

# Surrogate Modeling of Time-Dependent Metocean Conditions during Hurricanes

Chi Qiao

Northeastern University

Andrew T. Myers (✉ [atm@neu.edu](mailto:atm@neu.edu))

Northeastern University

---

## Research Article

**Keywords:** Time-dependent surrogate modeling, Kriging, Neural Networks, Deep Learning, ocean waves, storm surge

**Posted Date:** February 23rd, 2021

**DOI:** <https://doi.org/10.21203/rs.3.rs-234266/v1>

**License:** © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

**Version of Record:** A version of this preprint was published at Natural Hazards on August 26th, 2021. See the published version at <https://doi.org/10.1007/s11069-021-05002-2>.

# Surrogate Modeling of Time-Dependent Metocean Conditions during Hurricanes

C. Qiao <sup>a</sup>, A.T. Myers <sup>a</sup>

<sup>a</sup> *Department of Civil & Environmental Engineering, Northeastern University, Boston, MA, USA*

## Abstract

Metocean conditions during hurricanes are defined by multiple parameters (e.g., significant wave height and surge height) that vary in time with significant auto- and cross-correlation. In many cases, the nature of the variation of these characteristics in time is important to design and assess the risk to offshore structures, but a persistent problem is that measurements are sparse and time history simulations using metocean models are computationally onerous. Surrogate modeling is an appealing approach to ease the computational burden of metocean modeling, however, modeling the time-dependency of metocean conditions using surrogate models is challenging because the conditions at one time instant are dependent on not only the conditions at that instant but also on the conditions at previous time instances. In this paper, time-dependent surrogate modeling of significant wave height, peak wave period, peak wave direction, and storm surge is explored using a database of metocean conditions at an offshore site. Three types of surrogate models, including Kriging, Multilayer Perceptron (MLP), and Recurrent Neural Network with Gated Recurrent Unit (RNN-GRU), are evaluated, with two different time-dependent structures considered for the Kriging model and two training set sizes for the MLP model, resulting in a total of five models evaluated in this paper. The performance of the models is compared in terms of accuracy and sensitivity towards hyperparameters, and the MLP and RNN-GRU models are demonstrated to have extraordinary prediction performance in this context.

**Keywords:** Time-dependent surrogate modeling, Kriging, Neural Networks, Deep Learning, ocean waves, storm surge

## 1. Introduction

Modeling of metocean conditions (e.g., significant wave height, peak wave period, etc.) during hurricanes is a situation well-suited for surrogate modeling, as the underlying physical processes are complex and numerical models are so computationally demanding that their potential for engineering applications is limited. The term surrogate model refers to a data-driven statistical model that approximates the behavior of a complex process characterized by input (measurable quantities that have relevant predictive power) and output (the variable(s) of interest). Some parametric surrogate models of metocean conditions during hurricanes [1] have been proposed, but these are limited to deep and open waters. However, for more general situations, such as those with limited water depth or fetch length, the conventional practice is to use numerical, time-dependent models [2, 3] that account for important nonlinear phenomena such as wind-sea and wave-wave interactions, energy dissipation, etc. Because the historical record of hurricanes is relatively short, a catalog of synthetic hurricanes representing a much longer period of time is commonly applied in engineering design and risk assessments. Numerical simulations of metocean conditions for each of the thousands of hurricanes in such a catalog are intractable, and a promising approach is to

41 instead simulate only a small subset of these hurricanes numerically and then create a surrogate  
42 model to replace the time-consuming numerical model for simulating the remaining hurricanes.

43 Surrogate modeling for time-dependent processes, such as metocean conditions during hurricanes,  
44 is a challenging task. In this context, time-dependent processes refer to processes with  
45 characteristics that vary in time and are affected not only by the current conditions but also by the  
46 conditions at previous instances of time. This kind of behavior is difficult for surrogate models to  
47 represent because the correlation between the predictions made for different time instances also  
48 needs to be captured by surrogate models. Many types of surrogate models have been explored for  
49 modeling metocean conditions, including decision trees [4], response surfaces [5], Kriging [6], and  
50 support vector machine (SVM) [7]. When applying these models to a time series, each set of input  
51 and output are independent of each other, and so these models cannot explicitly model the time-  
52 dependency of metocean processes. Some researchers have proposed approaches to adapt these  
53 models for time-dependent processes. For example, Jia et al. [8] used Kriging to model storm surge  
54 with a high-dimensional vector designed to preserve time-dependence. In addition to the techniques  
55 above, Neural Networks are also used for surrogate modeling of metocean conditions, such as [14-  
56 16], where the networks utilized by these researchers have a structure with one hidden layer. In  
57 recent years, Neural Networks have evolved alongside the rapid development of Deep Learning,  
58 resulting in a new family of techniques that are referred to as Deep Neural Networks, which have  
59 network structures with multiple hidden layers and more sophisticated layer operations – from the  
60 basic operations of a Multilayer Perceptron, MLP, to the complex operations of a Recurrent Neural  
61 Network, RNN. These advances have yielded better performance for many complex tasks such as  
62 natural language processing [17] and computer vision [18], but their application in the surrogate  
63 modeling of metocean conditions is so far limited (refer to [7] as one example).

