Journal of Marine Science and Engineering

MDPI

Article

Towards Real-Time Reinforcement Learning Control of a Wave Energy Converter

Enrico Anderlini [1,*], Salman Husain [2], Gordon G. Parker [2], Mohammad Abusara [3] and Giles Thomas [1]

1 Department of Mechanical Engineering, University College London, London WC1E 6BT, UK; giles.thomas@ucl.ac.uk

2 Department of Mechanical Engineering—Engineering Mechanics, Michigan Technological University, Houghton, MI 49931, USA; shusain@mtu.edu (S.H.); ggparker@mtu.edu (G.G.P.)

3 College of Engineering, Mathematics and Physical Sciences, University of Exeter, Penryn Campus, Penryn, Cornwall TR10 9FE, UK; M.Abusara@exeter.ac.uk

* Correspondence: e.anderlini@ucl.ac.uk

Received: 30 September 2020; Accepted: 23 October 2020; Published: 28 October 2020

Abstract: The levellised cost of energy of wave energy converters (WECs) is not competitive with fossil fuel-powered stations yet. To improve the feasibility of wave energy, it is necessary to develop effective control strategies that maximise energy absorption in mild sea states, whilst limiting motions in high waves. Due to their model-based nature, state-of-the-art control schemes struggle to deal with model uncertainties, adapt to changes in the system dynamics with time, and provide real-time centralised control for large arrays of WECs. Here, an alternative solution is introduced to address these challenges, applying deep reinforcement learning (DRL) to the control of WECs for the first time. A DRL agent is initialised from data collected in multiple sea states under linear model predictive control in a linear simulation environment. The agent outperforms model predictive control for high wave heights and periods, but suffers close to the resonant period of the WEC. The computational cost at deployment time of DRL is also much lower by diverting the computational effort from deployment time to training. This provides confidence in the application of DRL to large arrays of WECs, enabling economies of scale. Additionally, model-free reinforcement learning can autonomously adapt to changes in the system dynamics, enabling fault-tolerant control.

Keywords: wave energy converter; control; reinforcement learning; deep reinforcement learning; deep learning; adaptive control

1. Introduction

Ocean wave energy is a type of renewable energy with the potential to contribute significantly to the future energy mix. Despite an estimated global resource of 146 TWh/yr [1], the wave energy industry is still in its infancy. In 2014, there were less than 10 MW of installed capacity worldwide due to the high levellised cost of energy (LCoE) of approximately EUR 330–630/MWh [1]. The main operational challenge is the maximisation of energy extraction in the common, low-energetic sea states, whilst ensuring the survival of the wave energy converters (WECs) in storms [2]. Two major contributors to the lowering of the LCoE to EUR 150/MWh by 2030 are expected to be the achievement of economies of scale and the development of effective control strategies [3].

Over the past decade, model predictive control (MPC) has attracted much research interest, as it can offer improved performance over the control strategies developed in the 1970s and 1980s, based on hydrodynamic principles. Assuming knowledge of the wave excitation force, MPC computes the control action, typically the force applied by the power take-off system (PTO), that results in optimal

energy absorption over a future time horizon using a model of the WEC dynamics. The controller applies only the first value of the PTO force, recomputing the optimal control action at the next time horizon. The iterative procedure enables the controller to reduce the negative impact of inaccuracies in the prediction of the future excitation force and modelling errors. Additionally, the MPC framework enables the inclusion of constraints on both the control action and the system dynamics. A good review of MPC for WEC control can be found in [4]. Li and Belmont first proposed a fully convex implementation, which trades off the energy absorption, the energy consumed by the actuator and safe operation [5]. An even more efficient implementation cast in a quadratic programming form has been proposed by Zhong and Yeung [6]. Fundamentally, the convex form enables the strategy to be generalised to the control of multiple WECs in real-time [7,8]. However, linear MPC relies on a linear model of the WEC dynamics. In energetic waves, nonlinearities in the static and dynamic Froude–Krylov forces (i.e., the hydrostatic restoring and wave incidence forces) and viscous drag effects can become significant [9]. Although nonlinear MPC strategies have been proposed [10,11] and even tested experimentally [12], achieving a successful real-time, centralised control implementation for multiple WECs is expected to be challenging [13].

As described in [14,15], alternative strategies have been developed for the control of WECs. Some, like simple-and-effective control [16], present similar performance to MPC at a much lower computational cost. Alternative solutions based on machine learning have been recently considered thanks to the advancements in the field of artificial intelligence. Most commonly, the neural networks are used to provide a data-based, nonlinear model of the system dynamics, i.e., for system identification. After training, the identified model is coupled with standard strategies used for the control of WECs, e.g., resistive or damping control in [17], reactive or impedance-matching in [18] and latching control in [19,20]. On the one hand, some studies have proposed the use of neural networks to find the optimal parameters for impedance-matching control on a time-averaged basis [21], thus being readily applicable to the centralised control of multiple WECs [22]. On the other hand, other works have focused on real-time control [19,20,23], exploiting the capability of neural networks to handle the predicted wave elevation over a future time horizon, similar to MPC. The main advantage of machine learning models for the system identification of WECs is that the same method can be used for different WEC technologies and is potentially adaptive to changes in the system dynamics, e.g., to subsystem failures or biofouling.

A promising solution to developing an optimal, real-time nonlinear controller for WECs inclusive of constraints on both the state and action is to cast the problem in a dynamic programming framework, similarly to MPC for WEC problems. In non-linear dynamic programming solutions, a neural network is used as a critic to approximate the time-dependent optimal cost value expressed as a Hamilton–Jacobi–Bellman equation. Numerical studies have shown the effectiveness and robustness of this approach for the control of a single WEC [24–26]. In particular, dynamic programming, also classified as model-based reinforcement learning (RL), is much more data-efficient than model-free RL [27]. Using a machine learning model trained on the collected data, such as Gaussian processes [27] or neural networks [24–26], enables the controller to plan off-line, thus significantly speeding up the learning of a suitable control policy even from a small set of samples. Conversely, model-free RL methods which learn from direct interactions with the environment require a much larger number of samples (in order of 10^8 as opposed to 10^4 for complex control tasks [28]). For this reason, to date model-free RL has been applied only to the time-averaged resistive and reactive control of WECs with discrete PTO damping and stiffness coefficients [29–31], with lower level controllers necessary to ensure constraints abidance [32]. However, model-free RL schemes are known to find the optimal control policy, even for real-time applications and very complex systems [28].

This paper introduces the world-first deep reinforcement learning (DRL) control method for WECs. The novel approach enables the real-time, nonlinear optimal control of WECs based on model-free RL. Deep learning allows the method to treat continuous control input and output features efficiently at deployment time.

To avoid unpredictable behaviour during the initial learning stage, WECs are expected to be controlled with model-based, robust methods once first deployed in the future. For a real-time implementation on complex WECs, these are likely to rely on linear models considering current technologies. After sufficient data samples are collected, the controller can move to the proposed data-driven model, whose computational cost at deployment does not increase if nonlinearities are present and can adapt to changes in the system dynamics or noncritical faults if retrained regularly. Hence, first of all, the WEC is operated in a range of representative sea states under the convex MPC proposed in [6,8]. Data samples are collected from 15-minute-long wave traces in each sea state. Subsequently, the dataset is used to train a deep neural network (DNN), defined as a neural network with more than one hidden layer according to [33], which mimics the controller behaviour. The DNN thus corresponds to the actor of an actor-critic RL strategy. The actor will be then used to initialise a model-free RL controller, as in [28]. The agent will then be further trained to optimise its behaviour as in [34].

