Data-driven modeling of an oscillating surge wave energy converter using dynamic mode decomposition

B. Lydon,[1, a)] B. Polagye,[1] and S. Brunton[1]

Department of Mechanical Engineering, University of Washington, Seattle, WA, USA

(Dated: 21 July 2025)

Modeling wave energy converters (WECs) to accurately predict their hydrodynamic behavior has been a challenge for the wave energy field. Often, this results in either low-fidelity linear models or high-fidelity numerical models that are too computationally expensive for operational use. To bridge this gap, we propose the use of dynamic mode decomposition (DMD) as a purely data-driven technique that can generate an accurate and computationally efficient model of WEC dynamics. Specifically, we model and predict the behavior of an oscillating surge wave energy converter (OSWEC) in mono- and polychromatic seas without an equation of motion or knowledge of the incident wave field. We generate data with the open-source code WEC-Sim, then evaluate how well DMD can describe past dynamics and predict future behavior. We consider realistic challenges including noisy sensors, nonlinear dynamics, and irregular wave forcing. Specifically, by using an extension of DMD we reduce the effect of noise on our system and significantly increase model accuracy outside the training region. Additionally, by introducing time delays we accurately describe weakly nonlinear dynamics, even though DMD is a linear algorithm. Finally, we use Optimized DMD (optDMD) to model OSWEC behavior in response to irregular waves. While optDMD accurately models training data, future prediction was inaccurate, demonstrating the limits of modeling efforts without access to information about the incident wave field. These findings provide insight into the use of DMD, and its extensions, on systems with limited time-resolved data and present a framework for applying similar analysis to lab- or field-scale experiments.

[a)]Electronic mail: BrittLyd@uw.edu

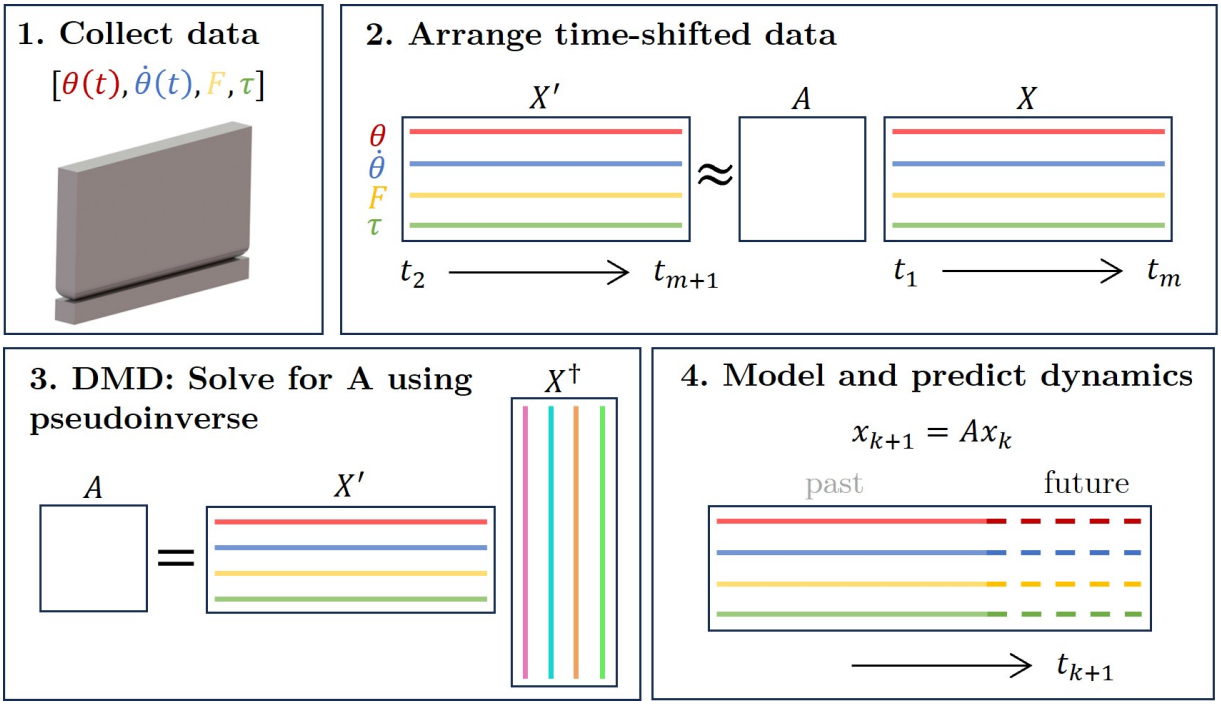

FIG. 1. Graphical representation of DMD algorithm. (1) We consider kinematic and dynamic time series from an oscillating surge wave energy converter (OSWEC). (2) The data are arranged into matrices, where linear operator $\boldsymbol{A}$ maps data at time step m to $m+1$. (3) We then solve for $\boldsymbol{A}$ using the pseudoinverse of data matrix $\boldsymbol{X}$ and then (4) use this model and predict system dynamics.

I. INTRODUCTION

Wave energy converters (WECs) have promise for carbon-free energy generation, especially for coastal communities[1]. Although there has been significant innovation in the wave energy field, modeling and controlling WEC behavior continues to be a difficult task due to the volatile nature of the ocean and the complicated fluid-structure interaction between the device and the waves[2]. A promising WEC technology is the oscillating surge wave energy converter (OSWEC)[3] due to its ability to absorb power over a wide range of wave frequencies[4] and its applications in nearshore desalination[5]. OSWECs, an example of which is shown in Figure 1, are flap-type devices that primarily harness the surge motion of ocean waves, which causes a buoyant flap to oscillate in pitch about a hinge[6].

Despite the benefits of OSWECs, there are significant challenges to their development, as evidenced by the commercial failure of the Aquamarine Oyster in 2015[6]. Although OS-

WECs operate in a single degree of freedom, their dynamics are dominated by radiation and diffraction torques and they can exhibit viscous drag effects[7], which can be difficult to accurately model. Because of this, models based in linear wave and potential flow theory may have insufficient fidelity[4], while computational fluid dynamics (CFD) or smoothed particle hydrodynamics (SPH) may incur too high a computational cost to be used for applications such as real-time state estimation and optimal control[8]. This motivates modeling techniques for OSWECs that are both accurate enough to model these systems in realistic conditions and fast enough to be used in a real-time control scheme, such as model predictive control (MPC).

To address these limitations, a number of studies use data-driven techniques to model the behavior of different WEC archetypes. Data-driven algorithms learn the dynamics of the system directly from collected data and are often more computationally efficient than CFD or SPH. Folley (Chap. 7)[9] and Davidson and Costello[10] provide in-depth reviews of data-driven WEC modeling. Here, we review a few common methods. Multiple authors have used system identification techniques such as training autoregressive exogenous input (ARX) and Kolmogorov-Gabor polynomial (KGP) models using experimental[11–15] or numerical[16–19] wave tank data to build input/output models that relate WEC kinematics to wave elevation. Specifically, Giorgi et al. built accurate linear (ARX) and nonlinear (KGP) polynomial models that relate float position to wave elevation for a point absorber WEC, where the model weights, order, and structure, were all determined using experimental wave tank data[11]. In addition, van't Hoff et al.[20] used Volterra theory to build accurate models of OSWEC kinematics where they calculated the first-order Volterra kernel using experimental data. These kernels were then used in a numerical simulation to model and predict OSWEC position and velocity, with goals of expanding this work to include higher-order kernels to model nonlinear behavior. Finally, artificial neural networks (ANN) have also shown to be effective in creating black-box data-driven models of WEC behavior[21–23]. In particular, Katsidoniotaki et al. used an ANN to model mooring forces on a heaving point absorber in response to extreme waves[23]. The neural network was trained on CFD data and the resulting data-driven model mapped extreme sea states (including significant wave height and peak period) and WEC power take-off characteristics to mooring forces acting on the WEC. Despite the body of work dedicated to data-driven WEC modeling, the majority of studies focus on point absorbers or oscillating water columns, which can exhibit vastly

different hydrodynamic behavior than OSWECs[4].

An additional approach to data-driven modeling is dynamic mode decomposition (DMD)[24–27] (Figure 1). DMD is an equation-free modal decomposition technique that can model and predict the behavior of complex dynamical systems, including fluid flows[24,28,29], as well as create robust and computationally efficient models for optimal control[30–32]. DMD is an established and powerful method in multiple fields (a summary of applications is given by Kutz et al.[27]) and has been effective in modeling experimental systems[24,28]. Here, our aim is to use DMD to generate a purely data-driven system model of an OSWEC to describe and predict its behavior in a computationally efficient and accurate manner.

In its most general form, DMD decomposes spatiotemporal time series data into spatially coherent dynamic modes with associated complex eigenvalues that determine the frequency and growth/decay rates of the respective mode behavior in time. These modes and eigenvalues can be used to build reduced-order models of dynamical systems and forecast system behavior, all without knowing the physics of the system. DMD eigenvalues and eigenvectors approximate those of the Koopman operator[25,29,33–38], which is an infinite-dimensional linear operator that describes the dynamics of a fully nonlinear system. DMD contains properties of both proper orthogonal decomposition (POD) and the Fourier transform, as it identifies spatial modes (characteristic of POD) whose time behavior is characterized by a corresponding frequency (characteristic of the Fourier transform)[30,39]. DMD has three major applications: identifying and characterizing system behavior, state estimation and/or future state prediction, and control[27]. Although often applied to spatiotemporal data, DMD also can be used with time-resolved sensor data with minimal modification.

DMD is potentially well suited to model WEC hydrodynamics for three main reasons:

- **Equation-free modeling.** DMD generates a model purely from time-resolved data, which means DMD can be used to predict WEC behavior without knowing or solving the underlying governing equations. Since OSWEC dynamics are not always well described with analytical techniques[4], DMD can identify dynamics that are not well represented by linear simplifications of OSWEC governing equations. Although we are using DMD to model data from a specific WEC archetype, DMD can be used for a range of other WEC devices and can develop linear models using data from experiments, simulations, or field measurements, making it advantageous as a general

tool for representing WEC dynamics.

- **Simple, but adaptable.** DMD is based in straight-forward linear algebra, resulting in low computational cost and interpretable outputs. The algorithm only requires a few lines of code, which means it is not a "black box" approach and also allows for adaptations of the basic algorithm to expand its use and address common issues in practical applications. Schmid[40] summarizes the expansive adaptations that have been developed for DMD. These adaptations, along with its connection to the Koopman operator, are a unique benefit of DMD and make the algorithm applicable to a wider range of systems. For example, DMD can be sensitive to noise, causing high error in systems with low signal-to-noise ratio. Both Hemati et al.[41] and Dawson et al.[42] developed extensions to DMD to significantly reduce the effect noise has on DMD models with a minimal increase in computation efforts (further discussed in Section II C 1). Because data from WECs can range in quality and complexity, these extensions can drastically increase the applicability of DMD.

- **Optimal control.** Finally, because DMD can forecast system behavior, the reduced-order model generated from DMD can be used in model predictive control (MPC). For example, Jia et al. successfully built a model predictive controller for a nonlinear heaving point absorber in simulation and used a variation of DMD called extended DMD (eDMD) to create the system model[43]. They showed that DMD was capable of building accurate models to predict and optimize the behavior of the WEC, and even outperformed other linear modeling techniques. However, there are some limiting assumptions, including noise-free data, that we explore in this work. MPC is a particularly attractive option for WECs since optimal control schemes have been shown to increase the efficiency of different WEC archetypes compared to passive control methods[44,45] and traditional linear models may not be accurate or fast enough to be used for this application[46,47].

Separate from WECs, DMD has been applied to several ocean engineering problems. For example, Serani et al.[48] and Diez et al.[49] successfully applied DMD to ship maneuvering in both regular and irregular waves. Both studies found that models generated from DMD were able to accurately forecast ship forces and trajectories for various ship operations in multiple wave periods. In general, the studies found while DMD works well for situations

with predominantly monochromatic dynamics, irregular wave inputs and nonlinear dynamics can reduce the accuracy of the DMD forecasts and often benefit from variants of DMD.

Our objective is to generate a data-driven model using DMD that can predict OSWEC behavior without incurring an infeasible computational cost or sacrificing accuracy. In particular, we aim for this model to address well known issues in WEC modeling, including handling noisy signals, modeling nonlinear WEC dynamics, and modeling WEC behavior in irregular seas. While the models generated in this study are specific to this OSWEC system, the method should generalize to other WEC archetypes. The remainder of the paper is laid out as follows: Section II describes the methods we used in this study, as well as the three main cases we are testing. Section III presents the results and discussions of DMD performance on OSWEC behavior, with an emphasis on treatment of realistic challenges. Specifically, Section III A considers DMD with noisy data, Section III B considers DMD when weakly nonlinear dynamics are present, and Section III C considers DMD modeling of an OSWEC in an irregular wave field. Finally, Section IV outlines our conclusions and future work.

II. METHODS

An overview of the methods we used for this study are outlined in Figure 2. There are four major steps: generate data, preprocess data, run DMD, and evaluate performance.