64 Development of a surrogate model of time-dependent metocean conditions involves several loosely  
65 coupled choices: the size of the training dataset, the selection of the type of surrogate model (e.g.,  
66 Kriging, MLP, RNN, etc.), and the design of the time-dependent structure. A model with a time-  
67 dependent structure has capabilities that can predict a series of output that preserves correlation in  
68 time. Some models, like RNN, have inherent time-dependent structures, while others, like Kriging  
69 and MLP, do not have such structures inherently. For the latter type of models, the effect of time-  
70 dependent processes can be modeled with *ad hoc* manipulation of the input and/or output vectors  
71 of the models. In this paper, a database of metocean conditions during hurricanes for an offshore  
72 site is used to compare five combinations of the surrogate model, the time-dependent structure, and  
73 the size of the training dataset. This paper has two goals: (1) to illustrate how intrinsic  
74 characteristics of the surrogate model along with details of its time-dependent structure and size of  
75 training dataset affect the overall performance of the model, and (2) to demonstrate the enormous  
76 potential of so-called Deep Neural Networks in surrogate modeling of metocean conditions. For  
77 the Neural Networks considered in this paper, the influence of the network structure (i.e., the  
78 number of hidden layers and the number of units within each hidden layer) is also explored. This  
79 paper is organized as follows. First, the background of the surrogate models utilized in this paper  
80 is provided in Section 2. Then, the database of metocean conditions during hurricanes, which is  
81 used for model training, is described in Section 3. Details of the implementation of the five models  
82 considered in this paper are presented in Section 4. Results are provided and discussed in Section  
83 5, and conclusions are summarized in Section 6.

84

## 85 2. Background

86 Three types of surrogate models are considered in this paper: Kriging, the Multilayer Perceptron  
 87 (MLP), and Recurrent Neural Network with Gated Recurrent Unit (RNN-GRU) [19]. The latter  
 88 two models are categorized as Deep Neural Networks. These three types of models are chosen to  
 89 represent a relevant range of characteristics. For example, the RNN-GRU model has an inherent  
 90 time-dependent structure and can naturally model time-dependent processes, while the other two  
 91 require an *ad hoc* arrangement of the input and output vectors to model such processes. Another  
 92 important distinction is that Kriging is a memory-based approach (i.e., all training data are  
 93 memorized to make predictions), while the other two are parametric models (i.e., training data is  
 94 only used to determine a set of parameters of the model and is not directly involved in making  
 95 predictions). Key features of these models are provided in Table 1, and the details of each model  
 96 are presented separately in the remainder of this section.

97 *Table 1. Summary of the three surrogate models used in this paper to model time-dependent*  
 98 *processes. A list of choices for the key hyperparameters explored in this paper is provided in*  
 99 *square brackets.*

Surrogate Model	Time-dependent structure?	Memory-based?	Key hyperparameters
Kriging	No	Yes	<ul style="list-style-type: none"> <li>• Type of regression function: linear</li> <li>• Type of correlation function: exponential</li> </ul>
MLP	No	No	<ul style="list-style-type: none"> <li>• Number of hidden layers: [1, 3, 5, 7]</li> <li>• Number of units per layer: [16, 32, 64, 128, 256, 512, 1024, 2048, 4096]</li> <li>• Activation function: ReLU</li> </ul>
RNN-GRU	Yes	No	<ul style="list-style-type: none"> <li>• Number of hidden layers: [1, 3, 5, 7]</li> <li>• Number of units per layer: [16, 32, 64, 128, 256, 512, 1024, 2048, 4096]</li> <li>• Activation function: hyperbolic tangent</li> <li>• Recurrent activation function: hard sigmoid</li> </ul>

100

### 101 2.1 Kriging model

102 For a training dataset  $(\mathbf{X}_{m \times n}, \mathbf{Y}_{m \times q})$  composed of  $m$  samples of  $n$  input variables and  $q$  output  
 103 variables, the Kriging model uses a linear combination of training output  $\mathbf{Y}$  to provide the  
 104 prediction  $\hat{\mathbf{y}}$  for any unknown input  $\tilde{\mathbf{x}}$  as,

$$105 \quad \hat{\mathbf{y}} = \mathbf{c}^\top \mathbf{Y} = \mathbf{f}(\tilde{\mathbf{x}})^\top \boldsymbol{\beta}^* + \mathbf{r}(\tilde{\mathbf{x}})^\top \boldsymbol{\gamma}^* \quad (1)$$