In this article, the analysis is limited to the simulation of a standard spherical point absorber constrained to heaving motions [9]. The simulation environment is currently based on a linear model as presented in Section 2. The new DRL-based control method for WECs is described in Section 3. Finally, the performance of the trained actor is assessed directly against the original linear MPC developed in [6] in Section 4, with conclusions drawn in Section 5.

2. Linear Model of a Heaving Point Absorber

A heaving point absorber is shown schematically in Figure 1. Assuming linear potential wave theory, the equation of motion of a heaving point absorber can be expressed in the time domain as [35]

$$m\ddot{z}(t) = f_{\mathrm{e}}(t) + f_{\mathrm{r}}(t) + f_{\mathrm{h}}(t) + f_{\mathrm{m}}(t) + f_{\mathrm{PTO}}(t), \tag{1}$$

where t indicates time, z the heave displacement, f_{e} the wave excitation force inclusive of wave incidence and diffraction effects (or dynamic Froude–Krylov and scattering forces), f_{r} the wave radiation force, f_{h} the hydrostatic restoring force (or static Froude–Krylov), f_{m} the mooring force, f_{PTO} the PTO or control force, and m is the mass of the buoy. In this study, the mooring forces are ignored for simplicity.

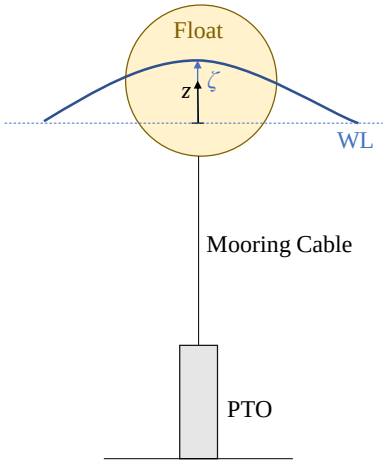

Figure 1. Heaving point absorber.

The linear excitation wave force can be obtained using the excitation impulse response function $H(t)$ as

$$f_{\mathrm{e}}(t) = \int_{-\infty}^{\infty} H(t-\tau)\zeta(t)\mathrm{d}\tau, \tag{2}$$

where ζ is the wave elevation. Similarly, the linear hydrostatic restoring force is

$$f_{\mathrm{h}}(t) = -\rho g A_{\mathrm{w}} z(t), \tag{3}$$

where ρ is the water density, g the gravitational acceleration and A_{w} the waterplane area. Using Cummins' equation [36], the radiation force can be expressed as

$$f_{\mathrm{r}}(t) = -A(\infty)\ddot{z}(t) - \int_{-\infty}^{t} K(t-\tau)\dot{z}(\tau)\mathrm{d}\tau, \tag{4}$$

where $A(\infty)$ is the heave added mass at infinite wave frequency and K is the radiation impulse response function. The convolution integral in (4) causes significant challenges for control tasks. Hence, it is common practice to approximate the convolution integral with a state-space model to improve the computational performance and ensure controllability. Here, the approach based on moment matching proposed in [37] is followed. Hence, (4) is reformulated as

$$\dot{\boldsymbol{x}}_{\mathrm{ss}}(t) = \boldsymbol{A}_{\mathrm{ss}}\boldsymbol{x}_{\mathrm{ss}} + \boldsymbol{B}_{\mathrm{ss}}\dot{z}, \tag{5a}$$

$$\int_{-\infty}^{t} K(t-\tau)\dot{z}(\tau)\mathrm{d}\tau \approx \boldsymbol{C}_{\mathrm{ss}}\boldsymbol{x}_{\mathrm{ss}} + D_{\mathrm{ss}}\dot{z}. \tag{5b}$$

Substituting (3)–(5) into (1) allows the equation of motion of the heaving point absorber to be expressed in state space form:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}_u u(t) + \boldsymbol{B}_{\mathrm{v}} v(t), \tag{6a}$$

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}, \text{ where} \tag{6b}$$

$$\boldsymbol{x} = \begin{bmatrix} z & w & \boldsymbol{x}_{\mathrm{ss}}^T \end{bmatrix}^T, \tag{6c}$$

$$u = f_{\mathrm{PTO}}, \tag{6d}$$

$$v = f_{\mathrm{e}}, \tag{6e}$$

$$\boldsymbol{B}_u = \boldsymbol{B}_v = \begin{bmatrix} 0 & M^{-1} & \boldsymbol{0}^T \end{bmatrix}^T, \tag{6f}$$

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & \boldsymbol{0} \\ -M^{-1}C & 0 & -M^{-1}\boldsymbol{C}_{\mathrm{ss}} \\ \boldsymbol{0} & \boldsymbol{B}_{\mathrm{ss}} & \boldsymbol{A}_{\mathrm{ss}} \end{bmatrix}, \tag{6g}$$

$$\boldsymbol{C} = \begin{bmatrix} 1 & 1 & \boldsymbol{0}^T \end{bmatrix}^T, \tag{6h}$$

with $M = m + A(\infty)$. The net useful energy that can be absorbed from the waves between times t_0 and t_{f} is given by

$$E = -\int_{t_0}^{t_{\mathrm{f}}} f_{\mathrm{PTO}}(t)\dot{z}(t)\mathrm{d}t. \tag{7}$$

3. Real-Time Reinforcement Learning Control of a Wave Energy Converter

RL is a decision-making framework in which an agent learns a desired behaviour, or policy π, from direct interactions with the environment [38]. As shown in Figure 2, at each time step, the agent is in a state s and takes and action a, thus landing in a new state s' while receiving a reward r. A Markov decision process is used to model the action selection depending on the value function

$Q(s,a)$, which represents an estimate of the future reward. By interacting with the environment for a long time, the agent learns an optimal policy, which maximises the total expected reward.

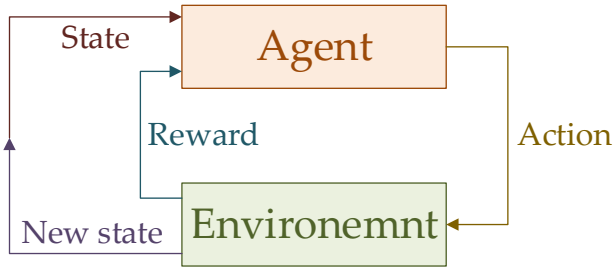

Figure 2. Block diagram of reinforcement learning (RL) control.

3.1. Problem Formulation

As a decision-making framework, RL is typically used to train an agent, or system, to perform a task that is particularly challenging to express in a standard control setting, e.g., walking for a legged robot. These tasks are usually described as episodic, i.e., the experience can be subdivided into discrete trials whose end is determined by either success in achieving the desired task, e.g., the robot has correctly made a walking step, or failure, e.g., the robot has fallen over and needs to start again. However, WEC control is clearly continuous, which will require a reformulation of the RL schemes as shown in [29–32].