A. Generate data

The first step of our workflow is to generate the data that will be used to train the DMD model (Figure 2, Block 1). This step includes choosing the data source, which OSWEC parameters to include as system states, and wave conditions. Because there is a limited quantity of benchmark experimental or high-fidelity CFD data for OSWECs in the public domain, we use the mid-fidelity modeling tool, WEC-Sim[50,51] to generate the training and testing data for this work. However, this same process can be used with limited variations on data from high-fidelity CFD or experiments. The synthetic data approach also allows us to highlight the existing network of DMD extensions in a systematic manner.

A diagram of the OSWEC system we use in this study is shown in Figure 3 and consists

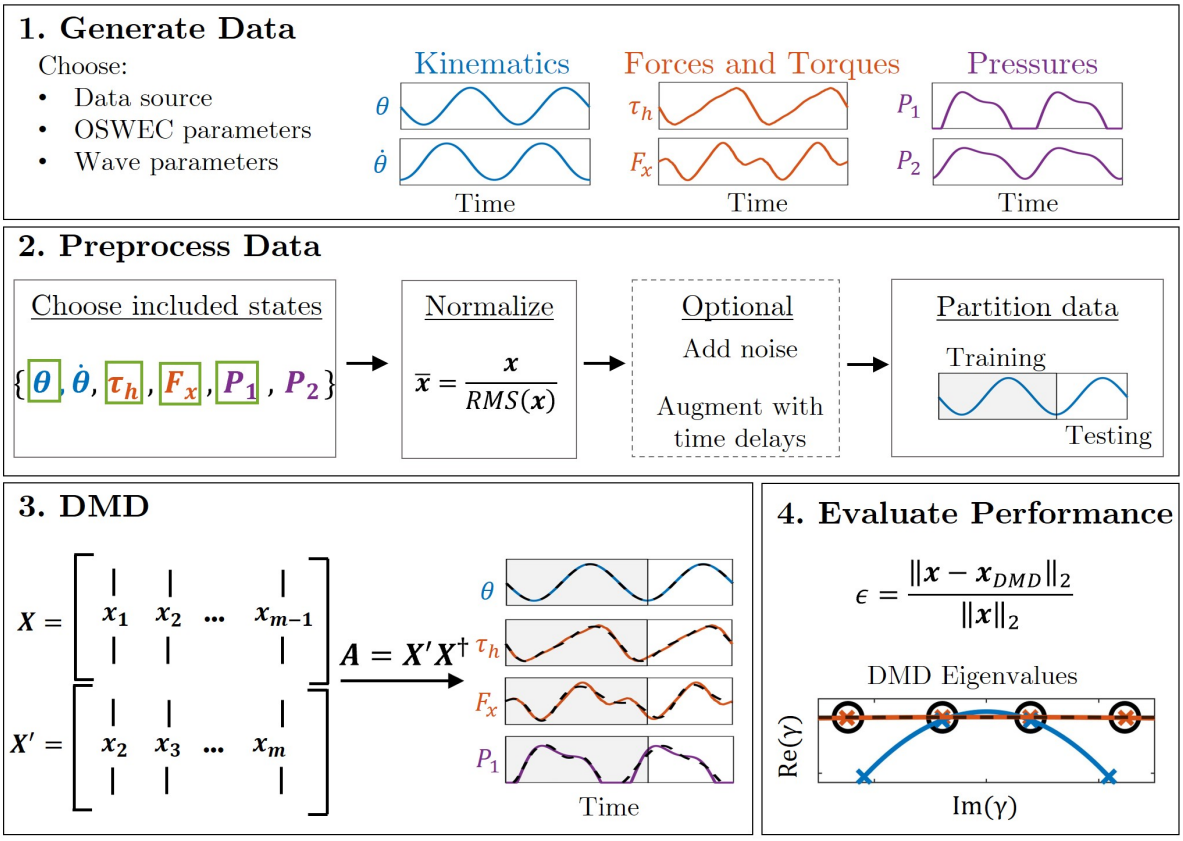

FIG. 2. Data analysis workflow. The first step is to generate data by choosing the data source (WEC-Sim, CFD, etc.) and wave input. Next, we preprocess data to prepare for DMD algorithm. We then run DMD on training data, and finally we evaluate the performance of the algorithm on both the training and testing data using a nondimensionalized error parameter, ϵ.

of a surface-piercing flap oscillating in the x-z plane with wave input, system states, and control input. For all of our data generation, we consider a utility-scale OSWEC analogous to the Aquamarine Oyster[6] with a flap width of 18 m operating in a still water depth of 10.9 m. Relevant OSWEC system dimensions and mass properties shown in gray in Figure 3 are enumerated in Table I, including moment of inertia about the flap center of mass (I) and hydrostatic stiffness (C) of the flap.

Figure 3 also shows the states of this system. These states represent time series measurements that describe system behavior and are known for the purposes of DMD (these form our data matrices in the DMD algorithm). We chose the states of this system to be quantities that can be realistically measured in OSWEC systems and have been measured in previous physical studies[52–54]. The kinematic states include angular position, $\boldsymbol{\theta}$, and an-

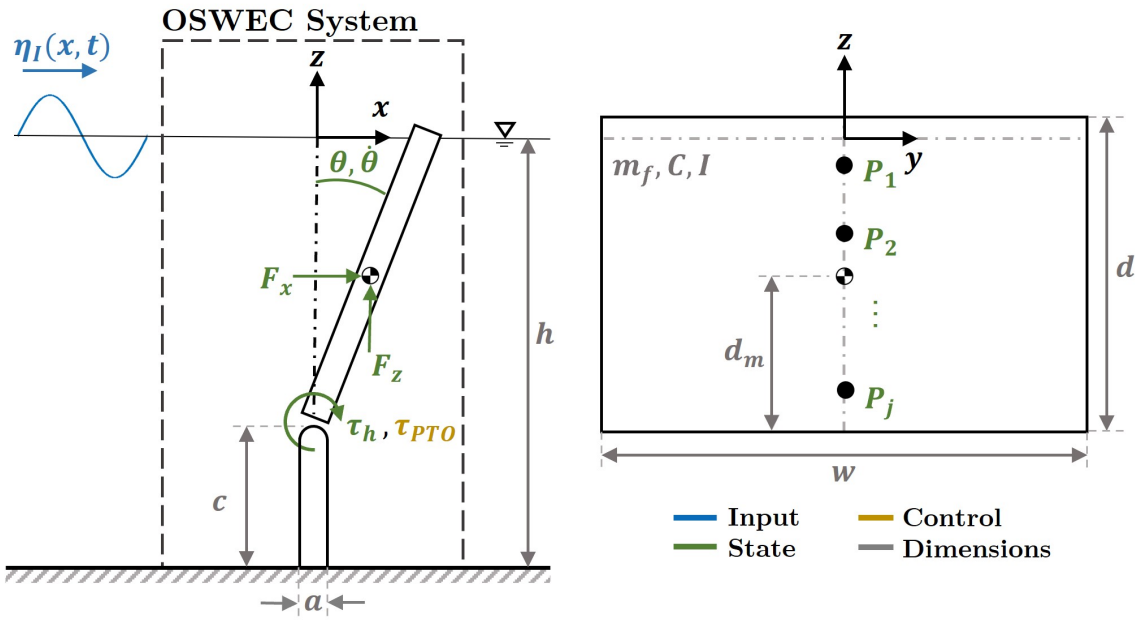

FIG. 3. Diagram of OSWEC system with incident wave input in blue, system states in green, control parameters in yellow, and dimensions in gray.

TABLE I. OSWEC system properties.

Parameter	Value
still water depth, h	10.9 m
flap width, w	18 m
flap height, d	9.4 m
flap thickness, a	1.8 m
foundation height, c	1.5 m
CoM to hinge distance, d_m	5 m
flap mass, m_f	127 * 10^3 kg
flap moment of inertia, I	1.85 * 10^6 kg m^2
flap hydrostatic stiffness, C	6.4 * 10^6 N m

gular velocity, $\dot{\boldsymbol{\theta}}$. Other dynamic states include the total surge and heave force acting on the flap, $\boldsymbol{F}_x$ and $\boldsymbol{F}_z$, respectively. These are quantities calculated from WEC-Sim directly and are the difference between the inertial force of the flap and the force required to constrain the device in the respective direction. We also consider the hydrodynamic torque acting

about the hinge, $\boldsymbol{\tau}_h = I\ddot{\boldsymbol{\theta}} - \boldsymbol{\tau}_{PTO}$, where $\boldsymbol{\tau}_{PTO}$ is the control torque applied by the power take-off (PTO). For this study we consider a simple linear damping control scheme, where $\boldsymbol{\tau}_{PTO} = -\nu_{PTO}\dot{\boldsymbol{\theta}}$, and ν_{PTO} is the linear damping coefficient (12000 N m s). Finally we consider pressure measurements, $\boldsymbol{P}_j$, from an array of sensors aligned with the midline of the flap in the cross-flow direction.

The input to the system is the incident wave field, $\boldsymbol{\eta}_I(\boldsymbol{x}, t)$. This wave field can be regular or irregular and is a function of both space and time. For the purposes of DMD, the incident wave field is an unknown quantity. This reflects the practical difficulty of obtaining time-resolved wave field measurements, which is a general challenge in WEC modeling and control[55].

1. WEC-Sim

WEC-Sim[50] is an open-source program implemented in MATLAB that solves the following six degree-of-freedom equation of motion in the time domain:

$$\boldsymbol{M}\ddot{\boldsymbol{y}}(t) = \boldsymbol{F}_e(t) + \boldsymbol{F}_{rad}(t) + \boldsymbol{F}_b(t) + \boldsymbol{F}_\nu(t) + \boldsymbol{F}_{md}(t) + \boldsymbol{F}_{PTO}(t), \tag{1}$$

where $\boldsymbol{M}$ is the mass matrix of the system, $\boldsymbol{F}_e$ is the excitation force vector, $\boldsymbol{F}_{rad}$ is the radiation force vector, $\boldsymbol{F}_b$ is the buoyancy force vector, $\boldsymbol{F}_\nu$ is damping force vector, $\boldsymbol{F}_{md}$ is the mean drift force vector, $\boldsymbol{F}_{PTO}$ is the control (PTO) force vector, and $\boldsymbol{y}$ is the six degree-of-freedom translation vector. WEC-Sim requires a separate BEM solver based on linear potential flow theory to determine the hydrodynamic parameters, including added mass/inertia, radiation damping, and excitation force/torque coefficients, then solves Equation 1 in the time domain using a six degree-of-freedom multi-body solver in Simulink (Simscape Multibody). Details of WEC-Sim development and validation, including a study on an OSWEC with the same physical and mass properties used here, can be found in Lawson et al.[51] and the source code is open access[50].

We utilize built-in WEC-Sim capabilities to model weakly nonlinear OSWEC dynamics in regular waves in Section III A and III B. Nonlinearity is introduced through the Froude-Krylov (FK) and restoring forces. There are two options for including nonlinearity in these terms. The first option uses the instantaneous body position, rather than the body's neutral position, to calculate the FK force and body restoring force. The second option uses both

the instantaneous body position and wave elevation to calculate both forces, increasing the nonlinearity. Lawson et al.[56] describes and validates this process. Nonlinearity must be included in the system for WEC-Sim to report pressure measurements, which is an important state variable for DMD. When pressure is reported, WEC-Sim outputs a linear (calculated with no displacement and mean water line) and nonlinear (calculated with the instantaneous displacement and free surface) wave pressure. In this work, we utilize the first nonlinearity option with linear wave pressure in Section III A and the second nonlinearity option and nonlinear wave pressure in Section III B. We note that significant nonlinear events such as overtopping and slamming cannot be modeled with WEC-Sim and are outside the scope of this work. WEC-Sim also allows for user input for incident wave elevation, which we use in Section III C where we input an irregular wave elevation time series collected from field data. Unfortunately, currently WEC-Sim does not report pressure when a time series is used as the wave input.

B. Preprocess data

The next step is to preprocess the data for the DMD algorithm (Figure 2, Block 2), beginning with choosing which measurements to include. For systems with limited sensor measurements, as is the case here, the number of states we include determines the rank of our data matrix (as described in Section II C), which must be large enough to resolve the system dynamics. Because of this, in cases with more complex dynamics, it may be necessary to include more of the available states (Figure 3) or employ a DMD variant with time delays (as described in Section II C 2). The number of states and which states to include may not be possible to know a priori, but is critical to DMD accuracy, so care must be taken to include enough dynamically-rich states.

Next, we normalize each state time series by its root mean square (RMS) value. In general, normalizing data is a common step in DMD studies to promote accuracy and stability[48,57,58]. For states with the same units (e.g., $\boldsymbol{F}_x$ and $\boldsymbol{F}_z$), we normalize each time series by the maximum RMS value of all of the states with those units. This normalization step scales the relative importance of states with similar units, and eliminates the significant difference in magnitude between states with different units.