106 where  $\mathbf{c}^\top$  is the coefficient matrix of the linear combination,  $\mathbf{f}_{p \times 1}(\mathbf{x})$  is the regression function,  
 107  $\boldsymbol{\beta}_{p \times q}^*$  is the result of the generalized least squares method  $\boldsymbol{\beta}^* = (\mathbf{F}^\top \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{F}^\top \mathbf{R}^{-1} \mathbf{Y}$ ,  $\mathbf{F}_{m \times p}$  is the  
 108 result of the regression function evaluated at all the training inputs  $\mathbf{F}_{m \times p} = [\mathbf{f}(\mathbf{x}_1) \dots \mathbf{f}(\mathbf{x}_m)]^\top$ ,  
 109  $\mathbf{R}_{m \times m}$  is the result of the correlation function  $\mathcal{R}(\boldsymbol{\theta}, \mathbf{x}_i, \mathbf{x}_j)$  evaluated at all of the training inputs  
 110 (i.e.,  $i, j = 1, \dots, m$ ),  $\mathbf{r}(\tilde{\mathbf{x}})_{m \times 1}$  is the result of the correlation function evaluated between the  
 111 training inputs and  $\tilde{\mathbf{x}}$ , and  $\boldsymbol{\gamma}_{m \times q}^*$  is the result of  $\boldsymbol{\gamma}^* = \mathbf{R}^{-1}(\mathbf{Y} - \mathbf{F}\boldsymbol{\beta}^*)$ . The parameter  $\boldsymbol{\theta}$  is optimized  
 112 to minimize the mean squared error  $E[(\hat{\mathbf{y}} - \mathbf{y})^2]$  of the training dataset.

113 There are two hyperparameters in the Kriging model, the regression function  $f(\mathbf{x})$  and the  
 114 correlation function  $\mathcal{R}(\boldsymbol{\theta}, \mathbf{x}_i, \mathbf{x}_j)$ . Possible forms for the function  $f(\mathbf{x})$  include a constant function  
 115  $f^{(0)}(\mathbf{x}) = 1$ , a linear function  $f^{(1)}(\mathbf{x}) = [1, x_1, \dots, x_n]^\top$ , and a quadratic function  $f^{(2)}(\mathbf{x}) =$   
 116  $[1, x_1, \dots, x_n, x_1^2, x_1x_2, \dots, x_n^2]$ . In this paper, the linear regression function and the exponential  
 117 correlation function are used. The constant regression function and two other types of correlation  
 118 functions, linear and Gaussian (detailed expressions for these are provided in [20]), were included  
 119 in preliminary experiments, but were found to have higher prediction errors and are not discussed  
 120 here. Quadratic or higher-order regression functions are not considered because the dimension of  
 121  $f(\mathbf{x})$  exceeds the number of training samples for some cases considered in this paper. Note that the  
 122 inclusion of some quadratic terms might improve the performance of the regression term  $\mathbf{f}(\hat{\mathbf{x}})^\top \boldsymbol{\beta}^*$ ,  
 123 however, such implementation requires domain knowledge and is not common practice.

## 124 2.2 MLP

125 An MLP model is the most basic Neural Network. It includes several layers of units, with the first  
 126 layer representing the input vector and the last layer representing the output vector. Figure 1 shows  
 127 the structure of an MLP model with two hidden layers as an example. The operation of each hidden  
 128 layer is expressed as,

$$129 \quad \mathbf{z}_2 = g(\mathbf{z}_1 \cdot \mathbf{W} + \mathbf{b}) \quad (2)$$

130 where, for each layer of the model,  $\mathbf{z}_1$  is the input vector,  $\mathbf{z}_2$  is the output vector,  $\mathbf{W}$  is the weight  
 131 matrix,  $\mathbf{b}$  is the vector representing bias, and  $g$  is the so-called activation function, which  
 132 nonlinearly transforms the data elementwise.

133 Compared to the Kriging model, the MLP model has more hyperparameters. First, the MLP model  
 134 includes  $l$  hidden layers (the number of hidden layers is referred to herein as the depth of the  
 135 network) and  $k$  units per layer (the number of units per layer is referred to herein as the width of  
 136 the network), which together determine the structure of the network. The ability of the model to  
 137 approximate nonlinear behavior and the complexity of the training process increase with the  
 138 number of hidden layers and the number of units per layer. These parameters affect model  
 139 performance as does the activation function  $g$ .

140 In this paper, the selected activation function is ReLU (Rectified Linear Unit [21], expressed as  
 141  $y = \max(0, x)$ ). Three other types of activation functions, namely ELU (Exponential Linear Unit  
 142 [23]), sigmoid (i.e., standard logistic), and hyperbolic tangent, were considered during preliminary  
 143 experiments, but ReLU was found to have the best performance and so the others are not discussed  
 144 here. The number of hidden layers ( $l = 1, 3, 5$ , and  $7$ ) and the number of units per layer ( $k = 16, 32,$   
 145  $64, 128, 256, 512, 1024, 2048$ , and  $4096$ ) are investigated to determine the optimal network  
 146 structure for this application. Note that the number of units per layer  $k$  is assumed to be constant  
 147 for each hidden layer to simplify the optimization, and, as such, each network structure is identified  
 148 as  $l \times k$ .