In a feedforward configuration, the control of WECs is dependent on the wave excitation force and its predicted value over a future time horizon. Here, a simple state space is selected, which includes only the WEC displacement and velocity and capture the wave excitation information from the wave elevation and its rate. Therefore, the RL state space for a heaving point absorber is defined as

$$\boldsymbol{s} = \begin{bmatrix} z & \dot{z} & \zeta & \dot{\zeta} \end{bmatrix}^T. \tag{8}$$

The action space is identical to the control input u:

$$a = f_{\mathrm{PTO}}. \tag{9}$$

Although RL originated with the treatment of discrete actions, as for instance shown in [29–32], successful strategies with continuous action spaces have been recently proposed [34,39,40]. With either solution, constraints on the action can be easily imposed, so that $|a| < f_{\max}$.

Specifying an appropriate reward function is fundamental to have the agent learn the desired behaviour. Note that in RL, the optimisation problem is typically cast as a maximisation rather than a minimisation. Taking inspiration from [24–26], the reward function can be defined as

$$r = -f_{\mathrm{PTO}}\dot{z} - w_u f_{\mathrm{PTO}}^2 - w_z p_z, \tag{10}$$

where the weights w_u and w_z can be used to tune the penalty on the control action and heave displacement, respectively. Whilst a constraint can be placed on the PTO force, as it coincides with the control action, it is not possible to impose proper limits on the heave displacement. Therefore, a discontinuous function is used to determine the penalty term p_z to produce an aggressive controller:

$$p_z = \begin{cases} 0 & \text{if } |z| \leq z_{\max}, \quad \text{(11a)} \\ 1 & \text{if } |z| > z_{\max}, \quad \text{(11b)} \end{cases}$$

where z_{max} defines the displacement limit.

Another difference between the RL and MPC frameworks consists of the way the information on the future incoming waves is treated, i.e., the prediction step. In feedforward MPC, an external method, e.g., autoregressive techniques and the excitation impulse response function [41], is used to predict the incoming wave force and the information is included in the cost function to select the control action. In the RL framework, the agent learns an optimal policy for the maximisation of the total reward, which is a function of the current reward as well as discounted future rewards deriving from following either the current or the optimal policy. This means that the reward function should be specified for the current time step rather than include information from predicted future time steps. The prediction step is embedded within the RL system in a probabilistic setting.

3.2. RL Real-Time WEC Control Framework

Although trust region policy optimisation is used in [28] for a model-free controller initialised with samples obtained from an MPC controller, here actor-critic strategies are considered for the real-time control of a WEC. An example of a successful scheme with continuous state and action spaces, which are necessary for improved control performance, is soft actor-critic (SAC) [34]. In particular, SAC [34], which is the most advanced actor-critic DRL algorithm at the time of writing, is selected here for the control framework for the point absorber.

As shown in Figure 3, the controller would be split into an actor and a critic. The function of the critic is to evaluate the policy, thus updating the action-value function, which is a measure of the total discounted reward, using the samples collected from observations of the environment. The discounted reward estimated by the critic is then fed to the actor. Using the estimated action-value function, the actor selects an action based on the current state, directly interacting with the environment. The policy is then improved by learning from the collected observations. As SAC is an off-line, off-policy algorithm, the critic and actor can be updated using batches of data samples, known as experience replay buffer.

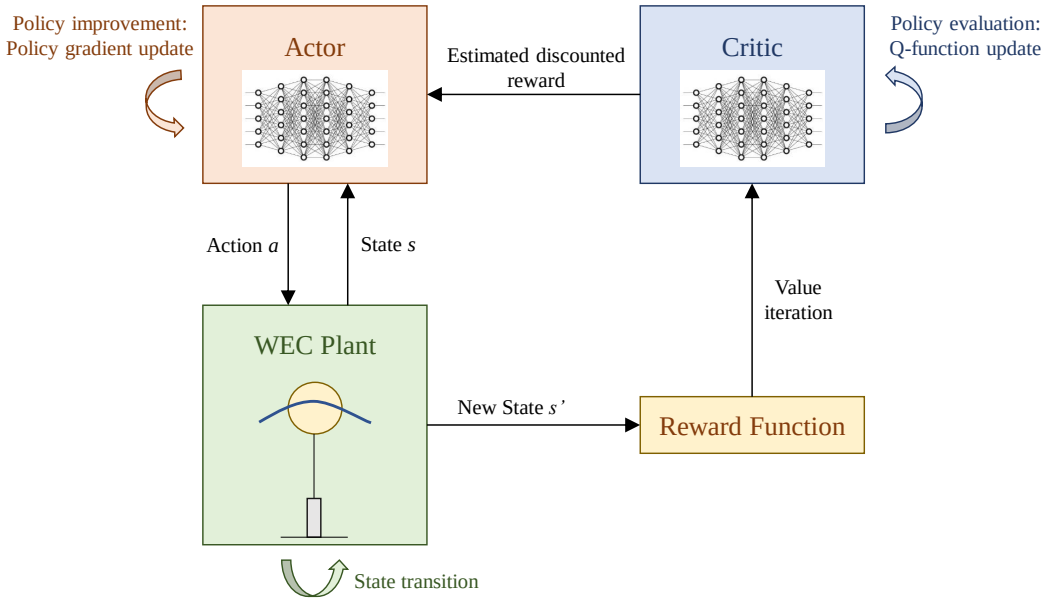

Figure 3. Diagram of the soft actor-critic (SAC) algorithm for the real-time control of a wave energy converter (WEC).

The agent seeks to maximise not only the environment's expected reward, but also the policy's entropy. The concept of entropy ensures that the agent selects random actions to explore the environment through a parameter α defined as the entropy temperature. The parameter is

automatically adjusted with gradient descent to ensure sufficient exploration at the start of learning, and subsequently a greater emphasis on the maximisation of the expected reward. A DNN is used to model the mean of the log of the standard deviation of the policy. For the policy improvement step, the policy distribution is updated towards the softmax distribution for the current Q function by minimizing the Kullback–Leibler divergence.

In SAC, two DNNs are used to approximate the critic's policy evaluation to mitigate positive bias in the policy evaluation step. The DNNs are trained off-line using batches of data collected by the actor during deployment. The minimum value of the two soft Q-functions is used in the gradient descent during training, which has been found to significantly speed up convergence. Additionally, target networks that are obtained as an exponentially moving average of the soft Q-function weights are used to smooth out the effects of noise in the sampled data.

The SAC algorithm is summarised in Algorithm 1. For a full explanation, the reader is referred to [34].

Algorithm 1: SAC algorithm taken from [34].

```
Result: Optimised actor and critic DNNs
initialise policy, two soft Q and two target soft Q DNNs;
initialise experience replay buffer with MPC samples;
for each episode do
    for each step do
        sample actions from the policy;
        sample transition from the environment;
        store the transition in the replay buffer;
    end
    for each gradient update step do
        update the soft Q DNN weights;
        update the policy DNN weights;
        adjust the entropy temperature;
        update the target DNN weights;
    end
end
```

As compared with the RL solutions presented in [29–32], once trained the new DRL implementation can be implemented in real time, with a control time step similar to the one used by other control methods, e.g., MPC.