The next steps are only used in specific test cases. The first option is to add artificial noise

to the state measurements, which we use in Section III A to test robustness. There, we add white, Gaussian noise with a defined signal-to-noise (SNR) ratio to approximate the noise present in physical systems. The second option is augmenting the available states using time delays[35] (Section II C 2), which we use for Section III B to model nonlinear hydrodynamics.

The final step of preprocessing is partitioning the state time series into training and testing regions. The training data is the input to the DMD algorithm to estimate $\boldsymbol{A}$, and the testing data is reserved to assess how well the reduced order model can predict future state behavior.

C. Dynamic mode decomposition

After preprocessing, we input the training data into the DMD algorithm (Figure 2, Block 3). This section includes a brief derivation of the exact DMD algorithm[26,27] with a detailed derivation included in Appendix A. DMD code is described in Kutz et al.[27] and is available at the accompanying website[59]. As previously mentioned, there are many variants to the DMD algorithm[27,36,40], some of which are also summarized here.

Consider a discrete-time dynamical system with state measurements sampled at a time step, Δt. The first step of the DMD algorithm is to arrange the preprocessed training data into data matrices of the form:

$$\boldsymbol{X} = \begin{bmatrix} | & | & & | \\ \boldsymbol{x}_1 & \boldsymbol{x}_2 & ... & \boldsymbol{x}_m \\ | & | & & | \end{bmatrix}, \quad \boldsymbol{X}^{'} = \begin{bmatrix} | & | & & | \\ \boldsymbol{x}_2 & \boldsymbol{x}_3 & ... & \boldsymbol{x}_{m+1} \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{nxm}, \tag{2}$$

where $\boldsymbol{x}_k$ is a column vector of all the states at time step t_k, and $k = 1, 2, ...m + 1$. In these data matrices, each state vector time series corresponds to one *row* of $\boldsymbol{X}$, while each *column* is a snapshot of *all* states at the associated time step, t_k. Note that the two data matrices $\boldsymbol{X}$ and $\boldsymbol{X}^{'}$ are shifted by a single time step. We can then define a best-fit linear operator, $\boldsymbol{A}$, that relates the time-evolution of our two data matrices:

$$\boldsymbol{X}^{'} \approx \boldsymbol{A}\boldsymbol{X}, \tag{3}$$

where $\boldsymbol{A}$ relates the states of the OSWEC system from one time step to the next. In other words, $\boldsymbol{A}$ represents the system dynamics, and if we can approximate $\boldsymbol{A}$, we can approximate past and future system behavior. To solve for $\boldsymbol{A}$, we use the pseudoinverse of data matrix

$\boldsymbol{X}$, denoted $\boldsymbol{X}^{\dagger}$, such that:

$$\boldsymbol{A} = \boldsymbol{X}^{'}\boldsymbol{X}^{\dagger}. \tag{4}$$

Because data matrices are often large, the singular value decomposition (SVD) is commonly used to calculate $\boldsymbol{X}^{\dagger}$, which allows for better scaling and numerical conditioning[27], and also allows for the choice of rank (i.e. how many singular values/SVD modes to include) when calculating $\boldsymbol{A}$ (Appendix A).

The eigenvectors of $\boldsymbol{A}$ are the DMD modes, $\boldsymbol{\phi}_j$, with corresponding discrete-time eigenvalues λ_j that represent their growth/decay rates and frequencies. Using the DMD eigenvectors and eigenvalues, we can describe the system dynamics:

$$\boldsymbol{x}(t) \approx \sum_{j=1}^{r} \boldsymbol{\phi_j} \exp(\gamma_j t) b_j = \boldsymbol{\Phi} \exp(\boldsymbol{\Gamma} t)\boldsymbol{b}, \tag{5}$$

where r is the SVD rank, the columns of $\boldsymbol{\Phi}$ contain the DMD modes $\boldsymbol{\phi}_j$, $\gamma_j = \ln(\lambda_j)/\Delta t$ are the complex continuous-time DMD eigenvalues, and $\boldsymbol{b} = \boldsymbol{\Phi}^{\dagger}\boldsymbol{x}_1$ are the DMD mode amplitudes with $\boldsymbol{x}_1$ being the states at the first time step, t_1. Note that $\boldsymbol{x}(t)$ contains the time series for all state variables. With Equation 5, we can now describe and predict system state dynamics using only snapshot data of our system. This approach is referred to as "exact" DMD and, while suitable for some dynamical systems, has limitations. For example, noise, nonlinearity, and sampling errors all can reduce the accuracy of DMD[60]. Even for a linear system with clean data, if the system is only partially measured (i.e., additional states are required to resolve the system dynamics), or there is a limited amount of training data, DMD may lead to inaccurate models. These limitations motivate a number of extensions to the exact DMD algorithm that can improve model accuracy.

1. Total-least-squares DMD

One of exact DMD's limitations is its sensitivity to noise[61,62]. In particular, noise biases eigenvalues in a way that can cause an artificial exponential decay in DMD dynamics[42,62]. These effects cannot be counteracted by increasing the duration of training data or sampling frequency[41]. To remedy this, Hemati et al. developed a modification to the DMD algorithm called total-least-squares DMD (TLS DMD) that uses a total-least-squares approach to solve the regression problem laid out in Equation 3, rather than the traditional least-squares method[41]. By doing this, TLS DMD solves for $\boldsymbol{A}$ assuming noise is present in both data

matrices, rather than just $\boldsymbol{X}$, as is the case with exact DMD[41]. The mathematical derivation of the TLS DMD algorithm is described in Appendix A 1 and we use this variation in Section III A. Code to implement this algorithm is described in Kutz et al.[27,59].

2. Time delays

Another variation of DMD involves the use of "time delays" (i.e., time delay coordinates or Hankel matrices), which augments the data matrices to include time-shifted copies of state measurements. There is a rich history of using time delays and Hankel matrices in system identification techniques, including the eigensystem realization algorithm (ERA)[63]. There are two ways that time delays can improve DMD performance. The first, which was discovered by Tu et al.[26], is that time-shifted matrices provide additional phase information that is critical to describing oscillatory dynamics. For example, DMD cannot model a standing wave without the addition of time-shifted data[26]. Second, including time delays increases the number of rows (and therefore the effective number of states) in $\boldsymbol{X}$, therefore increasing the number of SVD modes one can include when calculating $\boldsymbol{A}$. For systems with limited measurements, as is the case for this work, adding time delays increases the rank of data matrix $\boldsymbol{X}$, which can be necessary to resolve and recreate complicated dynamics. While the addition of time delays increases the size of the data matrices, because this system (and likely other WEC systems) has a relatively small number of sensor measurements, the additional cost of calculating linear operator $\boldsymbol{A}$ using the SVD is insignificant. Details on using time delays are provided in Appendix A 2 and we use this variation in Section III B.

3. Optimized DMD

The last variation of DMD we employ is optimized DMD (optDMD), as described in Askham and Kutz[64]. Instead of solving a regression problem for linear operator $\boldsymbol{A}$ using two time-shifted data matrices as shown in Equation 3 (i.e., exact DMD), optDMD aims to fit the data matrix $\boldsymbol{X}$ directly to a series of exponential functions, $\exp(\gamma_j t)$, where γ are the continuous-time DMD eigenvalues. The algorithm then uses variable projection techniques to efficiently solve the resulting nonlinear least squares problem and determine DMD eigenvalues, γj, eigenvectors, $\boldsymbol{\phi}_j$, and weights, b_j, simultaneously. One main difference

between optDMD and exact DMD is optDMD simultaneously optimizes DMD modes and amplitudes using the full data set, $\boldsymbol{X}$, rather than finding the DMD modes, and then solving for DMD mode amplitudes using only the first data snapshot like exact DMD (Equation 5). As a result, optDMD provides a model that can handle non-ideal data and has less bias for noisy inputs. This algorithm also allows for the addition of constraints on DMD eigenvalues that restrict the real part of the continuous-time eigenvalues to be approximately zero, which limits the artificial temporal decay and promotes stability in the DMD reconstruction. We use optDMD in Section III C to model OSWEC behavior in response to irregular waves and restrict the real part of the eigenvalue to be between -0.001 and 0, to prevent spurious decay or growth in the DMD modes. We provide a brief derivation of optDMD in Appendix A 3, and refer the reader to Askham and Kutz[64] for a full derivation, as well as open-source code.

D. Evaluate performance

The last step in the workflow (Figure 2, Block 4) is to evaluate the performance of the DMD model. For each case, the DMD model is initialized using the first time step of the training data, then propagated through both the training and testing region using Equation 5. We evaluate the accuracy with which the DMD model describes the training data (hindcasting) and predicts the testing data (forecasting) using a normalized error parameter ϵ for each state variable, $\boldsymbol{\psi}$, corresponding to one row in $\boldsymbol{X}$. We define ϵ as the ratio between the L_2 norm of the difference between the DMD output, $\boldsymbol{\psi}_{DMD}$, and the true state variable $\boldsymbol{\psi}$, normalized by the L_2 norm of $\boldsymbol{\psi}$:

$$\epsilon = \frac{||\boldsymbol{\psi} - \boldsymbol{\psi}_{DMD}||_2}{||\boldsymbol{\psi}||_2}. \tag{6}$$

This error parameter can be applied to the training or testing region.

E. Test cases

We consider three cases to evaluate how well DMD can model realistic OSWEC behavior. Analogous considerations apply to other WEC archetypes. The relevant parameters used in each case are outlined in Table II E.

The first case evaluates how well DMD can model OSWEC dynamics when noise is present in the training data. Because any real WEC system will have some level of sensor noise, it

is critical that our modeling method can account for that. For this case, we utilize the first nonlinearity option in WEC-Sim (weakly nonlinear restoring and Froude-Krylov force based on flap position) to generate the training data and a polychromatic wave input composed of two wave components. The first wave component emulates a "swell" wave, with a longer period and larger wave height ($T_{swell} = 8$ s, $H_{swell} = 0.75$ m) while the second emulates a "wind" wave with a shorter period and smaller wave height ($T_{wind} = 2.6$ s, $H_{wind} = 0.3$ m), as outlined in Table II E. Under these conditions, the maximum flap oscillation amplitude is about 6 degrees, resulting in effectively linear dynamics. We add Gaussian, white noise to the data after normalization (Figure 2, Block 2) at different signal-to-noise ratios (SNRs) and evaluate the accuracy of exact DMD and total-least-squares DMD, given this input. We use 32 seconds of training data ($4T_{swell}$), and consider a testing region of the same length. Finally, we include states $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\boldsymbol{\tau}_h$, and three linear pressure sensors (Table II E).

The second case evaluates how well DMD, a linear algorithm, can model weakly nonlinear dynamics. We use the second nonlinearity option in WEC-Sim (restoring and Froude-Krylov force based on flap position and water surface elevation) to generate the training data and evaluate the accuracy of exact DMD with and without time delays. For this case, we consider a significantly larger wave height than Case 1 and the same wave period (Table II E), which results in a maximum flap oscillation amplitude of about 23 degrees. We use 10 seconds of training data and consider a 30 second testing region. We include states $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\boldsymbol{\tau}_h$, $\boldsymbol{F}_x$, and three nonlinear pressure sensors. We include surge force as a state variable because it exhibits interesting nonlinear behavior that could be challenging to model.

For the third case, we evaluate how well DMD can model OSWEC behavior in response to an irregular wave forcing. We use a wave elevation time series collected from field data as input into WEC-Sim and evaluate the linear response of the OSWEC. Unlike the other cases, we calculate the power spectra of our state variables over time and use DMD to model the temporal behavior of the power spectra, rather than the time evolution of the state variables. We expect the resulting dynamics to be more complex and high-dimensional than the previous two cases, which can result in inaccurate results from exact DMD, especially with a limited number of system states (Section II C). Because of this, we compare the performance of exact DMD to optDMD. In addition, because of the increased complexity we use a significantly longer training time of 276 seconds ($35T_p$), and reserve the remainder of the time series (just over 100 seconds) as testing data (Table II E). Finally, we include

TABLE II. Summary of data parameters and algorithms used for the three cases. DMD refers to the exact DMD algorithm and serves to contrast with variants including total-least-squares DMD (TLS DMD), DMD with time delay (DMD+TD), and optimized DMD (optDMD).

	Case 1	**Case 2**	**Case 3**
Condition	Measurement noise	Nonlinearity	Irregular waves
Wave type	Polychromatic	Monochromatic	Irregular
Wave parameters	$H_{swell} = 0.75$ m, $H_{wind} = 0.3$ m $T_{swell} = 8$ s, $T_{wind} = 2.55$ s	H = 2 m T = 8 s	$H_s = 0.9$ m $T_p = 5.8$ s
Algorithms	exact DMD TLS DMD	exact DMD DMD+TD	exact DMD optDMD
Training time	32 s ($4T_{swell}$)	10 s	276 s ($35T_p$)
Testing time	32 s	30 s	102 s
States	$\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\boldsymbol{\tau}_h$, $\boldsymbol{P}_1 - \boldsymbol{P}_3$	$\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\boldsymbol{\tau}_h$, $\boldsymbol{F}_x$, $\boldsymbol{P}_1 - \boldsymbol{P}_3$	$\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\boldsymbol{\tau}_h$, $\boldsymbol{F}_x$

only four states: $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\boldsymbol{\tau}_h$, and $\boldsymbol{F}_x$. Because WEC-Sim does not currently report pressure outputs for tests with a user-defined wave elevation, pressure measurements are not included as state variables.

