Comparison of Data-Driven Modeling Approaches for Control Optimization of Floating Offshore Wind Turbines

Athul Krishna Sundarrajan[1]

[1] Graduate Student
Department of Systems Engineering
Colorado State University
Athul.Sundarrajan@colostate.edu

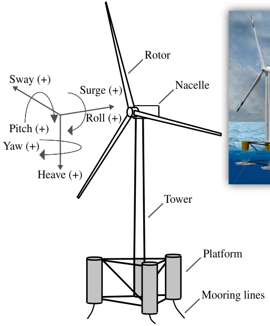
Speaker Series
Powerhouse Energy Institute
April 10, 2025

Outline

1

Motivation

→ Introduction (1)

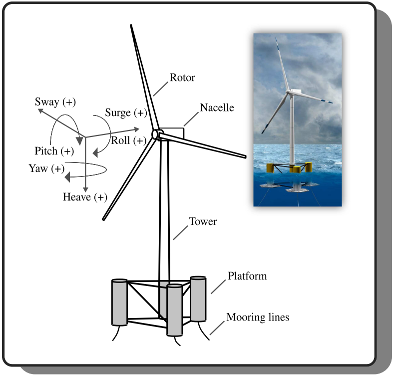


- There has been rising interest in using floating offshore wind turbines (FOWT) to harvest the energy present in offshore environments[1]
- A FOWT is a wind turbine that is mounted on a platform tethered by mooring lines

[1] Garcia-Sanz 2019

→ Introduction (2)

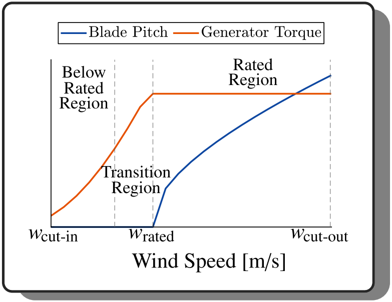
- A major impediment to the widespread adoption of FOWT is the current high cost of energy[1]
- Several factors impact the cost of energy, such as design of the system, choice of substructure, controller performance, installation and maintenance cost, etc.[2]
- The focus of this presentation will be on the controller and its impact on the turbine's performance
- FOWTs need to operate in stochastic offshore environments
- A controller is necessary for load mitigation and power generation of the turbine



[1] Garcia-Sanz 2019 [2] Garcia-Sanz 2019

Introduction (3)

- For FOWTs, the key control variables are the generator torque and blade pitch[1]
- Individual proportional-integral (PI) controllers are typically developed for both control variables
- The gains associated with the controllers are designed to be functions of specific parameters
- The controller's performance can be improved by selecting appropriate values of these parameters
- These values were traditionally selected with a control engineer's expertise
- But the design space can be nonlinear, and recently, optimization studies have been used to identify the parameter values



[1] Abbas et al. 2022

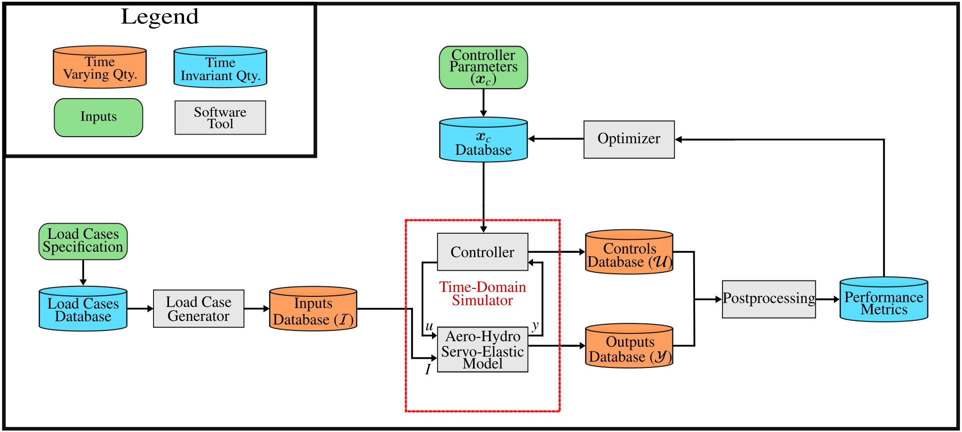
→ Introduction (4)