4. Results and Discussion

4.1. Case Study

A spherical point absorber as shown in Figure 1 is selected as a case study for the development of the real-time RL WEC control scheme. The spherical buoy represents a standard case study based on the Wavestar prototype WEC, which has also been used in [9,37,42] among other studies. Additionally, the simple geometry enables the inclusion of computationally efficient nonlinear Froude–Kryolv and viscous forces in the future. The properties of the point absorber in the simulation environment can be found in Table 1. The hydrodynamic coefficients have been computed in the panel-code WAMIT. However, the same matrices as in [37] have been used for the state-space approximation of the radiation convolution integral. The problem has been programmed in the Python/Pytorch framework for the SAC controller.

In addition, the robust and computationally efficient MPC strategy described in [6,8] is selected to initialise the training and benchmark the results of the DRL scheme. The method is implemented in the MATLAB/Simulink framework using the quadratic-programming solver quadprog, after discretising the matrix equation in (6) with a zero-order hold. Similarly to [6,8], the future wave elevation is

assumed here to be known exactly, as prediction methods with 90% accuracy up to 10 s into the future have been proposed [41,43].

For both control methods, a first-order Euler scheme is used for the time integration of the simulations with a time step of 0.01 s.

Table 1. Properties of the spherical point absorber in the simulation environment.

Property	Value
Buoy diameter [m]	5
Buoy mass [kg]	32.725
Buoy resonant period [s]	3.17
Water depth [m]	∞
ρ [kg/m^3]	1000
g [m/s^2]	9.81

Typical ocean waves have an energy wave period approximately ranging from 5 s to 20 s [44]. Hence, it is clear that the selected point absorber will need significant control effort to extract energy from realistic ocean waves, since its resonant period is lower, as shown in Table 1. In this work, the peak wave period is considered to range from 4 s to 10 s, which is expected to be realistic for the small point absorber. Additionally, as the simulation environment is based here on a linear model, only small wave amplitudes up to 1 m are analysed. As a result, no constraints on either the buoy displacement or the PTO force are set on the MPC controller. The penalty on the slew rate is expected to be sufficient for the achievement of a suitable WEC response, by setting $r_{\text{MPC}} = 10^{-5}$ to ensure convexity according to [8].

Zhong and Yeung [6] have shown that, in regular waves, the mean absorbed power does not increase with time horizon duration after the horizon is one wave period long. Hence, here we set $H = 10$ s, since it corresponds to the longest wave period that is analysed and corresponds to realistic prediction timeframes [41,43]. The control time step length is set to $\delta t = 0.2$ s.

A maximum PTO force $f_{\max} = 10^5$ N and displacement $z_{\max} = 2.5$ m are selected. Additionally, the weights of (10) are set to $w_u = 10^{-5}$ and $w_z = 10^6$. The hyperparameters used for the SAC agent are the same as in [34] and are reported in Table 2 for greater clarity.

Table 2. Hyperparameters of the SAC agent.

Parameter	Value
optimizer	Adam
learning rate	3×10^{-4}
discount factor	0.99
replay buffer size	10^6
number of hidden layers (all networks)	2
number of hidden units per layer	256
number of samples per minibatch	256
entropy target	-1
activation function	ReLU
target smoothing coefficient	0.005
target update interval	2
gradient steps	1

4.2. Results in Irregular Waves

To generate sufficient data samples for the training of the actor DNN, 28 wave traces of irregular waves lasting 15 min each are produced, with the significant wave height ranging from 0.5 m to 2 m in steps of 0.5 m and the peak wave period from 4 s to 10 s in steps of 1 s. A Bretschneider spectrum is used [44]. The controller is started only after 100 s from the start of the wave trace to avoid numerical instabilities during the initial transient. The wave trace is logged after an additional 50 s for 900 s.

4.2.1. Training

The sampled data is used to initialise the experience replay buffer of the SAC agent. Each episode consists of a randomly initialised wave trace whose significant wave height and peak wave period are randomly selected in the 1.5–2 m and 6–8 s range, respectively. The wave trace lasts for 200 s and is initialised with no control force for the first 100 s to avoid numerical instabilities. Hence, each episode lasts a total of 2001 steps for the selected control time step of 0.2 s. The same control time step is selected for the DRL controller. To ensure the robustness of the algorithm, the agent is trained with five different seed values to the random number generator.

As can be seen in Figure 4, after the initialisation with the samples collected by the MPC controller, the agent learns a policy to maximise the expected reward after approximately 50 episodes (or approximately 10^5 steps). Note that in Figure 4, the total reward per episode is highly dependent on the randomly selected significant wave height and peak period; hence, large variations are possible even after training, due to the different level of energy in the waves.

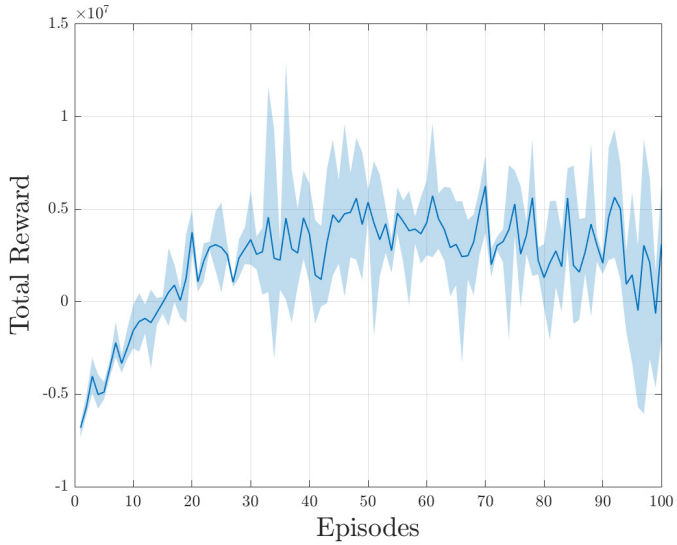

Figure 4. Total reward per episode during training.

The convergence time corresponds to approximately 5000 s of experience in addition to the previous 25,200 s with MPC control for a total of approximately 8.4 h. This is a really short time over the life time of the WEC and provides confidence in the controller being able to deliver adaptive control in practical implementations. However, the negative absorbed powers shown during the first episodes are highly worrying. At the start of learning, the agent is preferring random actions to ensure exploration. However, during exploration the device, and in particular the PTO, may fail. Therefore, in the future, a fixed entropy temperature may reduce exploration at the start and thus its associated risks if the controller is already initialised with data from a robust controller. This solution is however likely to slow down the training time.

4.2.2. Comparison between SAC and MPC

To assess the performance of the DRL control, MPC and SAC are tested in unseen waves. The traces have a Bretschneider spectrum in the same range of significant wave height and peak wave period, but different seed numbers to the random number generator from the training set. They last 1050 s, with the controller initialised after 100 s and the averaging to compute the mean power started after a further 50 s.