In summary, we chose these three cases as examples not only because they specifically address common conditions for WEC modeling and control, but also because these are cases where we expect the exact DMD algorithm to perform poorly. We then demonstrate that the different extensions of DMD can overcome each of these complications (i.e., noise, nonlinearity, irregular wave forcing). By highlighting these cases, we show that DMD can be an effective tool at modeling realistic WEC dynamics.

III. RESULTS AND DISCUSSION

A. Case 1: Noisy measurements

Figure 4 summarizes exact and TLS DMD performance on OSWEC data augmented with white, Gaussian noise (e.g., broadband electromagnetic interference). The top right panel shows normalized error, ϵ, for hydrodynamic torque, $\boldsymbol{\tau}_h$, as a function of SNR. We only report error for $\boldsymbol{\tau}_h$, however we observe these trends for all state variables. Although

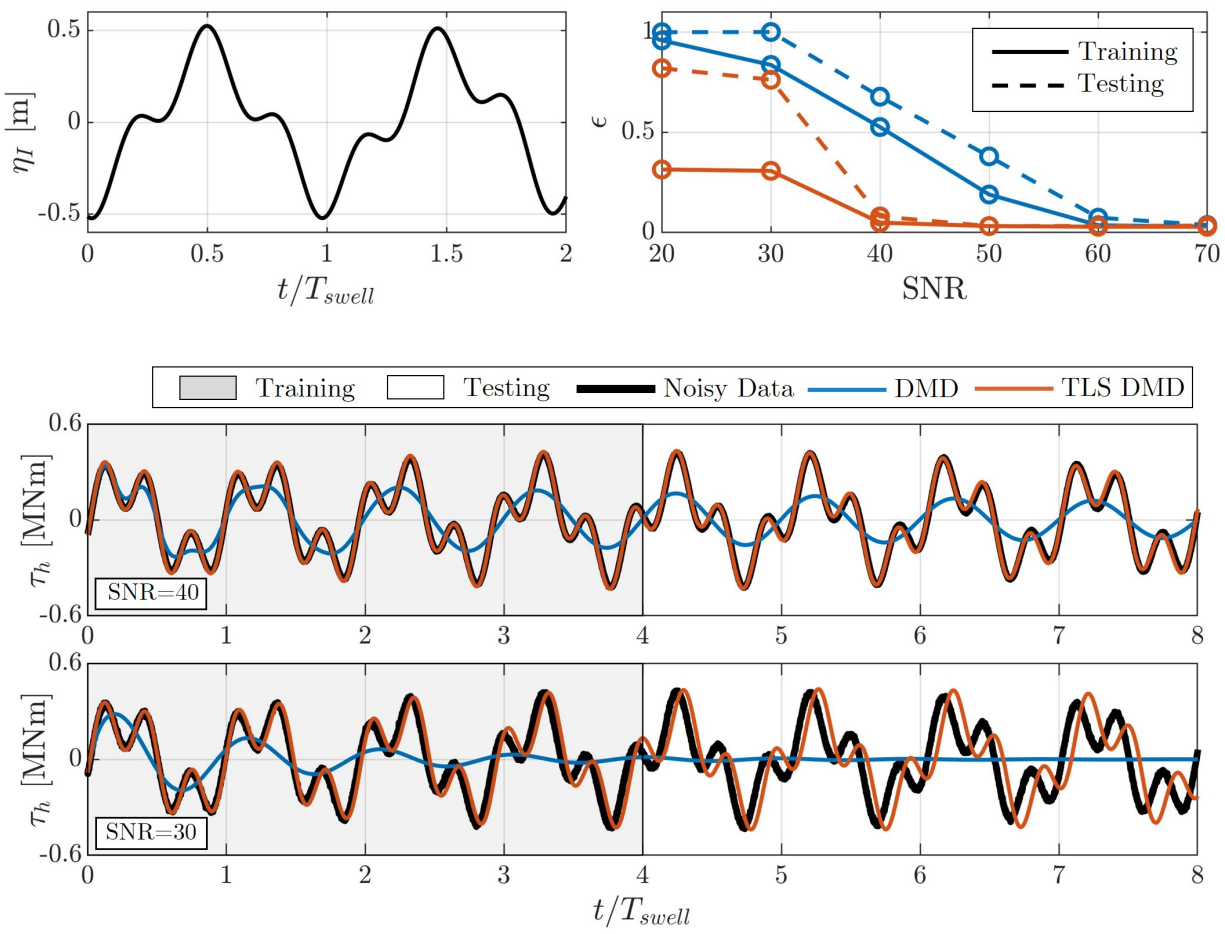

FIG. 4. (top left) Undisturbed incident wave field at the flap over time normalized by the wave period of the swell wave component, T_{swell}. (top right) Normalized error of DMD output, ϵ, for state variable $\boldsymbol{\tau}_h$ as a function of SNR. Lower values of SNR correspond to a noisier signal. Dashed lines represent errors in the training region (hindcasting) and solid lines represent error in the testing region (forecasting). As SNR decreases, TLS DMD (orange) consistently outperforms exact DMD (blue) in both the testing and training region. (bottom panels) Hydrodynamic torque from WEC-Sim (black), exact DMD (blue), and TLS DMD (orange) for SNR value of 40 (top) and SNR of 30 (bottom). The time series show the artificial decay in the DMD model, but much better accuracy for TLS DMD, consistent with the normalized error values.

error rises monotonically with increasing noise for both training and testing data, TLS DMD consistently outperforms exact DMD. For SNR < 40, the testing error for TLS DMD approaches the error for exact DMD, however the source of the errors are different. Time series of hydrodynamic torque, $\boldsymbol{\tau}_h$ with an SNR of 40 and 30 show that exact DMD predic-

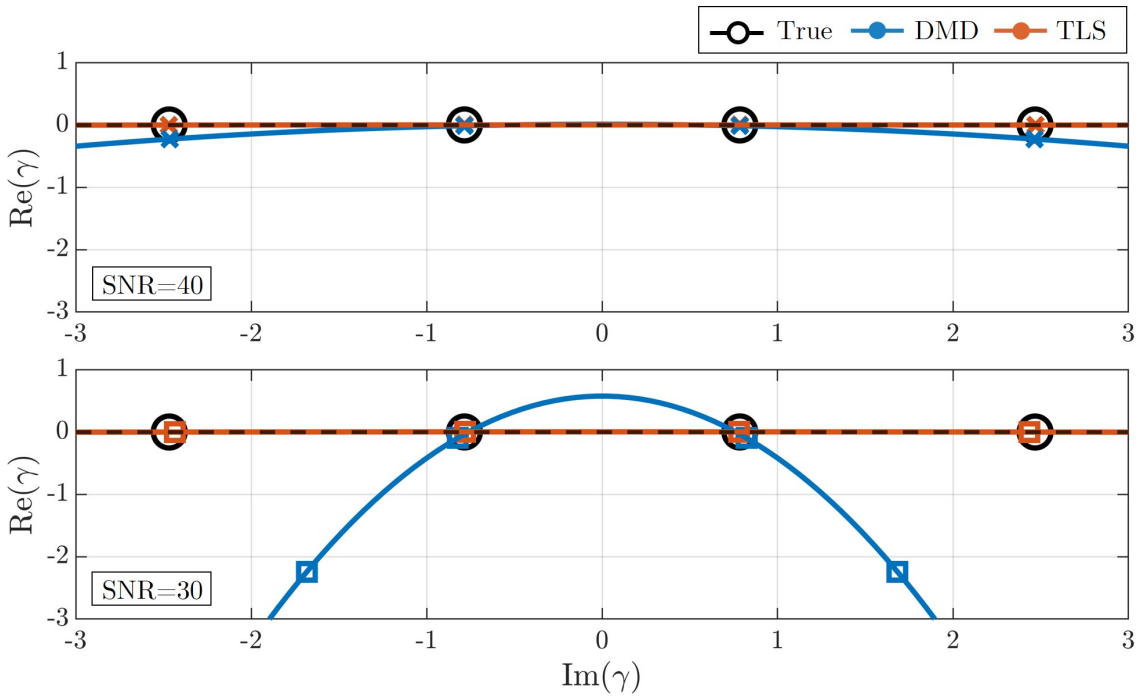

FIG. 5. Continuous-time DMD eigenvalues, γ, on the complex plane with best-fit parabolic curves. High curvature of the eigenvalues represent a decrease in the real part of the eigenvalues, which in turn results in higher exponential decay in the time series forecasts[62]. Black open circles are the true DMD eigenvalues of the system, the blue symbols represent exact DMD, and the orange symbols represent TLS DMD. The x markers represent the SNR=40 case (top panel) and the square markers represent the SNR=30 case (bottom panel). As SNR increases, the curvature of the best-fit lines for exact DMD increases significantly, which corresponds to the exponential decay seen in the two time series in Figure 4. In contrast, the best-fit curve for the TLS DMD eigenvalues (orange) show almost no curvature, which corresponds to significantly less error in the time series predictions.

tions artificially decay through the test region (Figure 4, bottom panels). This is common behavior of DMD models trained on noisy data[62]. However, TLS DMD is able to capture higher frequency oscillations from wind waves without significant decay. The TLS prediction accuracy does decrease over time, which is most notable in the SNR = 30 case, but this is caused by an increasing phase shift rather than an exponential decay like the exact DMD model.

For further insight into the different performance of exact and TLS DMD, we consider

the continuous-time DMD eigenvalues, $\gamma_j \in \mathbb{C}$, in the complex plane (Figure 5). Because the system is oscillatory and approximately linear, the "true" eigenvalues are purely imaginary conjugate pairs whose components correspond to the two frequencies included in the training data. As a result, we chose to include only four SVD modes in the DMD reconstruction, which results in four eigenvalues for exact and TLS DMD (Figure 5.) Bagheri[62] showed that adding noise to a system decreases the real component of the DMD eigenvalues and this effect is amplified for higher harmonics of the eigenvalue frequency. This results in a parabolic form of DMD eigenvalues in the complex plane, with higher curvature associated with higher noise levels. In addition, a negative real part of the DMD eigenvalue results in an exponential decay in the time dynamics, which can be anticipated from Equation 5. Figure 5 visualizes this pattern for SNR=40 and SNR=30. As expected, we see a significant increase in curvature with noise amplitude (i.e., SNR decrease), but we also see the dramatic reduction in curvature of the TLS DMD eigenvalues for both noise amplitudes. This means TLS DMD has significantly less spurious decay in time and models noisy data better than exact DMD, which is evident by the time series models in Figure 4. There is slight discrepancy in the higher frequency component for the "true" and TLS DMD eigenvalue for the SNR=30 case, which manifests as the TLS DMD model oscillating at a slightly different frequency than the true dynamics, causing a growing phase error in the time domain (Figure 4). While error introduced by noise cannot be remedied with increasing amounts of data[41], this discrepancy can be improved by increasing the length of training data. For example, doubling the training time reduces the testing error for TLS DMD at SNR = 30 to from 0.76 to 0.17, while the testing error for exact DMD remains at 1.

These results show that while exact DMD may not be suitable for systems with even low levels of noise, TLS DMD is an effective tool for modeling noisy data and can substantially improve the accuracy of the models in both the training and testing regions. This is promising because it extends the use of DMD to include training on non-ideal experimental or field-scale measurements, with only a slight modification to the original DMD algorithm. For substantially higher noise amplitudes, a different DMD variant, such as optDMD[64], may be more accurate, albeit with a slightly higher computational cost.

