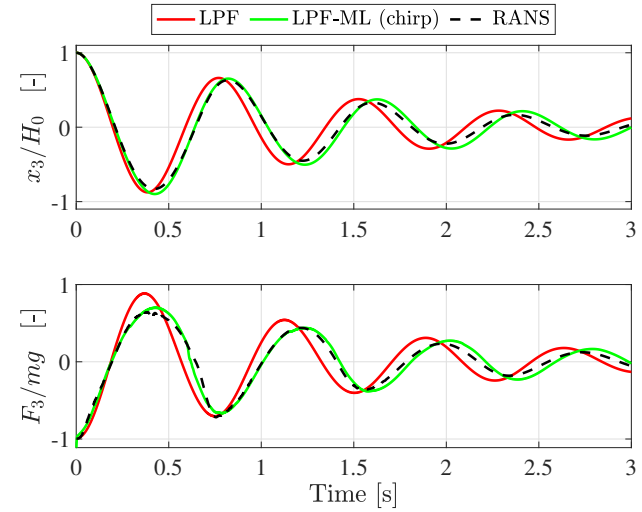

Fig. 9. Heave decay for $0.5D$ drop. Top panel: non-dimensional heave response. Bottom panel: non-dimensional heave force.

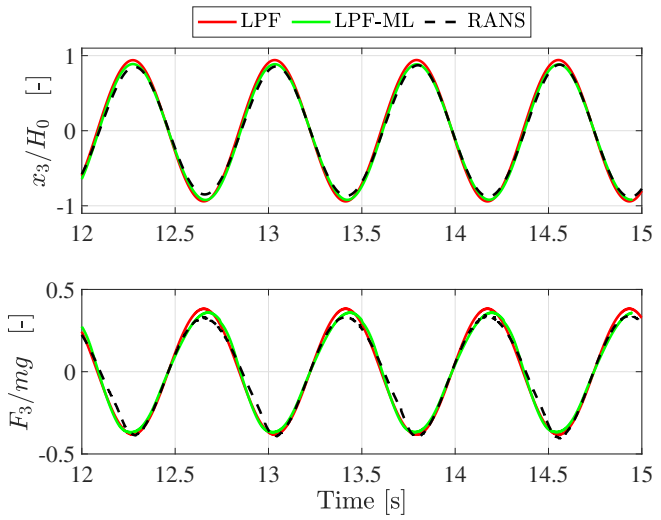

Fig. 10. Regular waves case. Top panel: non-dimensional heave response. Bottom panel: non-dimensional heave force.

standalone setting, see Fig. 6. We can see that the trained LSTM works well in the coupled setting. The small differences are simply due to force-motion feedback from the MoodyMarine model when applying external forces.

Figs. 7 – 9 show the computed heave responses and heave forces for the different drop heights. Please note that the LPF-ML forces include the correction δ.

The results of the $0.1D$ drop case in Fig. 7 show that both LPF-ML versions perform better than the standard LPF model. However, the chirp trained LPF-ML model outperforms the decay trained model. Indeed, the chirp trained model has a very good fit both in position and force for the first 2 s. During the last second of simulation, the chirp LPF-ML model overestimate the damping of the buoy. The results then become more similar to the decay trained LPF-ML.

For the $0.3D$ case the situation is reversed. The decay trained LPF-ML clearly outperforms the chirp trained. The decay trained model gives an almost perfect match to the RANS results. The chirp trained LPF-ML model performs better than the standard LPF model during the first half of the simulation. During the second half a significant phase error becomes evident.

For the $0.5D$ case there is a significant difference in response period between the LPF and RANS models, see Fig. 9. This was first noticed in [22] and was later confirmed by experimental results [20]. The hybrid LPF-ML model captures the correct response period very well. We see a slight phase error at the later stages of the simulation, but overall the results are very encouraging. Please remember that the LPF-ML model is not trained on decay tests, but on generic chirp multi-frequency signals. Still, the model manages well to compensate for the nonlinearities in the problem.

B. Heaving sphere in regular waves

We now step away from the decay tests and look at the regular wave case. We mention that even though we include the hydrodynamic excitation force $\vec{F}_e$ as an input feature, see Eq. (9), the training cases only involve still water. Thus, the LSTM is presently not trained to take wave excitation into account, which is a disadvantage.

Fig. 10 shows a comparison between the LPF, LPF-ML and RANS models. All models give similar heave response, although the LPF-ML model more closely mimics the RANS model. The advantage of the LPF-ML over LPF is more visible for the heave forces, where LPF overpredicts the forces. An additional issue that might cause differences between CFD and LPF models is that we have used an incident Stokes V:th order wave in the CFD while only Airy waves in the LPF models.