- For every optimization iteration, the FOWT is simulated for different wind speed profiles across its operating region
- The key performance metrics are calculated from the results of these simulations
- To get accurate measures of the performance, high-fidelity models of the FOWT must be used
- Evaluating the simulations using high-fidelity models can be computationally expensive, resulting in the study taking several hours or even several days to complete
- A real-time animation of a high-fidelity model simulation is shown in the following Link
- The focus of this study is to investigate different surrogate modeling approaches to reduce the computational expense of these controller optimization studies

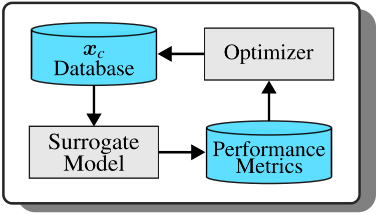
2

Surrogate Modeling Approach

Controller Optimization Process Workflow (1)

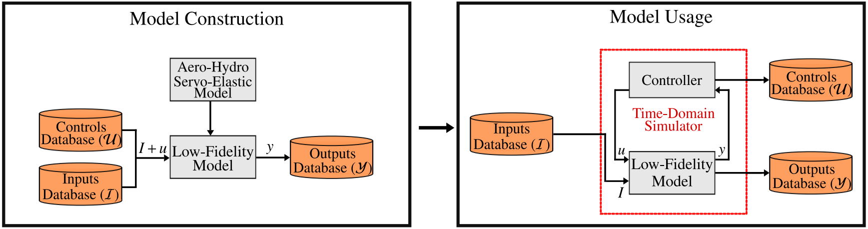


Surrogate Modeling Approach (1)



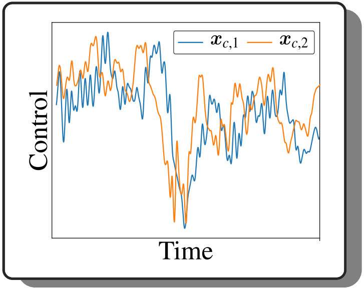
- Create a surrogate model that predicts the performance metrics given $\boldsymbol{x}_c$
- This approach would reduce the computational cost but has certain drawbacks
    - To understand why $\boldsymbol{x}_{c,\text{opt}}$ results in optimal performance, it is necessary to look at the time series signals of the controls and outputs
    - Additionally, if the variables as part $\boldsymbol{x}_c$ are changed, then the surrogate model needs to be trained again, which can be expensive
- A more effective surrogate modeling approach would overcome these drawbacks

Surrogate Modeling Approach (2)



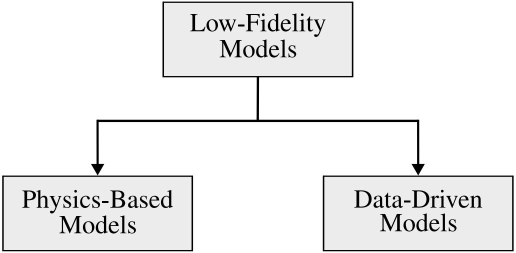
- Typically, evaluating the aero-hyrdo-servo-elastic model is the most computationally expensive part of the study
- The aero-servo-hydro-elastic model is physics-based model that captures the motions, loads, and responses of various FOWT subsystems
  - This is typically a differential algebraic equation (DAE)
  - This model can predict the timeseries of key outputs given the inputs and controls
- A surrogate model trained to predict this response can be coupled with the controller to perform simulations

→ Surrogate Modeling Approach (3)



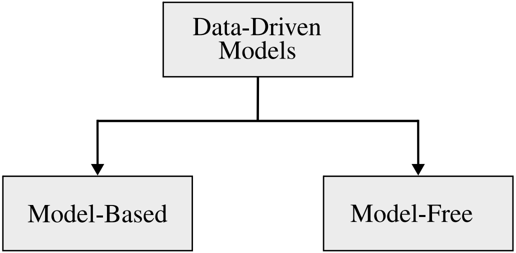
- Since the surrogate model does not depend on $\boldsymbol{x}_c$, it can be used for different studies
- Changing the value or even the variables associated with $\boldsymbol{x}_c$, changes the controls response as shown in the figure
- Different approaches can be used to train the surrogate model