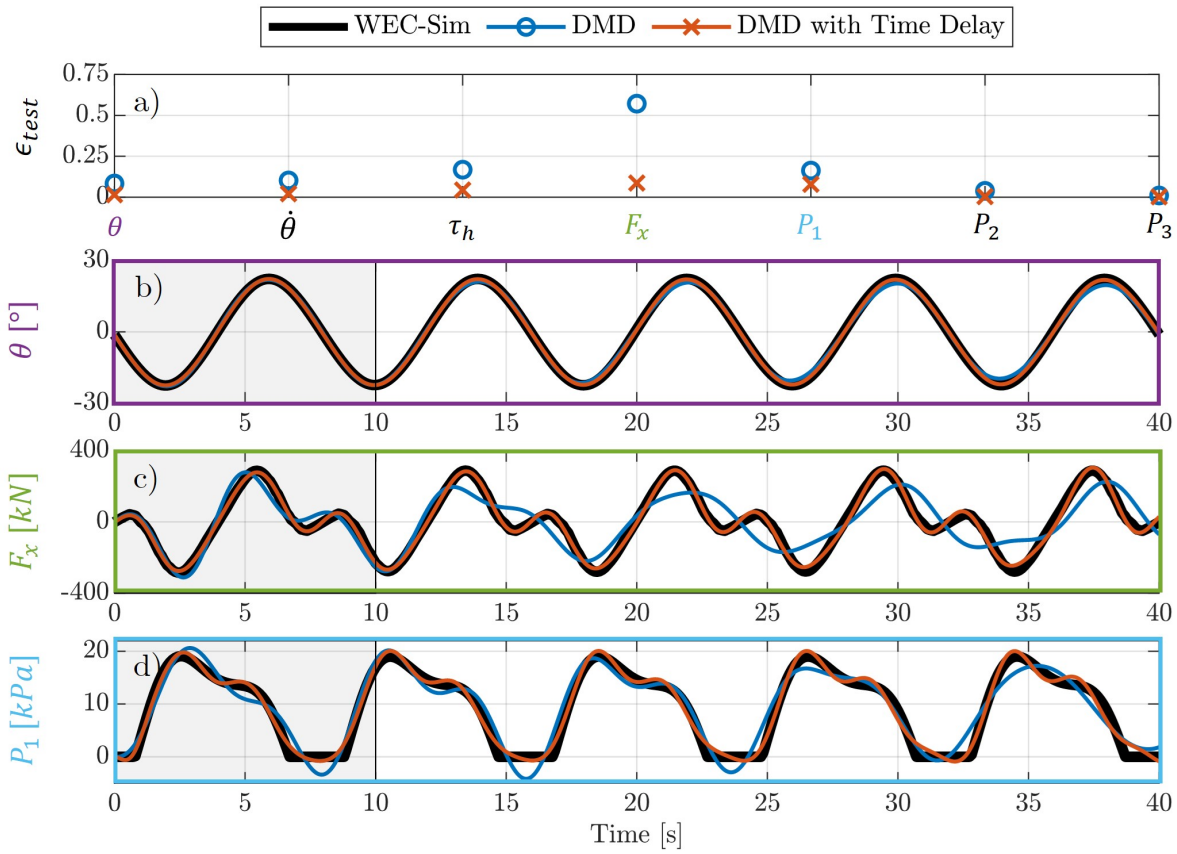

FIG. 6. (a) Normalized testing error, ϵ_{test}, as a function of system states, showing that DMD with time delay (orange crosses) outperforms exact DMD (blue circles) for all states with nonzero error. Time series of (b) angular rotation, $\boldsymbol{\theta}$, (c) surge force, $\boldsymbol{F}_x$, and (d) the top pressure sensor, $\boldsymbol{P}_1$. WEC-Sim output is in black, the exact DMD fit is in blue, the TLS DMD output is in orange, and the training region is highlighted in gray. By adding a time delay, the model captures higher-order oscillations with better accuracy for $\boldsymbol{F}_x$ and $\boldsymbol{P}_1$, and performs better over time in the testing region for all three states shown.

B. Case 2: Nonlinear behavior

Next we explore the potential for DMD, a linear algorithm, to model weakly nonlinear dynamics. Although DMD uses linear least-squares regression, its relation to the Koopman operator means it can model nonlinear systems with an appropriate choice of state variables[25,29,36–38]. Here, we summarize the ability of exact and time-delayed DMD to model weakly nonlinear OSWEC dynamics in response to a regular wave.

Figure 6a shows the normalized error for the testing period, ϵ_{test}, as a function of system

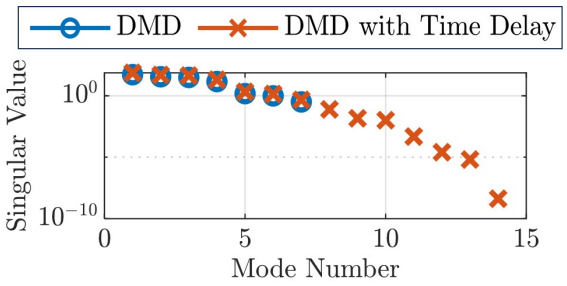

FIG. 7. Singular values of data matrix $\boldsymbol{X}$ from exact DMD (blue circles) and DMD with one time delay (orange crosses). The singular values for the time delay case show that the true system is a higher rank (contains important information on dynamics in higher modes) than can be captured with exact DMD.

states for DMD with and without a time delay. Some states have near zero error for both DMD models (such as $\boldsymbol{P}_2$ and $\boldsymbol{P}_3$), but in every case with nonzero error, DMD with time delay outperforms exact DMD. To contextualize these errors, we consider time series for three states with varying error: angular position ($\boldsymbol{\theta}$), surge force ($\boldsymbol{F}_x$), and the top pressure sensor ($\boldsymbol{P}_1$). For $\boldsymbol{\theta}$ (Figure 6b), both DMD with and without a time delay model the time series well, but even in this case where the dynamics appear linear and sinusoidal, there is a slight decay in the exact DMD result without a time delay, meaning we are losing accuracy as we make predictions further into the future. However, adding a time delay eliminates that decay, resulting in near-zero testing error (Figure 6a). Surge force, $\boldsymbol{F}_x$, contains more complicated dynamics (Figure 6c) and is most improved with a time delay. Pressure, $\boldsymbol{P}_1$, shows a similar trend (Figure 6d). These measurements have no negative values, leading to a piece-wise behavior that can be difficult to model. Both DMD with and without a time delay are able to model the general dynamics, the time delay improves prediction accuracy, particularly later in the testing period.

As previously mentioned in Section II C 2, there are two ways time delays improve performance relative to exact DMD. The first is by providing necessary phase information for oscillatory systems[26], and the second is increasing the available rank of the data matrix to better resolve complex dynamics. To illustrate the latter, we consider the singular values of data matrix $\boldsymbol{X}$ with and without a time delay (Figure 7). Because we used a full rank SVD, the number of singular values for each model corresponds to the number of rows in the data matrix $\boldsymbol{X}$, which is doubled when adding a time delay. For a system with enough states to

fully resolve the dynamics (i.e., the rank of data matrix $\boldsymbol{X}$ is larger than the rank of the underlying dynamics), there is often a steep drop in singular values indicating the number of SVD modes required for accurate modeling. However, the singular values without a time delay decay slowly and do not have a clear cutoff. Because the additional singular values available with a time delay have a similar magnitude, they provide additional, important information about the dynamics that are otherwise missed. This allows for inclusion of all dynamically-significant SVD modes when calculating $\boldsymbol{A}$ and explains why adding a time delay increases the accuracy of the DMD model for states with meaningful nonlinearity (e.g., $\boldsymbol{F}_x$, $\boldsymbol{P}_1$).

These results demonstrate that adding a time delay can significantly increase the performance of DMD for weakly nonlinear systems. Even in cases where DMD provides a reasonably good description of the training data, adding a time delay can increase accuracy of higher-order nonlinear patterns well into the testing region. We note that similar improvement to exact DMD may be possible by either increasing the duration of training data or including additional states. For example, when we increase the training duration from 10 to 40 seconds and consider the same amount of testing data, ϵ_{test} for $\boldsymbol{F}_x$ drops from 0.57 to 0.21. However, this the value is still more than double the error from DMD with a time delay trained on only 10 seconds of data ($\epsilon_{test} = 0.09$). Therefore, time delays offer a computationally efficient way to improve accuracy that can be more effective than increasing the duration of training data.

C. Case 3: Irregular wave input

For this final case, we investigate how well DMD can model OSWEC behavior in response to real wave elevation data from field measurements. Modeling WEC behavior in irregular waves raises a range of potential complications, as the process can be nonlinear, nonstationary, and stochastic. This means that with limited sensor measurements and a broad frequency range of incident waves, DMD may not be able to consistently model time-domain dynamics of the state variables with a reasonable number of sensors and training time, even using time delays, due to the rank of the dynamics being significantly larger than the rank of state measurements.

Because of this, rather than using DMD to model and predict magnitudes of state vari-

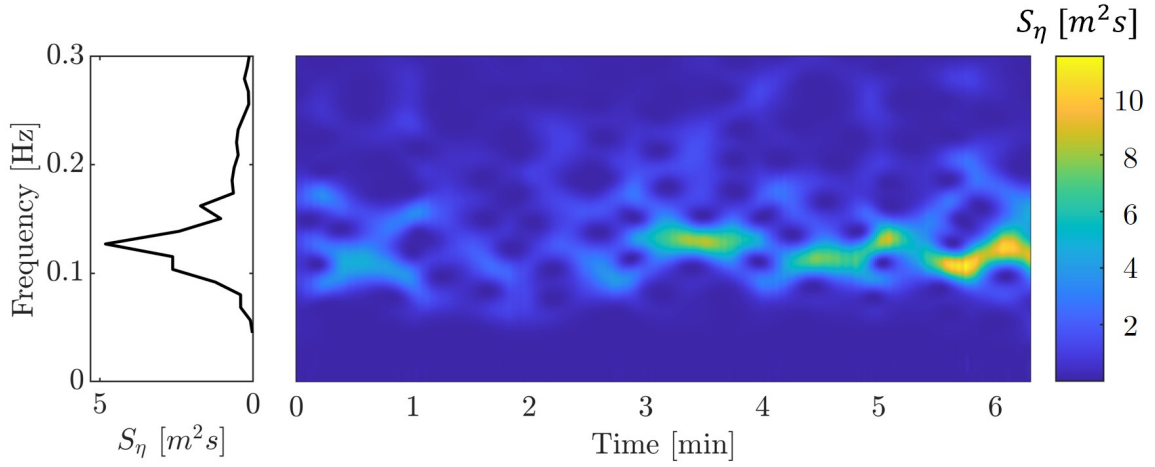

FIG. 8. Periodogram (left) and spectrogram (right) of the incident wave elevation time series used for Case 3. The time series was measured from field data[65] and used as input to WEC-Sim. To generate the spectrogram, we use a sliding window short-time Fourier transform on detrended and zero-padded time series data using a Hann filter (window width of 60 seconds and a window time step of 1 second). We note that parameters such as filter type, window width, and window time step are tunable and may affect the accuracy of the DMD model.

ables, as in Cases 1 and 2, we instead use DMD to model and predict the time behavior of the power spectral distributions (PSDs) of the state variables. The power spectrum of state $\boldsymbol{\psi}$ over a specified time window T_w is described as:

$$\boldsymbol{S}_{\boldsymbol{\psi}} = \frac{2\,|\mathrm{FFT}(\boldsymbol{\psi}')|^2}{m\,f_s}, \tag{7}$$

where $\boldsymbol{\psi}'$ is the windowed and detrended time signal of state $\boldsymbol{\psi}$, m is the number of time steps in the windowed signal, and f_s is the sampling frequency. We apply a 60-second sliding window Hann filter to the state variable time series, such that $\boldsymbol{S}_{\boldsymbol{\psi}}$ represents how the frequency content of the state varies in time.

For this case, WEC-Sim is driven by wave elevation time series measured by SWIFT drifters[65] off the east coast of O'ahu. The frequency distribution is relatively narrow, with a peak period of 7.9 s and a significant wave height of 2.1 m (Figure 8, left). The frequency content evolves over time (Figure 8, right) which we expect will translate to complex OSWEC dynamics. Our goal is to recreate the spectrograms of state variables in response to this

wave field to model and predict their spectro-temporal behavior. Similar to Case 1 and 2, we partition the spectrogram for each state variable into training and testing periods (Table II E).

Since DMD was developed for spatio-temporal data, this dimensional expansion to spectro-temporal data is readily accommodated, but requires a minor modification to the data matrices $\boldsymbol{X}$ and $\boldsymbol{X}^{'}$. Specifically, we stack the frequency data of each state in a single column at the corresponding time step. For example, if the state variables are $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$, the data matrix $\boldsymbol{X}$ is arranged as:

$$\boldsymbol{X} = \begin{bmatrix} | & | & & | \\ \boldsymbol{S}_{\boldsymbol{\theta},1} & \boldsymbol{S}_{\boldsymbol{\theta},2} & \dots & \boldsymbol{S}_{\boldsymbol{\theta},m} \\ | & | & & | \\ & & & \\ | & | & & | \\ \boldsymbol{S}_{\dot{\boldsymbol{\theta}},1} & \boldsymbol{S}_{\dot{\boldsymbol{\theta}},2} & \dots & \boldsymbol{S}_{\dot{\boldsymbol{\theta}},m} \\ | & | & & | \end{bmatrix}, \tag{8}$$

where $\boldsymbol{S}_{\boldsymbol{\theta},1}$ would correspond to the frequencies of state variable $\boldsymbol{\theta}$ at t_1, $\boldsymbol{S}_{\boldsymbol{\theta},2}$ at t_2, and so on. The rest of the DMD process is unchanged. Here, we contrast the performance of exact DMD with optDMD (Section II C 3) and use a reconstruction rank of 17 modes to model the state variable spectrograms. We compare their performance with a normalized error metric, $\bar{\boldsymbol{\epsilon}}$, defined as:

$$\bar{\boldsymbol{\epsilon}} = \frac{\boldsymbol{S}_{\boldsymbol{\psi}} - \boldsymbol{S}_{\boldsymbol{\psi},DMD}}{\max(\boldsymbol{S}_{\boldsymbol{\psi}})}, \tag{9}$$

where $\max(\boldsymbol{S}_{\boldsymbol{\psi}})$ is the maximum value of the true spectrogram of that state. Unlike the error parameter used in the previous two cases, $\bar{\boldsymbol{\epsilon}}$ is a matrix of values the same size as the state spectrograms, not a scalar value.