C. Heaving sphere in irregular waves

The final case deals with a heaving sphere in irregular waves, both freely floating and fitted with passive control through a negative spring and PTO damper. Here we do not compare to RANS solutions but only LPF vs LPF-ML, see Figs. 11 and 12. We see that

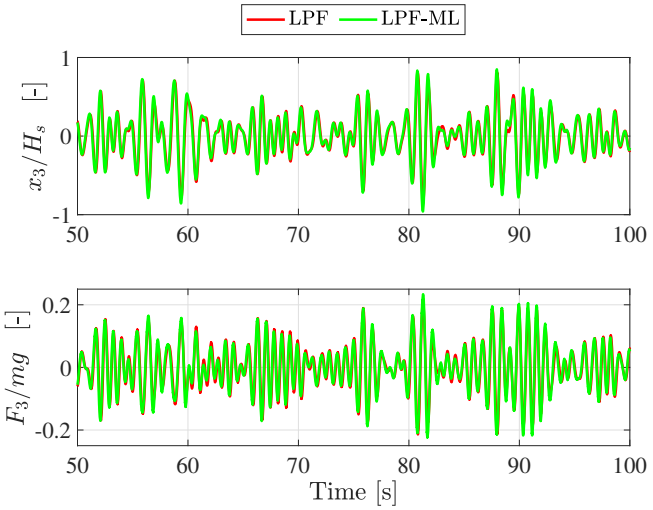

Fig. 11. Irregular waves case. Top panel: non-dimensional heave response, and bottom panel: non-dimensional heave force.

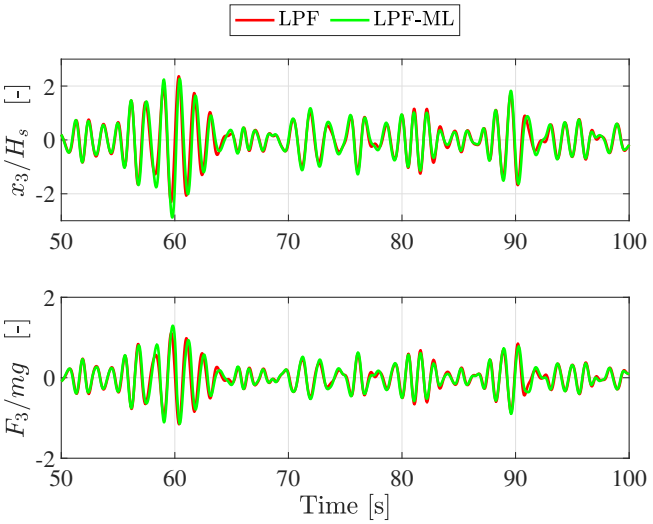

Fig. 12. Irregular waves case with negative spring. Top panel: non-dimensional heave response. Bottom panel: non-dimensional heave force.

the addition of negative spring increases the motion response by a factor of two and also decreases the frequency of the response. However, with regard to the influence of the nonlinear correction we can not see any. Neither in the freely floating case nor the negative spring case can we see any clear effects of including the LSTM corrector. This is surprising as the nonlinearity of the negative spring should be considerable. This requires further investigation and is ongoing work.

VI. Conclusion

Machine learning (ML) can be used to enhance the performance of linear potential flow (LPF) models, by giving better approximations to weakly nonlinear forces than parameterized formulas and tabulated coefficients. In this work we looked into to the performance of such a hybrid LPF-ML model. It was shown that we were able to capture the nonlinear corrections for large drop decay tests even when a very generic input signal (up-chirp) was used. This gives hope to find a standard setting for choosing input signals.

In addition, work is ongoing to decide on and include training cases with wave excitation. The results of this paper indicate that a sufficiently small data set can be found, which still provides enough nonlinear training data to span the nonlinearities captured by the LSTM network.

Acknowledgement

We gratefully acknowledge that the computations were performed on resources at (i) the National Supercomputer Centre provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) and (ii) LUMI through the Danish e-infrastructure Cooperation (DeiC) National HPC Grant Agreement No. DeiC-AAU-N3-2023017.

References

[1] J. Davidson, S. Giorgi, and J. Ringwood, "Identification of wave energy device models from numerical wave tank data – Part 1: Numerical wave tank identification tests," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 3, pp. 1012–19, 2016.

[2] S. Giorgi, J. Davidson, and J. Ringwood, "Identification of wave energy device models from numerical wave tank data - Part 2: Data-based model determination," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 3, pp. 1020–1027, 2016.

[3] M. Bon, F. Carapellese, S. Sirigu, G. Bracco, and G. Mattiazzo, "Excitation forces estimation for nonlinear wave energy converters: a neural network approach," *IFAC PapersOnLine 53-2*, pp. 12 334–12 339, 2020.

[4] K. Mahmoodi, E. Nepomuceno, and A. Razminia, "Wave excitation force forecasting using neural networks," *Energy*, p. 123322, 2022.

[5] C. Eskilsson, S. Pashami, A. Holst, and J. Palm, "Hierarchical approaches to train recurrent neural networks for wave-body interaction problems," in *Proceedings of the 33rd International Ocean and Polar Engineering Conference*, Ottawa, Canada, 2023.