→ Low-fidelity Models



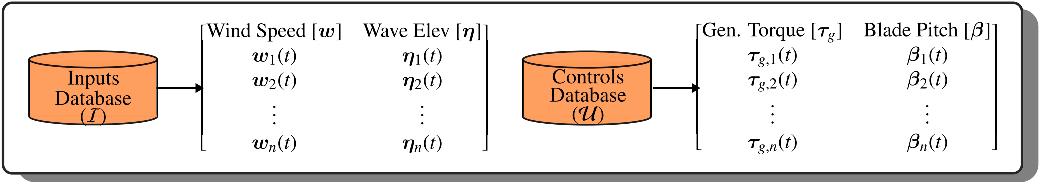
- Broadly, two different approaches can be used to construct low-fidelity models, physics-based, and data-driven
- Physics-based approaches try to approximate the key relationships using prior knowledge of the underlying physics of the system
- Data-driven approaches try to approximate the relationships by capturing the trends in the input-output data, which will be the focus of this talk

→ Data-Driven Low-fidelity Models



- Data-driven modeling approaches refer to a wide variety of approaches such as curve-fitting, machine learning, probabilistic models, systems identification approaches
- They can be classified further into two types: model-free (machine learning) and model-based (systems identification)
- Model-based approaches assume that the inputs and outputs are related through a mathematical model
- Model-free approaches capture the temporal relations between the input-output data
- We explore two model-based approaches and a model-free approach

→ Inputs



- The two inputs to the model are:
  1. Wind Speed (w) generated using the IEC Kaimal spectrum
  2. Wave Elevation (η) generated using the JONSWAP spectrum
- The two controls used as part of the model are:
  1. Generator torque (τ_g)
  2. Blade pitch (β)

→ Outputs

$$\text{Outputs Database } (\mathcal{Y}) \rightarrow \begin{bmatrix} \text{Gen. Speed } [\omega_g] & \text{Tower Top Acc. } [\ddot{x}_t] & \text{Tower-base Moment } [M_t] & \text{Ptfm. Pitch } [\Theta_p] & \text{Tower-Top Disp. } [\delta_t] \\ \omega_{g,1}(t) & \ddot{x}_{t,1}(t) & M_{t,1}(t) & \Theta_{p,1} & \delta_{t,1} \\ \omega_{g,2}(t) & \ddot{x}_{t,2}(t) & M_{t,2}(t) & \Theta_{p,2} & \delta_{t,2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \omega_{g,n}(t) & \ddot{x}_{t,n}(t) & M_{t,n}(t) & \Theta_{p,n} & \delta_{t,n} \end{bmatrix}$$

- There are a total of five different outputs that are included in the model:
  1. Generator speed (ω_g)
  2. Tower top acceleration ($\ddot{x}_t$)
  3. Tower base fore-aft moment (M_t)
  4. Platform pitch (Θ_p)
  5. Tower top fore-aft displacement (δ_t)
- The Aero-Hydro-Servo-Elastic model in OpenFAST[1] is used in this study
- The open-source tool ROSCO[2] is the controller used in this study

[1] Jonkman 2007 [2] Abbas et al. 2022

3

Model Construction

→ Derivative Function Surrogate Models (DFSM)

$$\frac{d\boldsymbol{\xi}}{dt} = \dot{\boldsymbol{\xi}} = \boldsymbol{f}(\boldsymbol{\xi}, \boldsymbol{u}) \approx \boldsymbol{f}_{\mathrm{LF}}(\boldsymbol{\xi}, \boldsymbol{u}) \tag{1a}$$

$$\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{\xi}, \boldsymbol{u}) \approx \boldsymbol{g}_{\mathrm{LF}}(\boldsymbol{\xi}, \boldsymbol{u}) \tag{1b}$$