Figure 9 shows the DMD and optDMD models of the spectrograms for state variables $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\boldsymbol{\tau}_h$, and $\boldsymbol{F}_x$. Each spectrogram is normalized by its maximum value. Overall, optDMD outperforms exact DMD in the training region. Exact DMD models the beginning of the training data with reasonable accuracy, but quickly degrades after about one minute. In contrast, optDMD is able to model the entire training region accurately, capturing the changing spectral behavior with low error. While neither model can accurately predict dynamics in the testing region, the structure of the two models vary considerably. Exact DMD only

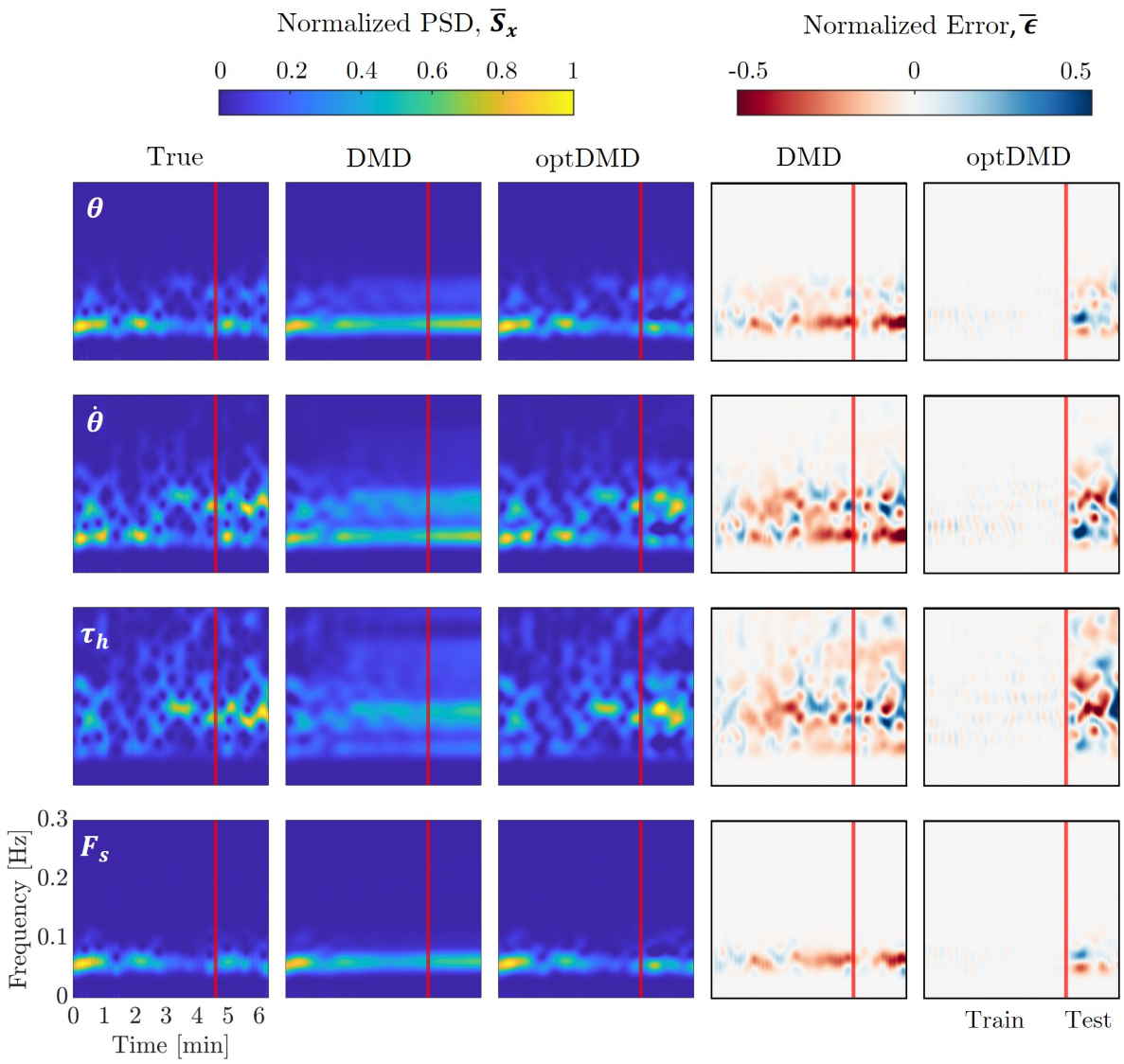

FIG. 9. Spectrograms of WEC-Sim output ("True") (first column), exact DMD (second column), and optDMD (third column) for state variables $\boldsymbol{\theta}$ (top row), $\dot{\boldsymbol{\theta}}$ (second row), $\boldsymbol{\tau}_h$ (third row), and $\boldsymbol{F}_x$ (bottom row). All spectrograms are normalized by the maximum value of the true spectrogram for the corresponding state variable. The last two columns represent the normalized error for exact DMD (fourth column) and optDMD (fifth column). optDMD outperforms exact DMD in the training region for all state variables, which ends at the red vertical line, while both models have significant error in the testing region.

shows low-resolution spectral characteristics and does not capture any underlying structure present in the state variables. However, optDMD models some of the general trends and structure present in the testing region. For example, optDMD predicts the spectral behavior well for both angular position, $\boldsymbol{\theta}$, and the surge force, $\boldsymbol{F}_x$. For velocity and hydrodynamic

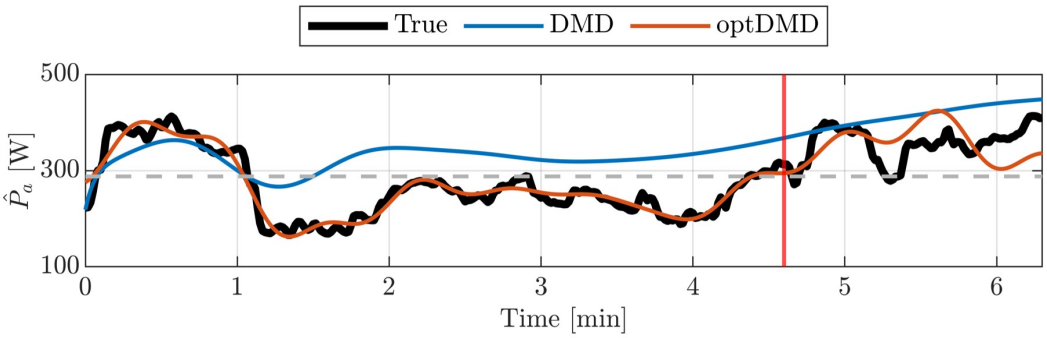

FIG. 10. Time series plot of average absorbed power, $\hat{P}_a(t)$, with true value in black, the DMD fit in blue, and optDMD in orange. Red line separates training region (left) and testing region (right), and the gray dashed line represents the true average absorbed power over the full time series.

torque, where the spectral behavior is more complex, optDMD identifies energetic peaks at similar frequencies and times to the true dynamics, even with no knowledge of the incident wave field.

One reason optDMD outperforms exact DMD is optDMD solves for the DMD mode amplitudes, b_j, using the full time series data contained in $\boldsymbol{X}$, whereas exact DMD uses only the first snapshot, $\boldsymbol{x}_1$ (Equation 5). This results in the exact DMD fit to only match the initial training data, whereas optDMD can accurately model the entire training region. We also enforced the optDMD eigenvalues to have nearly no real component to eliminate any growth or decay in time. This means that optDMD is more robust to non-ideal data (e.g., limited training data, small nonlinearities, or numerical errors) all of which can bias DMD eigenvalues and lead to non-physical growth or decay. This also contributes to the decay of spectral resolution of the exact DMD model in time that is not present with optDMD.

To further evaluate the performance of exact DMD and optDMD, we consider how well these algorithms can model the average power the OSWEC generates over time, or absorbed power. Since the device is operating in irregular seas, the power absorbed can vary significantly in time, making electricity generation and storage difficult. Average power is important to know and predict because it can inform when the device is generating more power than necessary and needs to shed power to accommodate energy storage constraints, or when energy generation is low and the device can operate at full efficiency. Because we cannot model the actual velocity time series to calculate instantaneous power absorption,

we instead look at the average power absorbed by the OSWEC over all frequencies, $\hat{\boldsymbol{P}}_a(t)$. Because the OSWEC model uses a linear PTO, the average power absorbed over a time period, T_w is $\hat{\boldsymbol{P}}_a(t) = (\nu_{PTO}/T_w)\int_t^{t+T_w} \left|\dot{\boldsymbol{\theta}}(\tau)\right|^2 d\tau$. By Parseval's theorem, we can approximate $\hat{\boldsymbol{P}}_a(t)$ by taking each "slice" of the $\dot{\boldsymbol{\theta}}$ spectrogram in time and integrating the power spectral density over all frequencies, as:

$$\hat{\boldsymbol{P}}_a(t) = \nu_{PTO}\int_f \boldsymbol{S}_{\dot{\boldsymbol{\theta}}}(f,t)df. \tag{10}$$

Figure 10 shows $\hat{\boldsymbol{P}}_a(t)$ calculated from the true spectrograms generated from WEC-Sim data as well as both exact DMD and optDMD spectrograms from Figure 9. Similar to the previous results, optDMD outperforms exact DMD in modeling the average absorbed power, especially in the training region where optDMD models $\hat{\boldsymbol{P}}_a(t)$ with near perfect accuracy. The exact DMD estimate of $\hat{\boldsymbol{P}}_a$ follows a similar trend as the true value, but does not accurately model its magnitude beyond the first minute of training data. This suggests that DMD can only identify the low rank dynamics of $\boldsymbol{S}_{\dot{\boldsymbol{\theta}}}$, which is also evident in Figure 9. While optDMD matches the general trend of the true values in the testing region, neither algorithm accurately forecasts the time evolution of absorbed power consistently in the testing region.

Even though optDMD shows improvement over exact DMD in modeling and predicting OSWEC behavior in irregular waves, there are limitations to these algorithms that could make them unfit in practice. OptDMD could be helpful for state identification in irregular seas, but is less helpful for state forecasting. We note that there are additional adjustments we can make to optDMD hyperparameters, such as adjusting the initial guess of the eigenvalues or optimization parameters, that could improve its accuracy. However it is possible that this algorithm may not be suitable for real-time state prediction. Extended DMD[66], which was used by Jia et al. to build a model predictive controller for a WEC in irregular seas[43] and incorporated wave elevation data directly, could be better fit for this problem and should be explored in future work.

IV. CONCLUSION

This study demonstrates the accuracy with which dynamic mode decomposition can model and predict OSWEC behavior using only physically attainable measurement data

from the system dynamics and without knowledge of the past or future incident wave field. Specifically, we tested the ability of DMD to model OSWEC behavior when considering three common WEC modeling considerations: noisy signals, nonlinear behavior, and irregular wave input. We showed that for noisy data, exact DMD breaks down with relatively low levels of noise (SNR < 50), but total-least-squares DMD can significantly improve the system model in both the training and testing region with minimal modifications to the original algorithm. Second, we showed that although DMD is a linear algorithm, it can model weakly nonlinear dynamics with reasonable accuracy. When nonlinear dynamics are present, adding even a single time delay can significantly improve DMD accuracy with the same amount of training time by increasing the available rank of the data matrices and providing additional phase information. Finally, we showed we can describe OSWEC behavior in response to irregular waves by transforming the state variables to the frequency domain and modeling the spectro-temporal behavior of the system states. We showed that optimized DMD can significantly improve accuracy in recreating the spectro-temporal structure of system states, particularly in the training region. Analysis of the irregular wave case also highlights the limitations of a predictive model without knowledge of the future, stochastic wave field.

Overall, we see potential for data-driven algorithms, like DMD, to bridge the gap between the wave energy and data science fields and solve problems encountered in WEC modeling and control. In this study, we highlight the capabilities of DMD to accurately model and predict WEC dynamics in non-ideal conditions, but acknowledge that data from experiments and field deployments will include combinations of the cases presented in this paper along with other complexities not considered. Consequently, future work should apply these methods to experimental or field systems to advance the ultimate objective of using data-driven models to inform model predictive control schemes and optimize power absorption. With these applications in mind, DMD is a promising tool that has significant potential to support the development of the wave energy field.

ACKNOWLEDGMENTS

The authors acknowledge support from the National Science Foundation AI Institute in Dynamic Systems (grant number 2112085), Department of Defense Naval Facilities Engineering Systems Command (contract N0002421D6400/N0002421F8712), and the National

Science Foundation Graduate Research Fellowship (grant number DGE-2140004).