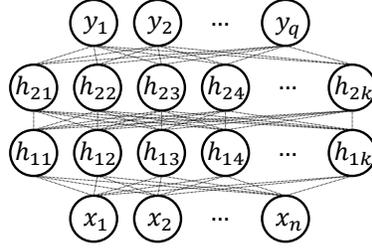


Figure 1. An example MLP model structure with  $l = 2$  hidden layers and  $k$  units per hidden layer.

### 2.3 RNN-GRU

An RNN-GRU model is structured similarly to the MLP model (see Figure 1), but with the output of each layer  $\mathbf{h}_t$  expressed as,

$$\mathbf{u}_t = \sigma(\mathbf{x}_t \cdot \mathbf{W}_{ux} + \mathbf{h}_{t-1} \cdot \mathbf{W}_{uh} + \mathbf{b}_u) \quad (3)$$

$$\mathbf{r}_t = \sigma(\mathbf{x}_t \cdot \mathbf{W}_{rx} + \mathbf{h}_{t-1} \cdot \mathbf{W}_{rh} + \mathbf{b}_r) \quad (4)$$

$$\tilde{\mathbf{h}}_t = g(\mathbf{x}_t \cdot \mathbf{W}_{hx} + (\mathbf{r}_t * \mathbf{h}_{t-1}) \cdot \mathbf{W}_{hh} + \mathbf{b}_h) \quad (5)$$

$$\mathbf{h}_t = (\mathbf{1} - \mathbf{u}_t) * \mathbf{h}_{t-1} + \mathbf{u}_t * \tilde{\mathbf{h}}_t \quad (6)$$

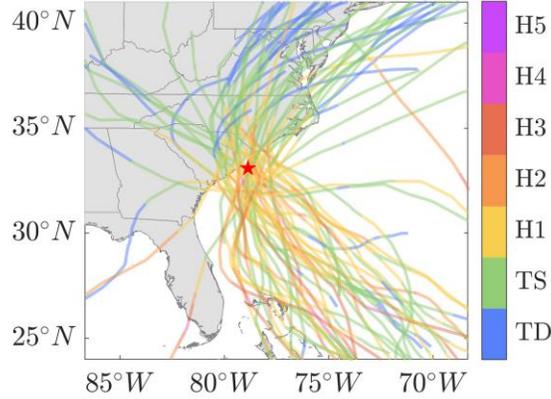
where  $\mathbf{W}_{ux}, \mathbf{W}_{uh}, \mathbf{W}_{rx}, \mathbf{W}_{rh}, \mathbf{W}_{hx}, \mathbf{W}_{hh}$  are weighting matrices,  $\mathbf{b}_u, \mathbf{b}_r, \mathbf{b}_h$  are bias vectors,  $\sigma$  is the recurrent activation function,  $*$  indicates elementwise multiplication between two vectors, and  $t$  indicates the time instance. For the first time instance  $t = 1$ , the vector  $\mathbf{h}_0$  at each layer is initialized as a zero vector and the vector  $\mathbf{h}_1$  at each layer is calculated according to Eq. 6, which is then used as the vector  $\mathbf{h}_{t-1}$  to begin predictions at time instance  $t = 2$ , and so on. The vector  $\mathbf{u}_t$  in Eq. 6 represents the percent of the past information (i.e.,  $\mathbf{h}_{t-1}$ ) to be updated,  $\mathbf{r}_t$  indicates the percent of the past information to forget, and  $\tilde{\mathbf{h}}_t$  represents the memory content.

The same 36 network structures are tested for the RNN-GRU model as for the MLP model (i.e., all combinations of four values of  $l$  and nine values of  $k$ , see Table 1). The activation function  $g$  is chosen as hyperbolic tangent, the recurrent activation function  $\sigma$  is chosen as hard sigmoid, following common practice.

## 3. Database of metocean conditions during hurricanes

The database of metocean conditions used in this paper for training the surrogate models includes conditions during a set of synthetic hurricanes. The synthetic hurricanes are selected from a catalog developed by Liu [25], which uses historical hurricanes to characterize potential hurricane activity in the northeastern part of the Atlantic basin over a span of 100,000 years. The hurricanes are defined using the Holland model [26, 27] in terms of seven parameters: longitude and latitude of the hurricane eye location, the central atmospheric pressure, the radius to maximum wind speed, the translational velocity, the translational direction, and the B parameter. The wind speed and atmospheric pressure fields defined by the Holland model are then used as input to a numerical metocean model Mike 21 (refer to [28] for details). The Mike 21 model couples a hydrodynamic module, which simulates two-dimensional flows based on the depth-integrated, incompressible, Reynolds-averaged Navier-Stokes equations [29], and a spectral wave module, which simulates the growth, propagation, and decay of wind-generated waves and swells based on wave action conservation equations [3], to provide predictions on metocean conditions.