The mean useful or net absorbed power is shown in the dashed lines in Figure 5 for MPC. The reactive power, P_{rea}, is defined as the power transferred from the PTO to the point absorber, whilst the active or resistive power, P_{act}, is the power transferred from the absorber to the PTO. The net useful power is thus $P_{\text{u}} = P_{\text{act}} - P_{\text{rea}}$. For the MPC, the extracted power at higher wave periods is curtailed by the penalty on the slew rate. In Figure 6, the ratio of the reactive and active power for the MPC can be seen in the dashed lines. Reactive power is primarily used to speed the WEC up in shorter waves, i.e., when the wave period is shorter than the resonant period, whereas passive damping can be used to slow the device down for longer wave periods. The steady increase in the ratio of the reactive and active power for higher wave periods for the MPC is thus unexpected. The greater control effort the further from the resonance period (3.17 s for the point absorber) is however visible also in [6] and can be explained with the decrease of the absolute value of the active power for longer waves. Furthermore, a comparison with the case studies in [6,8] shows that the selected value of r_{MPC} is providing a stronger influence in this example.

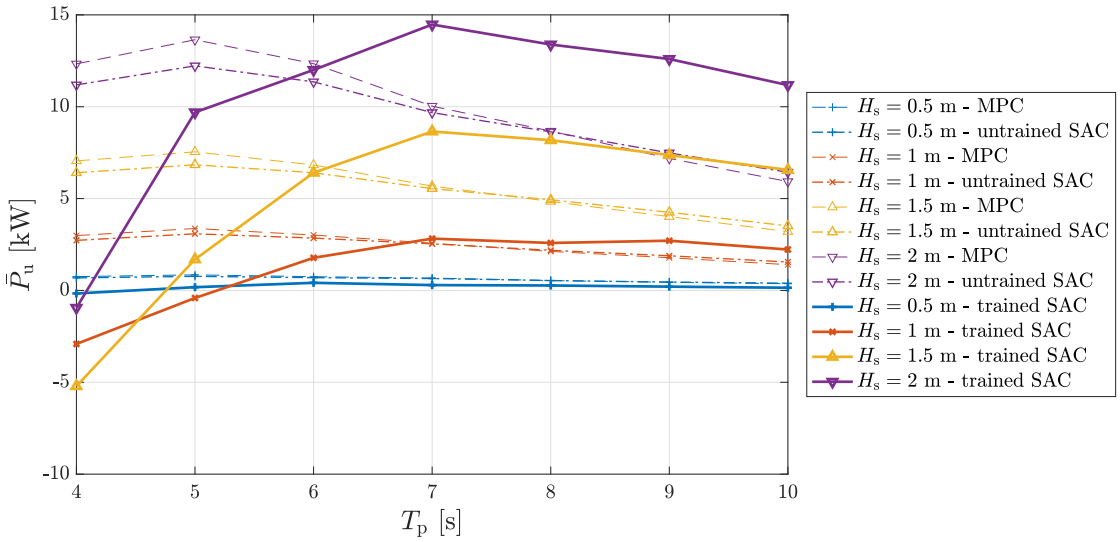

Figure 5. Variation with peak wave period of the mean useful power absorbed by the heaving sphere for the considered range of significant wave height.

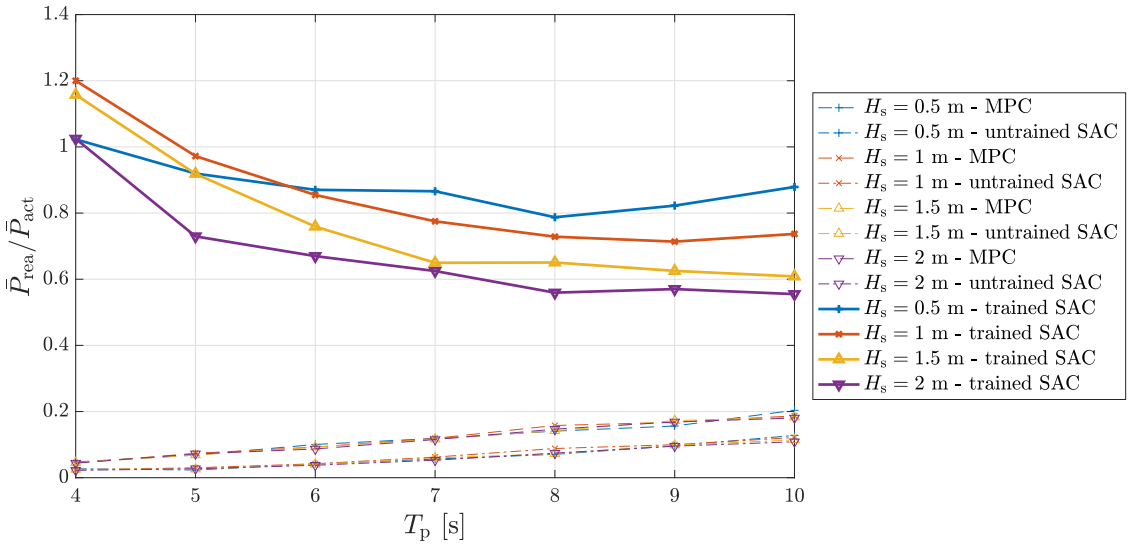

Figure 6. Variation with peak wave period of the ratio of the mean reactive and active power absorbed by the heaving sphere for the considered range of significant wave height.

Figures 5 and 6 also display the performance of the SAC agent before and after training. The dot-dash lines correspond to the SAC agent before training, with the entropy set to zero. In this case, the agent replicates closely the MPC behaviour, even though there are differences in the actual time-domain response. After training, the SAC agent (shown with thick continuous lines) improves the energy absorption for the higher peak wave period values ($T_p > 6$ s) and the higher significant wave height values ($T_p > 1$ m). These ranges correspond with the ranges used during training and show poor generalisation ability. The negative mean absorbed power values for the lower periods close to the WEC's resonant period are particularly worrying. From Figure 6, it is clear that the main cause for this behaviour is the aggressive policy that the SAC agents selects. The large flows of reactive power are useful for periods smaller than the resonant period, i.e., in shorter waves, but unhelpful close to the resonant period or for long wave periods, where damping is more useful to slow the WEC down. The problem may be caused by the low resonant period of the point absorber. A larger device whose resonant period is within the typical ocean waves period range should be selected in the future to assess the behaviour of the controller for both short and long waves.

In Figure 7, the magnitude of the maximum heave displacement and PTO force can be seen. Although the SAC algorithm presents higher displacements than MPC, the maximum value of 2.5 m is not exceeded. This hints at the efficacy of the discontinuous penalty term in (10). However, designing a method to guarantee the constraint handling for the displacement is critical for the DRL controller to find an industrial application in the future. The aggressive behaviour of the SAC agent is further underlined in Figure 7b, where the peak PTO force is hit in all sea states.