AUTHOR DECLARATIONS

Conflict of Interest Statement

The authors have no conflicts to disclose.

Author Contributions (CRediT)

B. Lydon: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - Original Draft, Visualization, Funding acquisition. **B. Polagye**: Conceptualization, Methodology, Resources, Writing - Review and Editing, Supervision, Project administration, Funding acquisition. **S. Brunton**: Conceptualization, Methodology, Resources, Writing - Review and Editing, Supervision, Project administration, Funding acquisition.

DATA AVAILABILITY STATEMENT

The data that supports the findings of this study are available from the corresponding author upon request.

Appendix A: Dynamic mode decomposition

This appendix includes a detailed derivation of the exact dynamic mode decomposition (DMD) algorithm, as defined in Tu et al.[26]. This algorithm is also described in detail in Kutz et al.[27], which includes sample code and data that is available at the accompanying website[59].

Consider a generic linear dynamical system with states $\boldsymbol{x}(t)$:

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{\mathcal{A}}\boldsymbol{x}, \tag{A1}$$

where $\mathcal{A}$ describes the system dynamics in continuous time. To model this system in discrete time, we use the linear operator $\boldsymbol{A} = \exp(\boldsymbol{\mathcal{A}}\Delta t)$ to describe the linear dynamics that shift

the system forward from time t_k to t_{k+1}, with $t_k = k\Delta t$:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k, \tag{A2}$$

where $\boldsymbol{x}_k$ is a column vector of all state measurements at time t_k. We can model Equation A2 using snapshot data at discrete times arranged into data matrices $\boldsymbol{X}$ and $\boldsymbol{X}^{'}$, such that:

$$\boldsymbol{X}^{'} \approx \boldsymbol{A}\boldsymbol{X}, \tag{A3}$$

where the data matrices are of the form:

$$\boldsymbol{X} = \begin{bmatrix} | & | & & | \\ \boldsymbol{x}_1 & \boldsymbol{x}_2 & ... & \boldsymbol{x}_m \\ | & | & & | \end{bmatrix}, \quad \boldsymbol{X}^{'} = \begin{bmatrix} | & | & & | \\ \boldsymbol{x}_2 & \boldsymbol{x}_3 & ... & \boldsymbol{x}_{m+1} \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{nxm}. \tag{A4}$$

The solution to the least-squares regression problem laid out in Equation A3 can be approximated by finding a matrix $\boldsymbol{A}$ that minimizes $||\boldsymbol{X}^{'} - \boldsymbol{A}\boldsymbol{X}||_F$, where $||\cdot||_F$ is the Frobenius norm. This can be done using the Moore-Penrose pseudoinverse of data matrix $\boldsymbol{X}$, denoted $\boldsymbol{X}^{\dagger}$:

$$\boldsymbol{A} = \boldsymbol{X}^{'}\boldsymbol{X}^{\dagger}. \tag{A5}$$

We calculate $\boldsymbol{X}^{\dagger}$ using the singular value decomposition (SVD) of data matrix $\boldsymbol{X}$:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{*}, \tag{A6}$$

where the columns of $\boldsymbol{U} \in \mathbb{R}^{nxn}$ and $\boldsymbol{V} \in \mathbb{R}^{mxn}$ are the left and right singular vectors of $\boldsymbol{X}$, respectively, and the diagonal of $\boldsymbol{\Sigma} \in \mathbb{R}^{nxn}$ contains the associated singular values. For large systems, it may be necessary to truncate these matrices and only take the first r singular values and modes to approximate a high-dimensional data matrix, such that:

$$\boldsymbol{X} \approx \boldsymbol{U}_r\boldsymbol{\Sigma}_r\boldsymbol{V}_r^{*}, \tag{A7}$$

where $\boldsymbol{U}_r \in \mathbb{R}^{nxr}$, $\boldsymbol{\Sigma}_r \in \mathbb{R}^{rxr}$, and $\boldsymbol{V}_r \in \mathbb{R}^{mxr}$. Because $\boldsymbol{X}$ is real, $\boldsymbol{U}_r$ and $\boldsymbol{V}_r$ are orthogonal, which means the psuedoinverse is easily calculated as $\boldsymbol{X}^{\dagger} = \boldsymbol{V}_r\boldsymbol{\Sigma}_r^{-1}\boldsymbol{U}_r^{*}$:

$$\boldsymbol{A} = \boldsymbol{X}^{'}\boldsymbol{X}^{\dagger} = \boldsymbol{X}^{'}\boldsymbol{V}_r\boldsymbol{\Sigma}_r^{-1}\boldsymbol{U}_r^{*} \in \mathbb{R}^{nxn}. \tag{A8}$$

The eigenvectors of $\boldsymbol{A}$ are the DMD modes, $\boldsymbol{\phi}_j$, with corresponding eigenvalues λ_j that represent their growth/decay rates and frequencies. For large systems, working with (n x n)

matrix $\boldsymbol{A}$ is too computationally expensive. To ease the computational burden, we project $\boldsymbol{A}$ onto the POD modes described by $\boldsymbol{U}_r$:

$$\tilde{\boldsymbol{A}} \doteq \boldsymbol{U}_r^* \boldsymbol{A} \boldsymbol{U}_r = \boldsymbol{U}_r^* \boldsymbol{X}^{'} \boldsymbol{V}_r \boldsymbol{\Sigma}_r^{-1} \in \mathbb{R}^{rxr}. \tag{A9}$$

To approximate the DMD modes in the POD basis, we set up an eigenvalue problem with $\tilde{\boldsymbol{A}}$:

$$\tilde{\boldsymbol{A}} \boldsymbol{W} = \boldsymbol{W} \boldsymbol{\Lambda}, \tag{A10}$$

where the columns of $\boldsymbol{W}$ are the eigenvectors of $\tilde{\boldsymbol{A}}$, and $\boldsymbol{\Lambda}$ contains the eigenvalues of both $\tilde{\boldsymbol{A}}$ and $\boldsymbol{A}$. As described by[26], the exact DMD modes are the columns of $\boldsymbol{\Phi}$, where:

$$\boldsymbol{\Phi} = \boldsymbol{X}^{'} \boldsymbol{V}_r \boldsymbol{\Sigma}_r^{-1} \boldsymbol{W}. \tag{A11}$$

With the DMD eigenvectors and eigenvalues, we can now describe the system dynamics:

$$\boldsymbol{x}(t) \approx \sum_{j=1}^{r} \boldsymbol{\phi_j} \exp(\omega_j t) b_j = \boldsymbol{\Phi} \exp(\boldsymbol{\Omega} t) \boldsymbol{b}, \tag{A12}$$

where $\omega_j = \ln(\lambda_j)/\Delta t$ are the continuous-time DMD eigenvalues and $\boldsymbol{b} = \boldsymbol{\Phi}^{\dagger} \boldsymbol{x}_1$ is the initial time vector translated to the DMD eigenbasis. In practice, it is likely the rank of the system cannot be known a priori, so the number of SVD modes to include is a free variable that may need adjusting to optimize accuracy and run time for the algorithm. With Equation A12, we can now describe and predict system state dynamics using only snapshot data.

1. Total-least-squares DMD

This appendix includes details of the total-least-squares DMD algorithm introduced in Section II C 1 and used in Section III A. We refer the reader to Hemati et al.[41] for a full derivation, and to Kutz et al.[27,59] for example code and data.

Consider a new combined data matrix $\boldsymbol{Z}$ and its SVD:

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{X}^{'} \end{bmatrix} = \boldsymbol{U}_Z \boldsymbol{\Sigma}_Z \boldsymbol{V}_Z^*. \tag{A13}$$

To account for noise in both $\boldsymbol{X}$ and $\boldsymbol{X}^{'}$, we project the data matrices onto a basis created by the right singular vectors of $\boldsymbol{Z}$, $\boldsymbol{V}_Z$:

$$\breve{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{V}_Z\boldsymbol{V}_Z^*, \quad \breve{\boldsymbol{X}}' = \boldsymbol{X}'\boldsymbol{V}_Z\boldsymbol{V}_Z^*. \tag{A14}$$

We now have a modified regression problem:

$$\breve{\boldsymbol{X}}' = \breve{\boldsymbol{A}}\breve{\boldsymbol{X}}, \tag{A15}$$

and can calculate the DMD matrix $\breve{\boldsymbol{A}}$ using the SVD of $\breve{\boldsymbol{X}}$:

$$\breve{\boldsymbol{A}} = \breve{\boldsymbol{U}}^*\breve{\boldsymbol{X}}'\breve{\boldsymbol{V}}\breve{\boldsymbol{\Sigma}}^{-1}, \tag{A16}$$

where $\breve{\boldsymbol{U}}$ and $\breve{\boldsymbol{V}}$ and right singular vectors of $\breve{\boldsymbol{X}}$, respectively, and $\breve{\boldsymbol{\Sigma}}$ contains the singular values of $\breve{\boldsymbol{X}}$, such that $\breve{\boldsymbol{X}} = \breve{\boldsymbol{U}}\breve{\boldsymbol{\Sigma}}\breve{\boldsymbol{V}}^*$. We can now follow the original DMD algorithm, replacing the original linear operator $\boldsymbol{A}$ with $\breve{\boldsymbol{A}}$.

2. Time delays

This appendix explains how to modify the data matrices using time delays to increase the rank of the data matrices and gain important phase information[26].

By stacking matrices that are shifted by a time step, we obtain new data matrices:

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{X}' \end{bmatrix}, \quad \boldsymbol{Y}' = \begin{bmatrix} \boldsymbol{X}' \\ \boldsymbol{X}'' \end{bmatrix}, \tag{A17}$$

where $\boldsymbol{X}''$ follows the same pattern as $\boldsymbol{X}'$, but is shifted two time steps rather than one. $\boldsymbol{Y}$ and $\boldsymbol{Y}'$ replace $\boldsymbol{X}$ and $\boldsymbol{X}'$ in the exact DMD algorithm in Section II C, respectively. Since the maximum rank we can consider when reconstructing $\boldsymbol{x}(t)$ is the length of the column vectors in the data matrices, time delays are particularly useful when higher rank is required for fidelity, but the number of sensors is limited. For example, a single time delay doubles the length of the column vectors in the data matrices, which doubles the number of SVD modes we can use to model the dynamics. Adding a shifted time matrix also provides more phase information, which can improve reconstruction accuracy, especially for the case of standing waves[26]. However, there are diminishing returns when using time delays that occur when adding more time-shifted copies only increases the number of columns in $\boldsymbol{X}$ without increasing its rank.

3. Optimized DMD

This appendix provides a brief derivation of the main ideas of optDMD that is introduced in Section II C 3 and used in Section III C. For a complete derivation as well as open-source code, we refer the reader to Askham and Kutz[64].

To begin, we assume our data $\boldsymbol{X}$ is a solution of a linear system of r differential equations, where r is the chosen rank, which means we can express $\boldsymbol{X}$ as a combination of exponentials,

$$\boldsymbol{X}^T \approx \boldsymbol{\Lambda}(\boldsymbol{\alpha})\boldsymbol{B}, \tag{A18}$$

where $\boldsymbol{\Lambda} \in \mathbb{C}^{mxr}$ is a matrix of exponentials, such that $\Lambda(\boldsymbol{\alpha})_{i,j} = \exp(\alpha_j t_i)$, $\boldsymbol{\alpha}$ are the equivalent of continuous-time DMD eigenvalues, and $\boldsymbol{B} \in \mathbb{C}^{rxn}$ are the weights of the exponential functions. To find the optimal values of $\boldsymbol{\alpha}$ and $\boldsymbol{B}$ that best approximates our data matrix $\boldsymbol{X}$, we want to solve the following nonlinear least squares problem:

$$\min_{\boldsymbol{\alpha},\,\boldsymbol{B}} \left|\left| \boldsymbol{X}^T - \boldsymbol{\Lambda}(\boldsymbol{\alpha})\boldsymbol{B} \right|\right|_F, \tag{A19}$$

which is achieved using variable projection methods. Due to the complexity of solving this optimization problem, we will not outline the process for solving for $\boldsymbol{\alpha}$ and $\boldsymbol{B}$ and refer the reader to Askham and Kutz[64] for the full derivation. In Eq. A19, F denotes the Frobenius norm. Once we have $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{B}}$ that minimizes Equation A19, we can calculate the DMD eigenvalues and eigenvectors:

$$\gamma_i = \hat{\alpha}_i, \quad \boldsymbol{\phi}_i = \frac{1}{\left|\left|\hat{\boldsymbol{b}}_i\right|\right|_2}\hat{\boldsymbol{b}}_i, \tag{A20}$$

and estimate system dynamics:

$$\boldsymbol{x}(t) = \sum_{i=1}^{r} \left|\left|\hat{\boldsymbol{b}}_i\right|\right|_2 \exp(\gamma_i t)\boldsymbol{\phi}_i, \tag{A21}$$

where $\hat{\boldsymbol{b}}_i$ is the ith column of $\hat{\boldsymbol{B}}^T$.