184 One offshore site near South Carolina is considered in this paper (see Figure 2). The coordinates of  
 185 this site are (78.87°W, 33.09°N) and the water depth is 17 m. All hurricanes in the catalog that  
 186 induce a mean wind speed of at least 33 m/s at an elevation of 10 m are simulated, resulting in a  
 187 total of 5,881 hurricanes over the 100,000-year span of the catalog. Four types of metocean  
 188 conditions are considered: the significant wave height  $H_s$ , the peak wave period  $T_p$ , the peak wave  
 189 direction  $\theta_{H_s}$ , and the sea surface elevation  $\eta$  (including both tide and storm surge). As such, the  
 190 database includes information from the 5,881 synthetic hurricanes with various durations and the  
 191 corresponding hourly values of  $H_s$ ,  $T_p$ ,  $\theta_{H_s}$ , and  $\eta$ .



192  
 193 *Figure 2. The location of the offshore site near South Carolina (red star) and the trajectories of*  
 194 *50 hurricanes randomly sampled from a total of 5,881 synthetic hurricanes included in the*  
 195 *hurricane database. The color of the trajectory indicates the hurricane intensity per the Saffir-*  
 196 *Simpson scale (TD stands for tropical depression and TS stands for tropical storm).*  
 197

## 198 4. Numerical experiments

199 Surrogate modeling is a mathematical mapping between input  $\mathbf{X}$  and output vectors  $\mathbf{Y}$ , i.e.,  $\mathbf{X} \rightarrow \mathbf{Y}$ .  
 200 For the hurricane database introduced in Section 3,  $\mathbf{Y}$  is a time-dependent vector of four metocean  
 201 conditions ( $H_s$ ,  $T_p$ ,  $\theta_{H_s}$ , and  $\eta$ ) during each hurricane. Five models are introduced in this section,  
 202 including three types of surrogate models (Kriging, MLP, and RNN-GRU), with two different time-  
 203 dependent structures considered for the Kriging model and two training set sizes for the MLP model.  
 204 Some key information for each of the five models is presented in Table 2. The time-dependent  
 205 structures are introduced in Section 4.1. The design of each of the input vectors  $\mathbf{X}$  is provided in  
 206 Section 4.2, and the details of model training and evaluation are provided in Section 4.3.

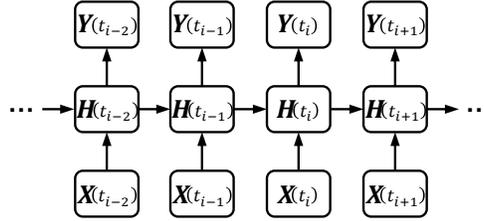
207 *Table 2. Summary of five models considered in this paper.*

	Surrogate model	Time-dependent structure	Number of training samples
Model 1	Kriging	$\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_j, \dots, t_{j+l})$	~5,000
Model 2	Kriging	$\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$	5,000
Model 3	MLP	$\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$	5,000
Model 4	MLP	$\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$	~660,000
Model 5	RNN-GRU	$\mathbf{X}(t_i) \rightarrow \mathbf{Y}(t_i)$	~660,000

208

### 209 4.1 Time-dependent structures

210 Modeling of metocean conditions during a hurricane can be abstracted as a sequence-to-sequence  
 211 problem (see Figure 3), where the time-dependent input  $\mathbf{X}$  represents a set of parameters  
 212 characterizing the hurricane and site-specific conditions, the time-dependent output  $\mathbf{Y}$  is the  
 213 metocean conditions of interest (e.g.,  $H_s$ ,  $T_p$ ,  $\theta_{H_s}$ , and  $\eta$  in this paper), and the hidden variables  $\mathbf{H}$   
 214 represent the complex interactions of metocean characteristics. The output at step  $t_i$  is affected by  
 215 not only  $\mathbf{X}(t_i)$ , but also by  $\mathbf{X}(t_{i-1})$ ,  $\mathbf{X}(t_{i-2})$ , etc.



216  
 217 *Figure 3. Schematic of the time-dependent structure of a sequence-to-sequence model, where*  
 218 *subscripts indicate the time instance,  $\mathbf{X}$  represents the input vector,  $\mathbf{H}$  represents a vector of*  
 219 *hidden variables, and  $\mathbf{Y}$  represents the output vector.*  
 220