$$\dot{\boldsymbol{\xi}} \approx \boldsymbol{f}_{\mathrm{LF}}(\boldsymbol{\xi}, \boldsymbol{u}) = \boldsymbol{A}(w)\boldsymbol{\xi} + \boldsymbol{B}(w)\boldsymbol{u} \tag{2a}$$

$$\boldsymbol{y} \approx \boldsymbol{g}_{\mathrm{LF}}(\boldsymbol{\xi}, \boldsymbol{u}) = \boldsymbol{C}(w)\boldsymbol{\xi} + \boldsymbol{D}(w)\boldsymbol{u} \tag{2b}$$

- Derivative function surrogate models (DFSM) assume the system can be approximated using the continuous time nonlinear state-space model as shown in Eq. (1)
- Further, we assume that the system can be approximated by a linear parameter varying (LPV) system as shown in Eq. (3)

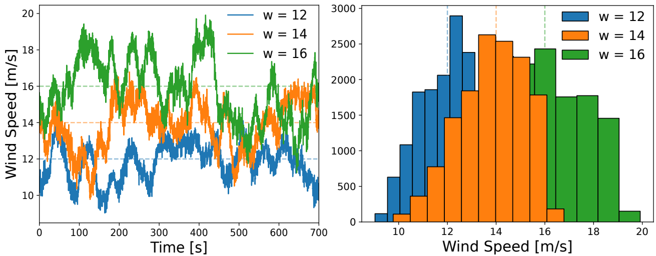
→ DFSM (2)

$$\dot{\boldsymbol{\xi}} \approx \hat{\dot{\boldsymbol{\xi}}} = \boldsymbol{f}_{\mathrm{LF}}(\boldsymbol{\xi}, \boldsymbol{u}) = \boldsymbol{A}(w)\boldsymbol{\xi} + \boldsymbol{B}(w)\boldsymbol{u} \tag{3a}$$

$$\boldsymbol{y} \approx \hat{\boldsymbol{y}} = \boldsymbol{g}_{\mathrm{LF}}(\boldsymbol{\xi}, \boldsymbol{u}) = \boldsymbol{C}(w)\boldsymbol{\xi} + \boldsymbol{D}(w)\boldsymbol{u} \tag{3b}$$

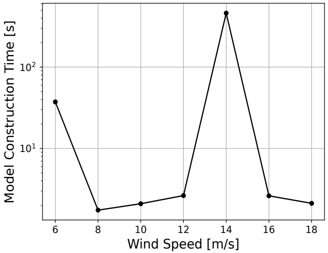
- An LPV system is a type of linear time varying system (LTV) where the system matrices are a function of a parameter, in this case wind speed
- The LPV model is constructed by identifying linear time-invariant (LTI) models corresponding to different values of w and interpolating over them
- We use an optimization study to identify the model parameters corresponding to the $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ matrices
- The objective of the optimization study is to minimize the error between the actual and predicted state derivatives and outputs, i.e., $||\dot{\boldsymbol{\xi}} - \hat{\dot{\boldsymbol{\xi}}}||$, $||\boldsymbol{y} - \hat{\boldsymbol{y}}||$
- By ensuring that the real-parts of the eigen values of $\boldsymbol{A}$ are non-positive, we can ensure that the system is stable, which is required for closed-loop simulations

→ DFSM (3)



- We use a hybrid-optimization strategy to identify the model parameters
  - A combination of genetic algorithm and an interior point solver is used
- Certain properties of FOWT dynamics can be used to accelerate the model identification process
- FOWT behavior is heavily dependent on the wind speed
- Because turbulent simulations are used to train the models, a model trained for predicting the response of a load case with $\bar{w} = 14$ [m/s] will also have to be accurate for simulations with $\bar{w} = 12$ [m/s] and $\bar{w} = 16$ [m/s]

→ DFSM (4)



- We use this property to accelerate the model identification process
- The hybrid optimization algorithm is used to identify the model parameters associated with $\bar{w} = 14$ [m/s]
- This is used as the starting point for $\bar{w} = 12$ [m/s] and $\bar{w} = 16$ [m/s], instead of the GA
- Subsequently the model identified for $\bar{w} = 12$ [m/s] is used as the starting point for $\bar{w} = 10$ [m/s], and so on