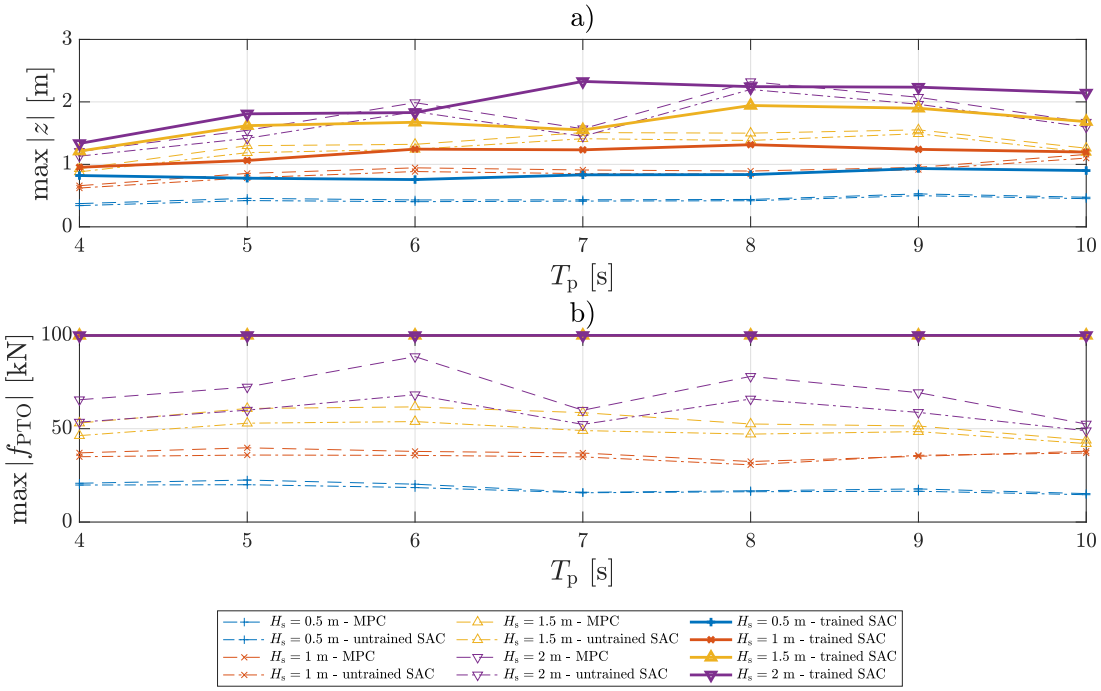

Figure 7. Variation with peak wave period of the maximum absolute displacement (**a**) and PTO force (**b**) for the heaving sphere for the considered range of significant wave height.

The response of the MPC and SAC algorithms in the time domain can be seen in Figure 8. The figure shows an extract of one of the simulations used to verify the performance of the SAC scheme against MPC ($H_s = 2$ m and $T_p = 6$ s). From the figure, it is clear that the SAC converges onto an aggressive bang-bang control policy.

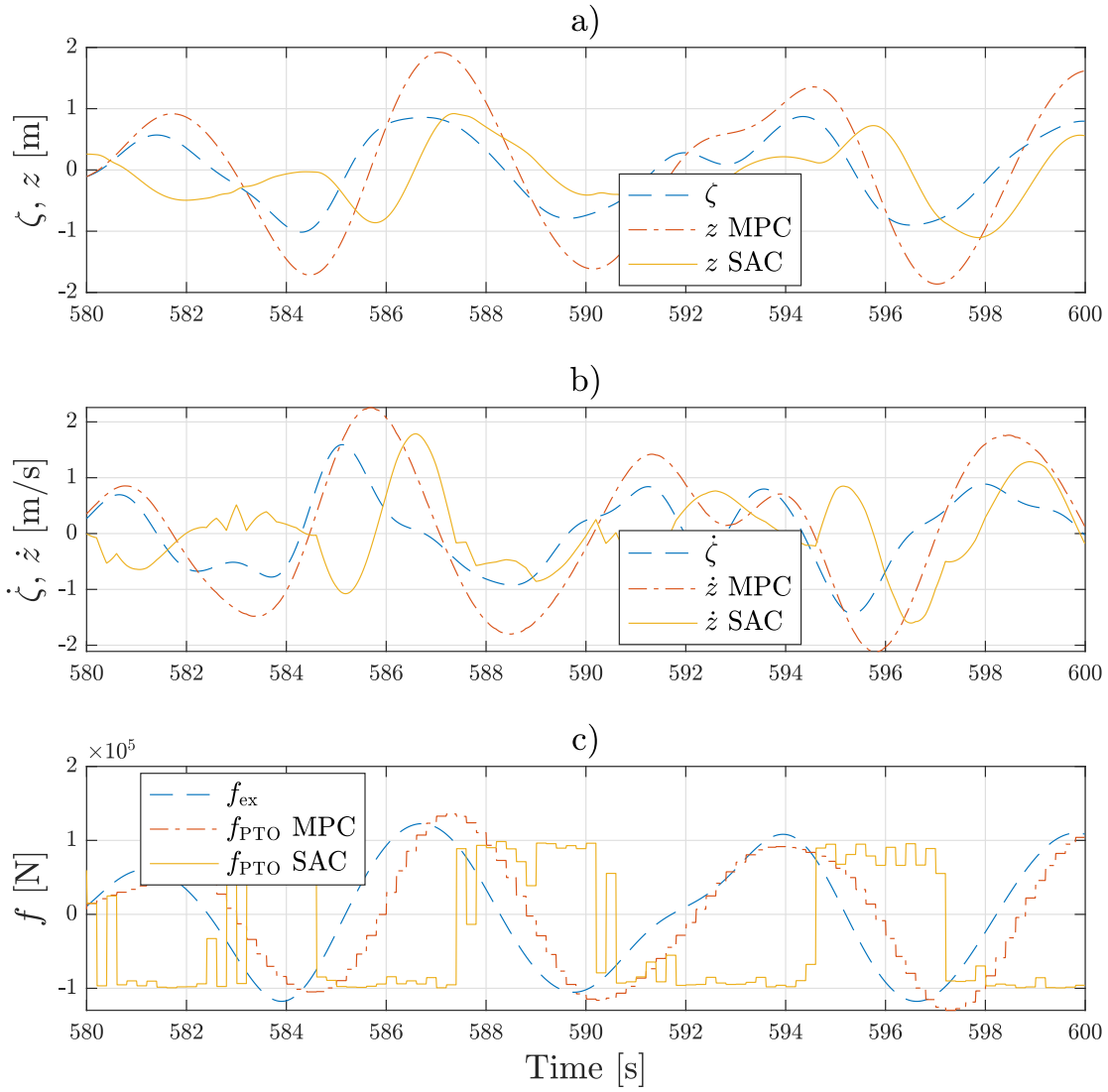

Figure 8. Time variation of the wave elevation and float heave displacement (**a**), wave and heave velocity (**b**) and wave excitation and PTO force (**c**) for the simulation of the WEC in an irregular wave trace with $H_s = 2$ m and $T_p = 6$ s.

The aggressiveness of the controller can be reduced by increasing the penalty on the PTO force (through w_u). The poor ability of the SAC agent to generalise to unseen wave conditions is problematic and symptomatic of a possibly over-simplistic selection of the state-space. In [34], the information from a number of past time steps is captured in the state-space to ensure convergence for the control of a walking robot. A similar approach will be needed for the control of a WEC to capture the oscillatory nature of gravity waves, similar to MPC. Furthermore, it is clear that the experience replay buffer should include data samples from a broad range of sea states, in particular with regards to period both below and above the resonance period of the device. Currently, the memory buffer is updated with new samples by removing the oldest sample if the memory is full. This technique will be changed by binning the data by wave period and height and ensuring a minimum number of samples per bin.