221 For surrogate models without inherent time-dependent structures (e.g., the Kriging and MLP  
 222 models), the effect of time-dependence can be included with an *ad hoc* arrangement of the input  
 223 and/or output vectors. Two structures are compared in this paper. The first structure is expressed  
 224 as  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_j, \dots, t_{j+l})$ , a structure that stacks features (the vector  $\mathbf{X}$  in Figure 3) at steps  
 225  $t_{i-k}, \dots, t_i$  to form the input vector and predicts an output vector which includes steps  $t_j, \dots, t_{j+l}$   
 226 (see Figure 4(a)). The vector  $\mathbf{Y}(t_j, \dots, t_{j+l})$  is selected to cover the time instances corresponding to  
 227 intense metocean conditions, as these conditions are most important to model accurately for  
 228 engineering applications. The second structure is expressed as  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$ , a structure  
 229 that predicts metocean conditions independently for each time instance and includes the time-  
 230 dependence implicitly by stacking input features at various time instances (see Figure 4(b)). At first  
 231 glance, this structure is a special case of the first structure with  $l = 0$ . But, there is a philosophical  
 232 difference between the two structures: the second structure trains and predicts metocean conditions  
 233 for each hurricane hour, rather than for each hurricane. As such, each hurricane produces multiple  
 234 training samples, and the duration of predictions is not constant. It is worth noting that the structure  
 235  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_j, \dots, t_{j+l})$  can also be implemented in a step-wise manner, i.e., generating  
 236 multiple training samples for each hurricane,  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_j, \dots, t_{j+l})$ ,  
 237  $\mathbf{X}(t_{i-k+1}, \dots, t_{i+1}) \rightarrow \mathbf{Y}(t_{j+1}, \dots, t_{j+l+1})$ , etc. However, this results in multiple predictions for  
 238 metocean conditions at the same time instance. Thus, the implementation of the first structure in  
 239 this paper is based on the idea that one hurricane produces one training sample. Also note that, for  
 240 both structures, the time interval of the input features does not have to match the time interval of  
 241 the training data. For example, the hurricane database introduced in Section 2 has hourly intervals,  
 242 but input features can be stacked for time instances with 3-hour intervals. This benefits model  
 243 performance in some cases, as metocean characteristics with 3-hour intervals are less correlated  
 244 with each other. Both structures are implemented using the Kriging model (see Model 1 and Model  
 245 2 in Table 2), while the second structure is also implemented using the MLP model (see Model 3  
 246 and Model 4 in Table 2).

247 For the RNN-GRU model, the prediction of time-dependent processes is straightforward because  
 248 it inherently models time-dependent processes. The RNN-GRU model considers input vectors at

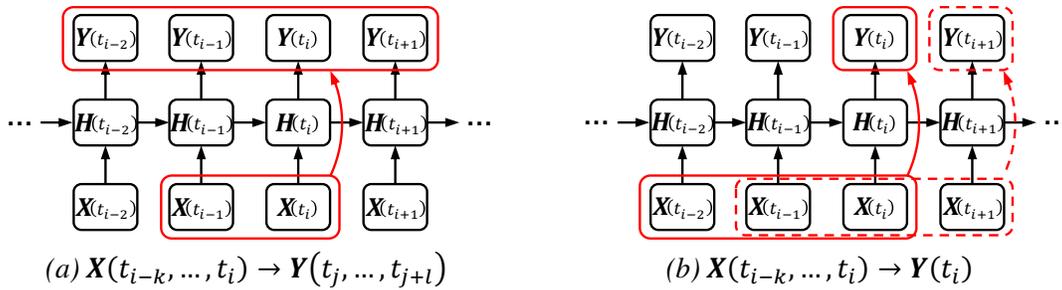
249 each time instance and makes predictions with the following the structure  $\mathbf{X}(t_i) \rightarrow \mathbf{Y}(t_i)$ . This  
 250 structure is considered in Model 5 in Table 2. All three time-dependent structures are summarized  
 251 in Table 3.

252 *Table 3. Summary of three time-dependent structures considered in this paper.*

Time-dependent structure	Requires surrogate models with time-dependent characteristics?	Training samples per hurricane	Model # in Table 2
$\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_j, \dots, t_{j+l})$	No	one	1
$\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$	No	multiple	2~4
$\mathbf{X}(t_i) \rightarrow \mathbf{Y}(t_i)$	Yes	multiple	5

253

254



255

256

257

258 *Figure 4. Two structures to predict time-dependent processes using surrogate models without*  
 259 *inherent time-dependent characteristics: (a) prediction of one output vector composed of various*  
 260 *time instances using input features at multiple time instances, and (b) prediction of output at each*  
 261 *time instance using input features at multiple time instances. Red rectangles indicate the inputs*  
 262 *and outputs selected to form the input and output vectors.*

263

#### 264 4.2 Design of input vector

265 Two aspects are involved when designing the input vector: the selection of features included at  
 266 each time instance (i.e., the specific form of  $\mathbf{X}(t_i)$  in Figure 4) and the time instances to stack these  
 267 features (i.e., the selection of  $(t_{i-k}, \dots, t_i)$  in Figure 4). The former affects all five models, and the  
 268 latter affects only Models 1~4 (see Table 2), as the time-dependent structure of Model 5 does not  
 269 require such stacking. Since the target output vector  $\mathbf{Y}$  is simulated from a numerical model driven  
 270 by wind and pressure fields, the seven hurricane parameters that define the wind and pressure fields  
 271 and the water depth (sum of still water depth and tide level) are the most straightforward features  
 272 to include in the input vector. Other features, including the maximum wind speed within the entire  
 273 wind field, local wind speed at the selected site, and local wind direction, are also considered. The  
 274 two circular variables, the hurricane translational direction and local wind direction, are expressed  
 275 in terms of sinusoidal and cosinoidal values, as is common practice. As such, a total of 13 features  
 276 are considered for each time instance.