→ Linear System Identification Methods

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \tag{4a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k \tag{4b}$$

- The goal of linear sys-id methods is to identify a discrete-time linear state-space model of the form shown in Eq. (4) using input-output time-series data $(\mathcal{U}, \mathcal{Y})$
- They are also an example of model-based data-driven approaches
- Sys-id methods have been extensively used to model and control complex real-world systems
- There is the added advantage of the identified models being numerically stable
- In this study, we use a class of subspace identification methods called 'n4sid' to identify Eq. (4),
- More details regarding the approaches can be found in Ref[1]

[1] Van Overschee and De Moor 1994

→ Long Short-Term Memory Networks

- LSTM networks are specifically developed to handle sequence data and have been widely utilized for time-series predictions
- Some aspects of LSTM models are particularly useful when approximating non-linear systems
  - Their ability to capture long-term trends
  - Can easily handle multi-output models
  - Ability to capture temporal variations
- LSTM models are an example of model-free data-driven approaches
- LSTM models suit for the modeling requirements in this study
- Although advanced architectures have been studied for LSTM applications, we use a ‘vannila’ LSTM model in this study
- The model consists of an LSTM layer and a dense layer

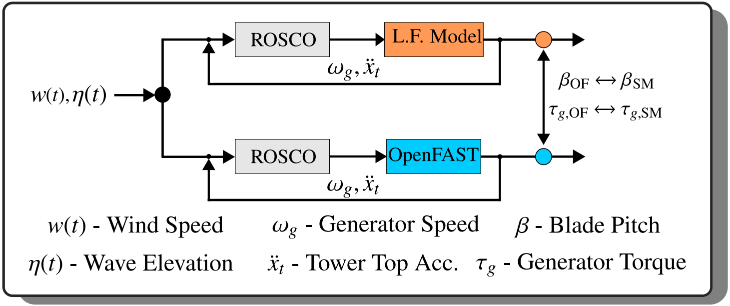
→ Model Training

- We generate 10 different load cases with an average wind speed of $w_{\text{avg}} = [6, 8, \cdots, 18]$ [m/s] to create the input, controls, and output data bases
  - We use 50 % to train the models, and 50% in testing
- The time taken to obtain the training data, in terms of CPU hours, is $5\ (n_s) \times 7\ (n_w) \times 0.33$ hours $= 11.6$ hours
- The DFSM is trained using the aforementioned method, and it takes around 8.4 minutes to train
- We use the systems-identification toolbox in `MATLAB` to identify the sys-id model
  - The model order of the LTI system is $n_x = 6$
  - The training time is 2 mins
- We use the LSTM implementation in `TensorFlow` in this study
  - Load cases with $w_{\text{avg}} = 12$ and $w_{\text{avg}} = 16$ [m/s] were included as part of the training data
  - The training time is 8.6 mins

4

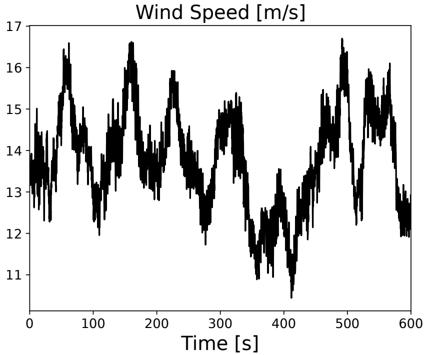
Model Validation

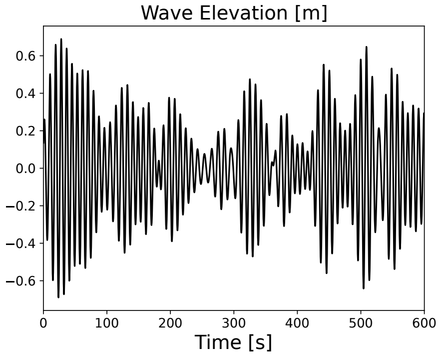
→ Validation Study



- For the same set of inputs, closed-loop simulations were carried out by coupling the low-fidelity models to the controller
- The controls and outputs predicted by the low-fidelity models are compared to the results from OpenFAST