Table 3 shows the computational time required to train the SAC controller (over 100 episodes) and to run a simulation of the WEC lasting 1050 s using the MPC and trained SAC schemes. The mean from the 28 simulations employed to compare the two strategies is used. Note that the first 150 s are needed to initialise the WEC dynamics and power averaging and that the control time step is 0.2 s for both algorithms. Hence, there are 4501 control time steps per simulation, leading to the values for

the computational time per time step shown in Table 3. The simulations are run on a laptop with an Intel 5, 2.3 GHz, dual-core processor and 16 GB RAM.

Table 3. Training time (if applicable), mean total simulation time and time per control time step for the simulations used to verify and compare the MPC and SAC control algorithms.

Scheme	Training Time [s]	Total Simulation Time [s]	Time Per Control Time Step [s]
MPC	-	11.798	2.6×10^{-3}
SAC	869	2.607	5.8×10^{-4}

As can be seen in Table 3, the SAC algorithm requires approximately 15 min to train over 100 episodes for analysed point absorber. The large computational time prevents an on-line application, although training can happen regularly off-line in practice, once significantly large new batches of data are collected. Conversely, once trained, the computational effort is 40 times lower than the simulation control time step, thus enabling a real-time implementation with ease. Additionally, the computational effort associated with SAC is one order of magnitude smaller than for linear MPC. In fact, the computational time per control time step shown in Table 3 is overly conservative for SAC, as it includes the overhead from the dynamic simulation in Python. Conversely, the linear MPC code is implemented in a very efficient MALTAB/Simulink script with C-coded S-functions. Therefore, a gain in performance as high as one additional order of magnitude is expected from compiled solutions if the system has to be implemented on an actual WEC [45].

For reproducibility, the results of the SAC algorithm in the unseen test wave traces can be accessed on Github[1], including the wave elevation, vertical velocity and excitation force.

5. Conclusions

In this article, an established convex MPC has been used to generate observations for a heaving point absorber in a range of irregular waves in a linear simulation environment. The samples have been used to initialise a DRL agent, which learns an optimal policy from direct interactions with with the environment for the maximisation of the energy absorption. By being off-line and off-policy, the SAC algorithm enables the training to be decoupled from deployment, thus shifting the computational effort on the training. This is a fundamental trait, as the control of large groups of WECs in the future to achieve economies of scale is reliant on having an effective, real-time centralised strategy.

The DRL control improves the energy absorption of the point absorber over convex MPC for wave periods higher than the resonant period of the device, whilst meeting the displacement and force constraints. This is achieved by adopting a more aggressive policy with higher slew rate. However, poorer performance is shown for lower wave height and period values. These problems will be addressed by reformulating the state-space, updating the reward function and the sampling of data for the experience replay. Additionally, the exploration will be reduced from the start of training to prevent the controller from taking actions that damage the PTO. Furthermore, the DRL controller will be tested in a simulation environment inclusive of nonlinear effects, e.g., nonlinear static and dynamic Froude–Krylov forces as in [42] and viscous drag. A sensitivity analysis will be run to assess the impact of modelling errors on the performance of the DRL and MPC algorithms.

Author Contributions: Conceptualization, E.A.; methodology, E.A.; software, E.A. and S.H. (hydrodynamics); validation, E.A.; formal analysis, E.A.; investigation, E.A.; resources, E.A. and G.G.P.; data curation, E.A. and S.H.; writing—original draft preparation, E.A.; writing—review and editing, G.G.P., M.A. and G.T.; visualization, E.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

[1] https://github.com/enricoande/Results-for-jmse-967614

Acknowledgments: The authors would like to acknowledge the help provided by Giuseppe Giorgi in identifying the Special Issue.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DNN	Deep Neural Network
DRL	Deep Reinforcement Learning
LCoE	Levellised Cost of Energy
MPC	Model Predictive Control
PTO	Power Take-Off
RL	Reinforcement Learning
SAC	Soft Actor-Critic
WEC	Wave Energy Converter

References

1. Kempener, R.; Neumann, F. *Wave Energy: Technology Brief 4*; International Renewable Energy Agency Technical Report; International Renewable Energy Agency: Abu Dhabi, UAE, 2014.
2. Sgurr Control; Quoceant. *Control Requirements for Wave Energy Converters Landscaping Study: Final Report*; Technical report; Wave Energy Scotland: Inverness, Scotland, 2016.
3. Luis Villate, J.; Ruiz-Minguela, P.; Berque, J.; Pirttimaa, L.; Cagney, D.; Cochrane, C.; Jeffrey, H. *Strategic Research and Innovation Agenda for Ocean Energy*; Technical report; ETIPOCEAN: Bruxelles, Belgium, 2020.
4. Faedo, N.; Olaya, S.; Ringwood, J.V. Optimal control, MPC and MPC-like algorithms for wave energy systems: An overview. *IFAC J. Syst. Control.* **2017**. [CrossRef]
5. Li, G.; Belmont, M.R. Model predictive control of sea wave energy converters—Part I: A convex approach for the case of a single device. *Renew. Energy* **2014**, *69*, 453–463. [CrossRef]
6. Zhong, Q.; Yeung, R.W. An efficient convex formulation for model predictive control on wave-energy converters. In Proceedings of the 36th International Conference on Ocean, Offshore and Arctic Engineering, Trondheim, Norway, 25–30 June 2017. [CrossRef]
7. Li, G.; Belmont, M.R. Model predictive control of sea wave energy converters—Part II: The case of an array of devices. *Renew. Energy* **2014**, *68*, 540–549. [CrossRef]
8. Zhong, Q.; Yeung, R.W. Model-Predictive Control Strategy for an Array of Wave-Energy Converters. *J. Mar. Sci. Appl.* **2019**, *18*, 26–37. [CrossRef]
9. Giorgi, G.; Ringwood, J.V. Nonlinear Froude-Krylov and viscous drag representations for wave energy converters in the computation/fidelity continuum. *Ocean Eng.* **2017**, *141*, 164–175. [CrossRef]
10. Richter, M.; Magana, M.E.; Sawodny, O.; Brekken, T.K.A. Nonlinear Model Predictive Control of a Point Absorber Wave Energy Converter. *IEEE Trans. Sustain. Energy* **2013**, *4*, 118–126. [CrossRef]
11. Li, G. Nonlinear model predictive control of a wave energy converter based on differential flatness parameterisation. *Int. J. Control* **2017**, *90*, 68–77. [CrossRef]
12. Son, D.; Yeung, R.W. Optimizing ocean-wave energy extraction of a dual coaxial-cylinder WEC using nonlinear model predictive control. *Appl. Energy* **2017**, *187*, 746–757. [CrossRef]
13. Oetinger, D.; Magaña, M.E.; Sawodny, O. Centralised model predictive controller design for wave energy converter arrays. *IET Renew. Power Gener.* **2015**, *9*, 142–153. [CrossRef]
14. Ringwood, J.V.; Bacelli, G.; Fusco, F. Energy-Maximizing Control of Wave-Energy Converters: The Development of Control System Technology to Optimize Their Operation. *IEEE Control Syst. Mag.* **2014**, *34*, 30–55. [CrossRef]
15. Korde, U.A.; Ringwood, J.V. *Hydrodynamic Control of Wave Energy Devices*; Cambridge University Press: Cambridge, UK, 2016.
16. Fusco, F.; Ringwood, J.V. A simple and effective real-time controller for wave energy converters. *IEEE Trans. Sustain. Energy* **2013**, *4*, 21–30. [CrossRef]