277 For Models 1~4, the 13 input features are stacked at various time instances to form the input vector.  
 278 Up to 9 time instances with 3-hour intervals are selected (i.e.,  $t_i, t_{i-3}, \dots, t_{i-24}$ ). For Model 1, only  
 279 one training sample is extracted for each hurricane, and  $t_i$  is selected as the hour of the maximum  
 280  $V$  at the selected site, and the output time instances  $(t_j, \dots, t_{j+l})$  include the 12 hours before and

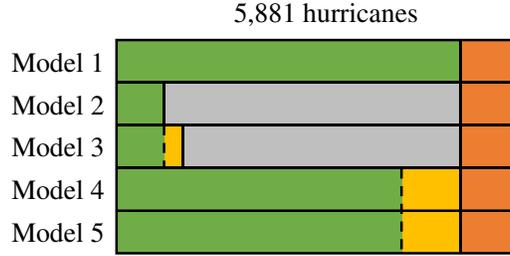
281 after  $t_i$  (i.e., 25 time instances in total). The design of the input vector directly affects the  
282 effectiveness of the surrogate model, and much research has been devoted to optimizing the input  
283 vector [30, 31]. In preliminary experiments, fewer time instances and features at each time instance  
284 are tested, and the results indicate that using all 13 features at each time instance always yields the  
285 best prediction performance for Models 1~5. For Models 1, 3, and 4, stacking the 13 features at all  
286 9 time instances yields the best prediction performance, while for Model 2, no stacking of the 13  
287 features (i.e., using only the 13 features at  $t_i$  as the input vector) yields the best prediction  
288 performance.

### 289 4.3 Model training and evaluation

290 The five models considered in this paper use three types of surrogate models and three time-  
291 dependent structures, see Table 1 and Table 3. As such, the training and evaluation processes are  
292 slightly different for each model. For the 5,881 hurricanes included in the database of metocean  
293 conditions, 881 hurricanes are used for model evaluation (see Figure 5), leaving 5,000 hurricanes  
294 (~800,000 hurricane hours) for model training. The Kriging models are trained using the DACE  
295 package [20] in MATLAB, and the Neural Networks are trained using TensorFlow [32]. The  
296 training datasets are described for each prediction model as follows.

- 297 • For Model 1, all 5,000 hurricanes are used for training, producing 5,000 training samples.
- 298 • For Model 2, model training is based on hurricane hours rather than hurricanes, and  
299 therefore 5,000 hurricane hours are randomly selected from the 5,000 hurricanes as the  
300 training dataset. According to the result of the random selection, a total of ~3,000  
301 hurricanes contribute to the 5,000 hurricane hours. The same number of 5,000 training  
302 samples is selected for Model 2 because (1) this allows a fair comparison with Model 1 to  
303 reveal the impact of the time-dependent structure on predictions and (2) it is intractable for  
304 the Kriging model to memorize ~800,000 training samples formed in terms of hurricane  
305 hours, as it takes ~5,000 Gigabytes of computer memory just to construct the  $\mathbf{R}$  matrix in  
306 MATLAB using double-precision floating-point values for ~800,000 training samples.
- 307 • For Model 3, 5,000 training samples are used again to train the MLP model to create a fair  
308 comparison with the Kriging model in Model 2. Since there are more hyperparameters for  
309 Neural Networks than Kriging models, the chance of overfitting (i.e., the model learns not  
310 only the general trend but also the local, noise-like variations, resulting in testing  
311 performance that is much worse than the training performance) is higher. To prevent  
312 overfitting, a common approach is employed: a validation dataset is prepared in addition  
313 to the training dataset, and optimal hyperparameters are selected based on the performance  
314 of the model for the validation dataset, rather than for the training dataset. The ratio  
315 between the training dataset the validation dataset used here is around 70:15. As such, a  
316 total of 1,070 samples of hurricane hours are used for validation. The training and  
317 validation datasets are shuffled and re-divided during the training process to improve  
318 training efficiency.
- 319 • For Model 4, the ~800,000 hurricane hours are divided into ~660,000 samples for training  
320 and ~140,000 samples for validation according to the 70:15 ratio. The training process is  
321 the same as for Model 3.
- 322 • For Model 5, the training samples are formed in terms of hurricane hours, but the training  
323 process is done in terms of hurricanes, as time instances are fed into the model in sequence  
324 for each hurricane, see Section 2.3. As such, training and validation datasets are divided  
325 based on individual hurricanes.

326 The division of the 5,881 hurricanes in the catalog among training, validation, and testing datasets  
 327 is illustrated in Figure 5. All models are tested for the same 881 hurricanes.



328  
 329 *Figure 5. Division of the training (in green), validation (in yellow), and testing (in orange)*  
 330 *datasets for each model.*

## 332 5. Results and discussions

333 For the four types of metocean conditions ( $H_s$ ,  $T_p$ ,  $\theta_{H_s}$ , and  $\eta$ ), separate prediction models are  
 334 trained and results of the  $H_s$  predictions are used as an example to compare the models. The overall  
 335 performance of the models is presented in Section 5.1, and detailed comparisons are discussed in  
 336 Section 5.2.