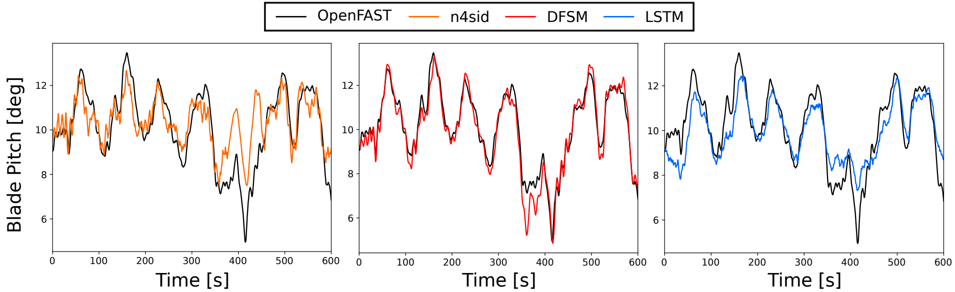
→ Inputs





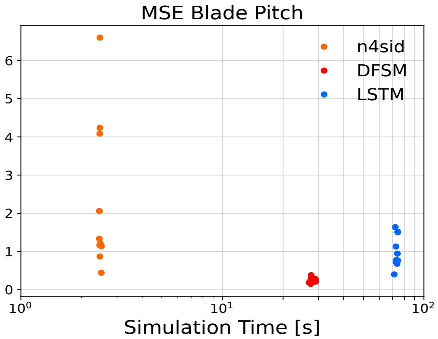
- Time series of a wind speed and wave elevation inputs for a test case

→ Validation Results (blade pitch)



- Timeseries comparison of blade pitch trajectory obtained using all three surrogate models and OpenFAST for a test case

→ Comparison

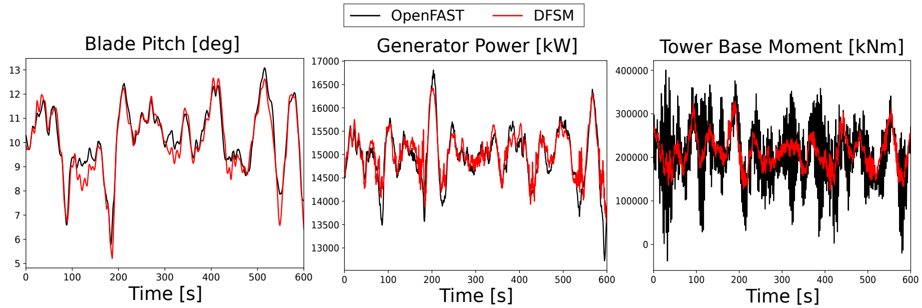


- All three models are significantly faster than the OpenFAST simulations
- The DFSM balances simulation time, accuracy, and variance
- A sample simulation for the DFSM in real-time is shown in the following Link

5

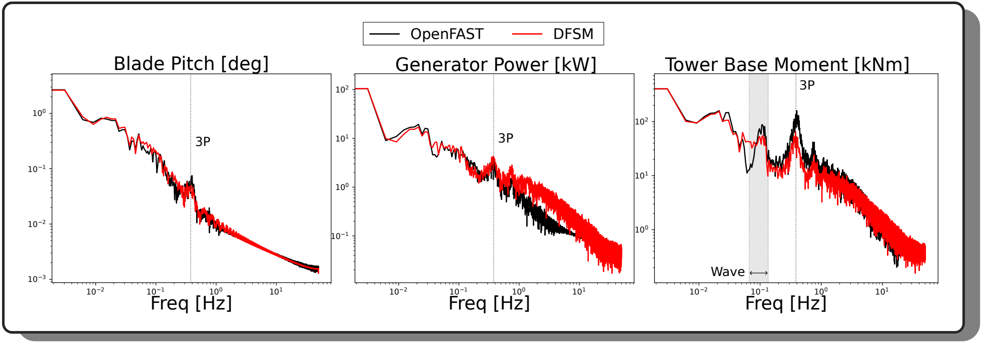
Additional Results and Parameter Sweep Results