17. Gaspar, J.F.; Kamarlouei, M.; Sinha, A.; Xu, H.; Calvário, M.; Faÿ, F.X.; Robles, E.; Soares, C.G. Speed control of oil-hydraulic power take-off system for oscillating body type wave energy converters. *Renew. Energy* **2016**, *97*, 769–783. [CrossRef]
18. Valério, D.; Mendes, M.J.G.C.; Beirão, P.; Sá da Costa, J. Identification and control of the AWS using neural network models. *Appl. Ocean Res.* **2008**, *30*, 178–188, [CrossRef]
19. Li, L.; Yuan, Z.; Gao, Y. Maximization of energy absorption for a wave energy converter using the deep machine learning. *Energy* **2018**, *165*, 340–349. [CrossRef]
20. Li, L.; Gao, Z.; Yuan, Z.M. On the sensitivity and uncertainty of wave energy conversion with an artificial neural-network-based controller. *Ocean Eng.* **2019**, *183*, 282–293. [CrossRef]
21. Anderlini, E.; Forehand, D.I.; Bannon, E.; Abusara, M. Reactive control of a wave energy converter using artificial neural networks. *Int. J. Mar. Energy* **2017**, *19*, 207–220. [CrossRef]
22. Thomas, S.; Giassi, M.; Eriksson, M.; Göteman, M.; Isberg, J.; Ransley, E.; Hann, M.; Engström, J. A Model Free Control Based on Machine Learning for Energy Converters in an Array. *Big Data Cogn. Comput.* **2018**, *2*, 36. [CrossRef]
23. Tri, N.M.; Truong, D.Q.; Thinh, D.H.; Binh, P.C.; Dung, D.T.; Lee, S.; Park, H.G.; Ahn, K.K. A novel control method to maximize the energy-harvesting capability of an adjustable slope angle wave energy converter. *Renew. Energy* **2016**, *97*, 518–531. [CrossRef]
24. Na, J.; Li, G.; Wang, B.; Herrmann, G.; Zhan, S. Robust Optimal Control of Wave Energy Converters Based on Adaptive Dynamic Programming. *IEEE Trans. Sustain. Energy* **2019**, *10*, 961–970. [CrossRef]
25. Na, J.; Wang, B.; Li, G.; Zhan, S.; He, W. Nonlinear constrained optimal control of wave energy converters with adaptive dynamic programming. *IEEE Trans. Ind. Electron.* **2019**, *66*, 7904–7915. [CrossRef]
26. Zhan, S.; Na, J.; Li, G. Nonlinear Noncausal Optimal Control of Wave Energy Converters via Approximate Dynamic Programming. *IEEE Trans. Ind. Infor.* **2019**, *15*, 6070–6079. [CrossRef]
27. Kamthe, S.; Deisenroth, M.P. Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control. In Proceedings of the Machine Learning Research, Lanzarote, Spain, 9–11 April 2018; Volume 84, pp. 1701–1710.
28. Nagabandi, A.; Kahn, G.; Fearing, R.S.; Levine, S. Neural Network Dynamics for Model-Based Deep Reinforcement Learning with Model-Free Fine-Tuning. *arXiv* **2017**, arXiv:1708.02596v2.
29. Anderlini, E.; Forehand, D.I.M.; Stansell, P.; Xiao, Q.; Abusara, M. Control of a Point Absorber using Reinforcement Learning. *IEEE Trans. Sustain. Energy* **2016**, *7*, 1681–1690. [CrossRef]
30. Anderlini, E.; Forehand, D.I.; Bannon, E.; Abusara, M. Control of a Realistic Wave Energy Converter Model Using Least-Squares Policy Iteration. *IEEE Trans. Sustain. Energy* **2017**, *8*, 1618–1628. [CrossRef]
31. Anderlini, E.; Forehand, D.I.; Bannon, E.; Xiao, Q.; Abusara, M. Reactive control of a two-body point absorber using reinforcement learning. *Ocean Eng.* **2018**, *148*, 650–658. [CrossRef]
32. Anderlini, E.; Forehand, D.I.; Bannon, E.; Abusara, M. Constraints Implementation in the Application of Reinforcement Learning to the Reactive Control of a Point Absorber. In Proceedings of the 36th International Conference on Ocean, Offshore and Arctic Engineering, Trondheim, Norway, 25–30 June 2017. [CrossRef]
33. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
34. Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; et al. Soft Actor-Critic Algorithms and Applications. *arXiv* **2018**, arXiv:1812.05905.
35. Falnes, J. *Ocean Waves and Oscillating Systems*, paperback ed.; Cambridge University Press: Cambridge, UK, 2005. [CrossRef]
36. Cummins, W.E. The impulse response function and ship motions. *Schiffstechnik* **1962**, *47*, 101–109.
37. Faedo, N.; Peña-Sanchez, Y.; Ringwood, J.V. Finite-order hydrodynamic model determination for wave energy applications using moment-matching. *Ocean Eng.* **2018**, *163*, 251–263. [CrossRef]
38. Sutton, R.S.; Barto, A.G. *Reinforcement Learning*, hardcover ed.; MIT Press: Cambridge, MA, USA, 1998; p. 344.
39. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. In Proceedings of the International Conference on Learning Representations, San Juan, PR, USA, 2–4 May 2016. [CrossRef]
40. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018.

41. Fusco, F.; Ringwood, J. Short-Term Wave Forecasting for time-domain Control of Wave Energy Converters. *IEEE Trans. Sustain. Energy* **2010**, *1*, 99–106. [CrossRef]
42. Giorgi, G.; Ringwood, J.V. Computationally efficient nonlinear Froude–Krylov force calculations for heaving axisymmetric wave energy point absorbers. *J. Ocean. Eng. Mar. Energy* **2017**, *3*, 21–33. [CrossRef]
43. Paparella, F.; Monk, K.; Winands, V.; Lopes, M.F.; Conley, D.; Ringwood, J.V. Up-wave and autoregressive methods for short-term wave forecasting for an oscillating water column. *IEEE Trans. Sustain. Energy* **2015**, *6*, 171–178. [CrossRef]
44. Holthuijsen, L.H. *Waves in Oceanic and Coastal Waters*; Cambridge University Press: Cambridge, UK, 2007. [CrossRef]
45. Fourment, M.; Gillings, M.R. A comparison of common programming languages used in bioinformatics. *BMC Bioinform.* **2008**, *9*, 82. [CrossRef] [PubMed]