[6] OpenFOAM, "http://www.openfoam.com," 2020.

[7] H. Weller, G. Tabor, H. Jasak, and C. Fureby, "A tensorial approach to computational continuum mechanics using object-oriented techniques," *Computers in Physics*, vol. 12, no. 6, pp. 620–631, 1998.

[8] J. Palm and C. Eskilsson, "On end-stops and snap loads for taut-moored wave energy converters," in *Proceedings from the 14th European Wave and Tidal Energy Conference (EWTEC)*, Plymouth, UK, 2021.

[9] ——, "Verification and validation of MoodyMarine: A free simulation tool for modelling moored MRE devices," in *Proceedings of the 15th European Wave and Tidal Energy Conference*, Bilbao, Spain, 2023.

[10] W. Cummins, "The impulse response function and ship motions," *Schiffstechnik*, vol. 9, pp. 101–109, 1962.

[11] C. Eskilsson, S. Pashami, A. Holst, and J. Palm, "Estimation of nonlinear forces acting on floating bodies using machine learning," in *Proceedings of the 9th International Conference on Marine Structures*, Gothenburg, Sweden, 2023.

[12] J. Palm, C. Eskilsson, L. Bergdahl, and R. Bensow, "Assessment of scale effects, viscous forces and induced drag on a point-absorbing wave energy converter by cfd simulations," *Journal for Marine Science and Engineering*, vol. 6, no. 4, p. 124, 2018.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[15] Mathworks, "https://se.mathworks.com/help/deeplearning/," 2022.

[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2014.

[17] J. Davidson, C. Windt, G. Giorgi, R. Genest, and J. V. Ringwood, "Evaluation of energy maximising control systems for wave energy converters using OpenFOAM," in *OpenFOAM - Selected papers from the 11th workshop*, M. Nobrega and H. Jasak, Eds. Springer, 2018.

[18] G. Giorgi and J. Ringwood, "Consistency of viscous drag identification tests for wave energy applications," in *Proceedings from the 12th European Wave and Tidal Energy Conference (EWTEC)*, Cork, Ireland, 2017.

[19] M. Farajvand, V. Grazioso, D. García-Violini, and J. V. Ringwood, "Uncertainty estimation in wave energy systems with applications in robust energy maximising control," *Renewable Energy*, vol. 203, pp. 194–204, 2023.

[20] M. B. Kramer, J. Andersen, S. Thomas, F. B. Bendixen, H. Bingham, R. Read, N. Holk, E. Ransley, S. Brown, Y.-H. Yu, T. T. Tran, J. Davidson, C. Horvath, C.-E. Janson, K. Nielsen, and C. Eskilsson, "Highly accurate experimental heave decay tests with a floating sphere: A public benchmark dataset for model validation of fluid–structure interaction," *Energies*, vol. 14, no. 2, p. 269, 2021. [Online]. Available: https://www.mdpi.com/1996-1073/14/2/269

[21] F. Wendt, Y.-H. Yu, K. Nielsen, K. Ruehl, T. Bunnik, I. Touzon, B. W. Nam, J. S. Kim, K.-H. Kim, C. E. Janson, K.-R. Jakobsen, S. Crowley, L. Vega, K. Rajagopalan, T. Mathai, D. Greaves, E. Ransley, P. Lamont-Kane, W. Sheng, R. Costello, B. Kennedy, S. Thomas, P. Heras, H. Bingham, A. Kurniawan, M. M. Kramer, D. Ogden, S. Girardin, A. Babarit, P.-Y. Wuillaume, D. Steinke, A. Roy, S. Beatty, P. Schofield, J. Jansson, and J. Hoffman, "International energy agency ocean energy systems task 10 wave energy converter modeling verification and validation," in *Proceedings of the 12th European Wave and Tidal Energy Conference*, Cork, Ireland, 2017.

[22] F. Wendt, K. Nielsen, Y.-H. Yu, H. Bingham, C. Eskilsson, M. Kramer, A. Babarit, T. Bunnik, R. Costello, S. Crowley, B. Gendron, G. Giorgi, S. Giorgi, S. Girardin, D. Greaves, P. Heras, J. Hoffman, H. Islam, K.-R. Jakobsen, C.-E. Janson, J. Jansson, H. Y. Kim, J.-S. Kim, K.-H. Kim, A. Kurniawan, M. Leoni, T. Mathai, B.-W. Nam, S. Park, K. Rajagopalan, E. Ransley, R. Read, J. V. Ringwood, J. M. Rodrigues, B. Rosenthal, A. Roy, K. Ruehl, P. Schofield, W. Sheng, A. Shiri, S. Thomas, I. Touzon, and I. Yasutaka, "Ocean energy systems wave energy modelling task: Modelling, verification and validation of wave energy converters," *Journal of Marine Science and Engineering*, vol. 7, no. 11, 2019. [Online]. Available: https://www.mdpi.com/2077-1312/7/11/379