➞ Validation Results (Time series)



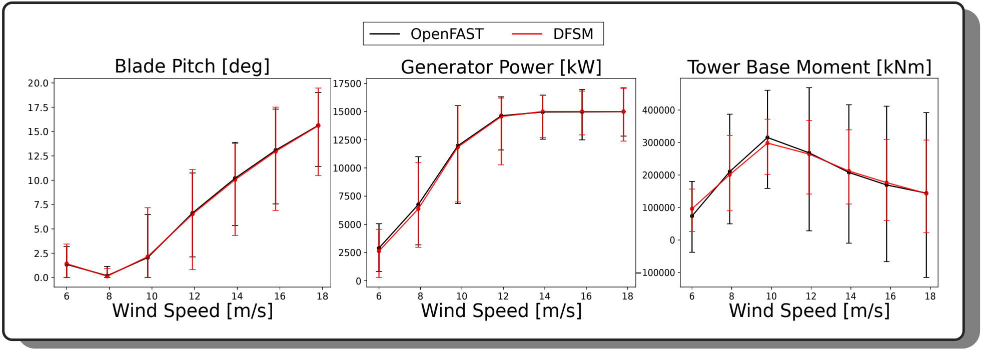
- Timeseries comparison of key signals between DFSM and OpenFAST for a test case

Validation Results (PSD)



- Power spectral density (PSD) comparison of key signals between DFSM and OpenFAST for a test case

→ Validation Results (Range)

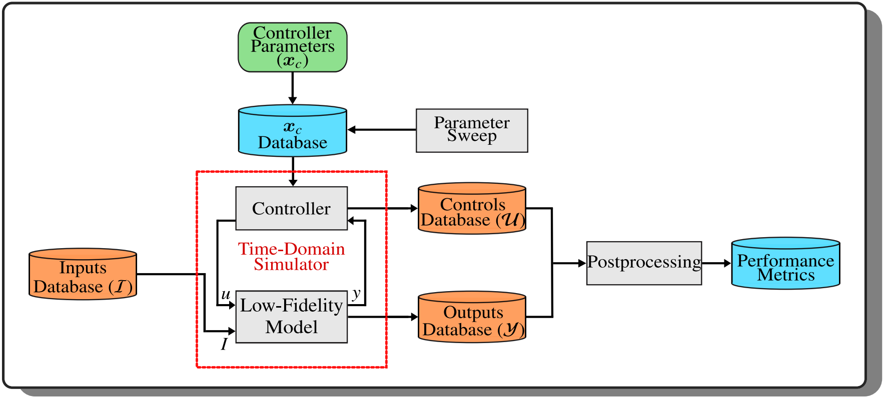


- Comparison of mean and range of key signals between DFSM and OpenFAST for all test cases

→ Discussion

- The results from these different validation tests show that the DFSM can balance accuracy and computational cost
- Since a linear model is used as part of the DFSM, it cannot capture the highly nonlinear trends in tower base moment signal
  - It constantly underpredicts this signal
- As a consequence of this, key metrics calculated, like the damage equivalent load (DEL) from this signal will be different between the OpenFAST and DFSM
- Even though the range is underpredicted, the linear approximation can capture the mean value of this quantity accurately
- Other quantities, such as the annual energy production (AEP), calculated from the generator power signal, will be closer between OpenFAST and DFSM

→ Parameter Sweep (1)