REFERENCES

[1]R. Manasseh, S. Sannasiraj, K. L. McInnes, V. Sundar, and P. Jalihal, "Integration of wave energy and other marine renewable energy sources with the needs of coastal societies," The International Journal of Ocean and Climate Systems **8**, 19–36 (2017).

[2]T. Aderinto and H. Li, "Ocean wave energy converters: Status and challenges," Energies **11**, 1250 (2018).

[3]M. Folley, T. Whittaker, M. Osterried, *et al.*, "The oscillating wave surge converter," in *The Fourteenth International Offshore and Polar Engineering Conference* (International Society of Offshore and Polar Engineers, 2004).

[4]M. Folley, A. Henry, and T. Whittaker, "Contrasting the hydrodynamics of heaving and surging wave energy converters," in *Proceedings of the 11th European Wave and Tidal Enerrgy Conference, Nantes, France* (2015) pp. 6–11.

[5]M. Folley and T. Whittaker, "The cost of water from an autonomous wave-powered desalination plant," Renewable Energy **34**, 75–81 (2009).

[6]T. Whittaker, D. Collier, M. Folley, M. Osterried, A. Henry, and M. Crowley, "The development of oyster—a shallow water surging wave energy converter," in *Proceedings of the 7th European wave and tidal energy conference* (2007) pp. 11–14.

[7]G. Giorgi and J. V. Ringwood, "Comparing nonlinear hydrodynamic forces in heaving point absorbers and oscillating wave surge converters," Journal of Ocean Engineering and Marine Energy **4**, 25–35 (2018).

[8]S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control* (Cambridge University Press, 2022).

[9]M. Folley, *Numerical modelling of wave energy converters: state-of-the-art techniques for single devices and arrays* (Academic Press, 2016).

[10]J. Davidson and R. Costello, "Efficient nonlinear hydrodynamic models for wave energy converter design—a scoping study," Journal of Marine Science and Engineering **8**, 35 (2020).

[11]S. Giorgi, J. Davidson, M. Jakobsen, M. Kramer, and J. V. Ringwood, "Identification of dynamic models for a wave energy converter from experimental data," Ocean Engineering **183**, 426–436 (2019).

[12]M. Rosati, T. Kelly, D. G. Violini, and J. Ringwood, "Data-based hydrodynamic modelling of a fixed owc wave energy converter," in *Proceedings of the 14th European Wave and Tidal Energy Conference, Plymouth, UK* (2021).

[13]M. Rosati, T. Kelly, and J. V. Ringwood, "Nonlinear data-based hydrodynamic modeling of a fixed oscillating water column wave energy device," IEEE Access **9**, 149756–149765 (2021).

[14]M. Rosati, J. Ringwood, H. Bingham, B. Joensen, and K. Nielsen, "A data-based modelling approach for a vented oscillating water column wave energy converter," Trends in Renewable Energies Offshore , 339–347 (2022).

[15]X. Wang, D. Liang, M. Li, P. Stansby, and L. Zhang, "Data-driven system identification modelling for multi-float m4 wave energy converter with elastic bed-buoy-bow float mooring," in *Proceedings of the 15th European Wave and Tidal Energy Conference* (2023).

[16]S. Giorgi, J. Davidson, and J. Ringwood, "Identification of nonlinear excitation force kernels using numerical wave tank experiments," in *Proceedings of the 11th European Wave and Tidal Energy Conference* (European Wave and Tidal Energy Conference 2015, 2015).

[17]J. Davidson, S. Giorgi, and J. V. Ringwood, "Numerical wave tank identification of nonlinear discrete time hydrodynamic models," Renewable Energies Offshore , 279 (2015).

[18]S. Giorgi, J. Davidson, and J. V. Ringwood, "Identification of wave energy device models from numerical wave tank data—part 2: Data-based model determination," IEEE Transactions on Sustainable Energy **7**, 1020–1027 (2016).

[19]G. Giorgi, M. Penalba, and J. Ringwood, "Nonlinear hydrodynamic force relevance for heaving point absorbers and oscillating surge converters," in *Proceedings of the AWTEC Asian Wave and Tidal Energy Conference, Singapore* (2016).

[20]J. van't Hoff, M. Folley, and T. Whittaker, "Numerical modelling of an oscillating wave surge converter using volterra theory," in *Proceedings of the 11th European Wave and Tidal Energy Conference, Nantes, France* (2015) pp. 6–11.

[21]D. Valério, M. J. Mendes, P. Beirão, and J. S. da Costa, "Identification and control of the aws using neural network models," Applied Ocean Research **30**, 178–188 (2008).

[22]C. Ni, X. Ma, and J. Wang, "Integrated deep learning model for predicting electrical power generation from wave energy converter," in *2019 25th International Conference on Automation and Computing (ICAC)* (IEEE, 2019) pp. 1–6.

[23]E. Katsidoniotaki, S. Guth, M. Göteman, and T. P. Sapsis, "Reduced order modeling of wave energy systems via sequential bayesian experimental design and machine learning," (2023).

[24]P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," Journal of fluid mechanics **656**, 5–28 (2010).

[25]C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," Journal of fluid mechanics **641**, 115–127 (2009).

[26]J. H. Tu, *Dynamic mode decomposition: Theory and applications*, Ph.D. thesis, Princeton University (2013).

[27]J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems* (SIAM, 2016).

[28]P. J. Schmid, "Application of the dynamic mode decomposition to experimental data," Experiments in fluids **50**, 1123–1130 (2011).

[29]I. Mezić, "Analysis of fluid flows via spectral properties of the koopman operator," Annual Review of Fluid Mechanics **45**, 357–378 (2013).

[30]J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Dynamic mode decomposition with control," SIAM Journal on Applied Dynamical Systems **15**, 142–161 (2016).

[31]M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," Automatica **93**, 149–160 (2018).

[32]B. Sharan, A. Dittmer, and H. Werner, "Real-time model predictive control for wind farms: a koopman dynamic mode decomposition approach," in *2022 European Control Conference (ECC)* (IEEE, 2022) pp. 1006–1011.

[33]B. O. Koopman, "Hamiltonian systems and transformation in hilbert space," Proceedings of the National Academy of Sciences **17**, 315–318 (1931).

[34]I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," Nonlinear Dynamics **41**, 309–325 (2005).

[35]S. L. Brunton, B. W. Brunton, J. L. Proctor, E. Kaiser, and J. N. Kutz, "Chaos as an intermittently forced linear system," Nature Communications **8**, 1–9 (2017).

[36]S. L. Brunton, M. Budišić, E. Kaiser, and J. N. Kutz, "Modern Koopman theory for dynamical systems," SIAM Review **64**, 229–340 (2022).

[37]M. J. Colbrook, "The mpedmd algorithm for data-driven computations of measure-preserving dynamical systems," arXiv preprint arXiv:2209.02244 (2022).

[38]M. J. Colbrook, L. J. Ayton, and M. Szőke, "Residual dynamic mode decomposition: robust and verified koopmanism," Journal of Fluid Mechanics **955**, A21 (2023).

[39]K. K. Chen, J. H. Tu, and C. W. Rowley, "Variants of dynamic mode decomposition: boundary condition, koopman, and fourier analyses," Journal of nonlinear science **22**, 887–915 (2012).

[40]P. J. Schmid, "Dynamic mode decomposition and its variants," Annual Review of Fluid Mechanics **54**, 225–254 (2022).

[41]M. S. Hemati, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "De-biasing the dynamic mode decomposition for applied koopman spectral analysis of noisy datasets," Theoretical and Computational Fluid Dynamics **31**, 349–368 (2017).

[42]S. Dawson, M. S. Hemati, M. O. Williams, and C. W. Rowley, "Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition," Experiments in Fluids **57**, 1–19 (2016).

[43]Y. Jia, J. Sun, Z. Xu, C. Sun, K. Meng, and Z. Y. Dong, "Data-driven economic mpc of a point absorber wave energy converter," IEEE Journal of Emerging and Selected Topics in Industrial Electronics (2024).

[44]P. Andersen, T. S. Pedersen, K. M. Nielsen, and E. Vidal, "Model predictive control of a wave energy converter," in *2015 IEEE Conference on Control Applications (CCA)* (IEEE, 2015) pp. 1540–1545.

[45]M. Richter, M. E. Magaña, O. Sawodny, and T. K. Brekken, "Power optimisation of a point absorber wave energy converter by means of linear model predictive control," IET Renewable Power Generation **8**, 203–215 (2014).

[46]G. Bacelli and J. V. Ringwood, "Nonlinear optimal wave energy converter control with application to a flap-type device," IFAC Proceedings Volumes **47**, 7696–7701 (2014).

[47]A. Mérigaud and J. V. Ringwood, "Optimal trajectories, nonlinear models and constraints in wave energy device control," IFAC-PapersOnLine **50**, 15645–15650 (2017).

[48]A. Serani, P. Dragone, F. Stern, and M. Diez, "On the use of dynamic mode decomposition for time-series forecasting of ships operating in waves," Ocean engineering **267**, 113235 (2023).

[49]M. Diez, A. Serani, E. F. Campana, and F. Stern, "Time-series forecasting of ships maneuvering in waves via dynamic mode decomposition," Journal of Ocean Engineering and Marine Energy , 1–8 (2022).

[50]K. Ruehl, D. Ogden, A. Yu, Yi-Hsiang Keester, N. Tom, D. Forbush, J. Leon, J. Grasberger, and S. Husain, "Wec-sim v5.0.1," (2022).

[51]M. Lawson, Y.-H. Yu, K. Ruehl, and C. Michelen, "Development and demonstration of the wec-sim wave energy converter simulation tool," (2014).

[52]A. Henry, O. Kimmoun, J. Nicholson, G. Dupont, Y. Wei, F. Dias, *et al.*, "A two dimensional experimental investigation of slamming of an oscillating wave surge converter," in *The Twenty-fourth International Ocean and Polar Engineering Conference* (International

Society of Offshore and Polar Engineers, 2014).

[53]P. Schmitt, S. Bourdier, T. Whittaker, D. Sarkar, E. Renzi, F. Dias, K. Doherty, J. van't Hoff, *et al.*, "Hydrodynamic loading on a bottom hinged oscillating wave surge converter," in *The Twenty-second International Offshore and Polar Engineering Conference* (International Society of Offshore and Polar Engineers, 2012).

[54]P. Lamont-Kane, A. McKinley, A. Henry, J. Nicholson, M. Folley, and B. Elsaesser, "Investigating extreme loads on an oscillating wave surge converter," in *Proceedings of the 11th European Wave and Tidal Energy Conference* (2015).

[55]N. Faedo, S. Olaya, and J. V. Ringwood, "Optimal control, mpc and mpc-like algorithms for wave energy systems: An overview," IFAC Journal of Systems and Control **1**, 37–56 (2017).

[56]M. Lawson, Y.-H. Yu, A. Nelessen, K. Ruehl, and C. Michelen, "Implementing nonlinear buoyancy and excitation forces in the wec-sim wave energy converter modeling tool," in *International Conference on Offshore Mechanics and Arctic Engineering*, Vol. 45547 (American Society of Mechanical Engineers, 2014) p. V09BT09A043.

[57]J. L. Proctor and P. A. Eckhoff, "Discovering dynamic patterns from infectious disease data using dynamic mode decomposition," International health **7**, 139–145 (2015).

[58]N. Mohan, K. Soman, and S. S. Kumar, "A data-driven strategy for short-term electric load forecasting using dynamic mode decomposition model," Applied energy **232**, 229–244 (2018).

[59]J. N. Kutz, S. L. Brunton, B. Brunton, and J. L. Proctor, "Dynamic mode decomposition," (2016).

[60]J. H. Tu, C. W. Rowley, J. N. Kutz, and J. K. Shang, "Spectral analysis of fluid flows using sub-nyquist-rate piv data," Experiments in Fluids **55**, 1–13 (2014).

[61]D. Duke, J. Soria, and D. Honnery, "An error analysis of the dynamic mode decomposition," Experiments in fluids **52**, 529–542 (2012).

[62]S. Bagheri, "Effects of weak noise on oscillating flows: Linking quality factor, floquet modes, and koopman spectrum," Physics of Fluids **26**, 094104 (2014).

[63]J. Juang and R. Pappa, "An eigensystem realization algorithm (era) for modal parameter identification and model reduction," in *JPL Proc. of the Workshop on Identification and Control of Flexible Space Struct., Vol. 3* (1985).

[64]T. Askham and J. N. Kutz, "Variable projection methods for an optimized dynamic mode

decomposition," SIAM Journal on Applied Dynamical Systems **17**, 380–416 (2018).

[65]J. Thomson, "Wave breaking dissipation observed with "swift" drifters," Journal of Atmospheric and Oceanic Technology **29**, 1866–1882 (2012).

[66]M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition," Journal of Nonlinear Science **25**, 1307–1346 (2015).