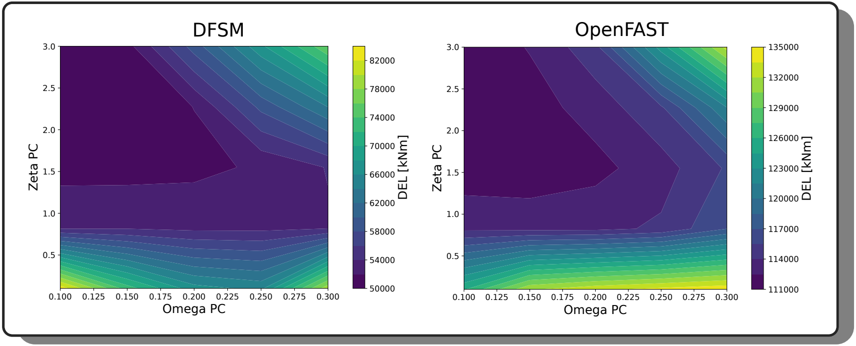


- Instead of an optimization study, we carry out a parameter sweep of key controller variables

→ Parameter Sweep (2)

- The natural frequency (ω_{pc}) and damping ratio (ζ_{pc}) are key parameters that affect the performance of the blade pitch controller
- A full factorial sampling scheme with five samples for each variable is used
- At each design point, the system is simulated for 30 different load cases
  - 5 wind speeds with 6 seeds for each wind speed
- The goal of this study is to see if the DFSM can replicate the correct trends in the DEL, which is calculated using the tower-base moment signal
- The total time in CPU hours required to run this study are as follows:
  - DFSM: $25 \times 30 \times 30$ sec = 5.2 hours
  - OpenFAST: $25 \times 30 \times 20$ min = 250 hours

Parameter Sweep (3)



- The design space for the DEL obtained using the OpenFAST and DFSM
- The DFSM underpredicts the DEL, but identifies the trends in the design space accurately

6

Conclusion

Conclusion

- We present a new approach to constructing a DFSM model from input-output time series data of FOWT
- We explored the trade-offs and performance of this DFSM model to LSTM and systems-identification approaches
- The use of the DFSM in closed-loop simulations and a parameter sweep of a key controller parameter have been explored
- The DFSM balances computational cost and accuracy
- Future studies will focus on improving the capabilities of the DFSM to predict nonlinear quantities
- The use of the DFSM in controller optimization studies utilizing more variables will also be tested
- The studies outlined here have also been carried out for marine hydrokinetic turbines

7

References

→ References

- N. J. Abbas et al. (2022). "A reference open-source controller for fixed and floating offshore wind turbines". *Wind Energy Sci.* 7.1. doi: 10.5194/wes-7-53-2022
- M. Garcia-Sanz (2019). "A Metric Space with LCOE Isolines for Research Guidance in wind and hydrokinetic energy systems". *Wind Energy* 23.2. doi: 10.1002/we.2429
- J. M. Jonkman (2007). *Dynamics modeling and loads analysis of an offshore floating wind turbine*. Tech. rep. doi: 10.2172/921803. DOI: 10.2172/921803
- P. Van Overschee and B. De Moor (1994). "N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems". *Automatica* 30.1. doi: 10.1016/0005-1098(94)90230-5. ISSN: 0005-1098

Questions?

Comparison of Data-Driven Modeling Approaches for Control Optimization of Floating Offshore Wind Turbines

Speaker Series

Athul Krishna Sundarrajan
Athul.Sundarrajan@colostate.edu

→ Damage Equivalent Load

- We use the damage equivalent load metric (DEL) metric as a measure of the fatigue damage incurred by the turbine
- DEL approximates the cumulative fatigue damage incurred over a given time span to a single equivalent load that would cause the same damage
  - We use the rainflow counting method and the Palmgren-Miner rule to the tower-base moment (M_t) estimate the DEL for the tower-base
  - The rainflow counting method identifies the loading cycles in the time series, and the amplitude and mean moment for every loading cycle are identified
  - This range of amplitudes for the different cycles is then separated into various bins, and the count for each bin is then calculated to obtain the corresponding $\mathrm{M}_{t,i}$ and n_i pair, where $\mathrm{M}_{t,i}$ refers to the amplitude of the "i-th" bin, and n_i refers to its count
  - DEL for a specific load case is then calculated as:

$$\mathrm{DEL} = \left(\sum_{i=1}^{100} \frac{M_{t,i}^m n_i}{t_{\text{span}}} \right)^{1/m} \tag{5}$$