### 337 5.1 Overall performance

338 Models 1~5 are trained in different ways but are tested for the same 881 testing hurricanes. The  
 339 root-mean-square error for the testing dataset is chosen as the metric to evaluate the performance  
 340 of each model, expressed as,

$$341 \quad \sigma_{Test} = \sqrt{\frac{1}{N} \sum_{i=1}^m \sum_{j=1}^{n_i} (\hat{y}_{i,j} - y_{i,j})^2} \quad (7)$$

342 where  $\hat{y}$  is the prediction value during a time instance of a hurricane,  $y$  is the corresponding true  
 343 value from the numerical time-history simulation, subscript  $j$  indicates the prediction time instance,  
 344 subscript  $i$  indicates the testing hurricane,  $m$  is the total number of testing hurricanes, which is 881  
 345 in this case,  $n_i$  is the total time instances for each testing hurricane, and  $N$  indicates the total time  
 346 instances included in the testing hurricanes. Note that for the same hurricane,  $n_i$  is different for  
 347 each prediction model. For instance, Model 1 provides predictions for 25 time instances regardless  
 348 of the hurricane duration, while Models 2~4 provide predictions for a duration slightly shorter than  
 349 the hurricane, since the mapping of  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$  determines that the first available time  
 350 instance for prediction is  $t_{k+1}$ ; Model 5 provides predictions for the entire hurricane duration. As  
 351 such, the number of time instances involved in  $\sigma_{Test}$  is different for each prediction model.

352 Each prediction model is tested for various sets of hyperparameters (see Table 1). The best  
 353 performance for predicting  $H_s$  and the corresponding hyperparameters are listed in Table 4. Overall,  
 354 Model 1 performs the worst, with  $\sigma_{H_s, Test} = 0.41$  m, and Model 5 performs the best, with  
 355  $\sigma_{H_s, Test} = 0.05$  m. The model performance is revealed with more details in Figure 6, which plots  
 356  $\sigma_{H_s, Test}$  versus  $H_s$ . It is interesting to note that  $\sigma_{H_s, Test}$  for Models 1 and 2 differs by ~32% as listed

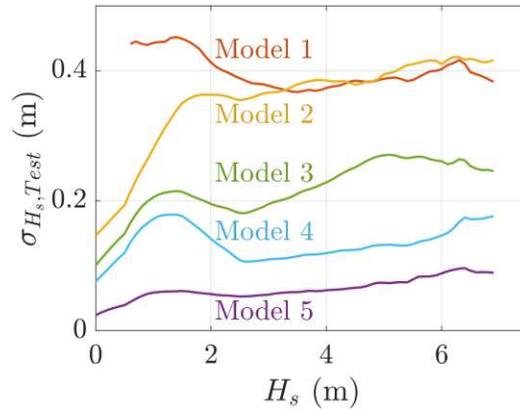
357 in Table 4, while the  $H_S \sim \sigma_{H_S, \text{Test}}$  curves behave quite similar for  $H_S > 2$  m. This is because the  
 358 values of  $H_S$  predicted by Model 1 are all relatively high because of the way the hurricane duration  
 359 is defined for Model 1, while the predictions of Model 2 cover almost the entire history of each  
 360 hurricane. For the latter case, the majority of  $H_S$  values are low when the hurricane is far away, and  
 361 the low values of  $\sigma_{H_S, \text{Test}}$  for  $H_S < 2$  m lead to a lower value of  $\sigma_{H_S, \text{Test}}$ . For risk analysis, which  
 362 focuses on extreme conditions, a lower prediction error at high values of  $H_S$  is preferable compared  
 363 to the trend in Figure 6, and this can be taken into account by including more intense hurricanes in  
 364 the training database and by adjusting the training process to increase the weight of high values of  
 365  $H_S$  during the model training.

366  
 367  
 368

*Table 4. The best prediction performance on the testing dataset and the corresponding network structure for the five models in Table 1.*

	$\sigma_{H_S, \text{Test}}$ (m)	Network structure
Model 1	0.41	-
Model 2	0.36	-
Model 3	0.19	$3 \times 1024$
Model 4	0.14	$5 \times 32$
Model 5	0.05	$3 \times 128$

369



*Figure 6. Prediction errors as a function of  $H_S$ .*

370  
 371  
 372  
 373  
 374  
 375  
 376  
 377  
 378

Prediction performance of Model 5, which performs the best for  $H_S$  among the five prediction models, is provided in Table 5 for the other three metocean conditions ( $T_p$ ,  $\theta_{H_S}$ , and  $\eta$ ). Note that for the prediction of  $\theta_{H_S}$ , which is a circular variable, the output variable is selected as  $\sin \theta_{H_S}$  and  $\cos \theta_{H_S}$ , so that values referring to the same direction (e.g.,  $-180^\circ$  and  $180^\circ$ ) have the same representation. However,  $\sigma_{T_p, \text{Test}}$  is evaluated based on the resulting direction errors for the range between  $-180^\circ$  and  $180^\circ$ .

379  
 380  
 381  
 382  
 383  
 384

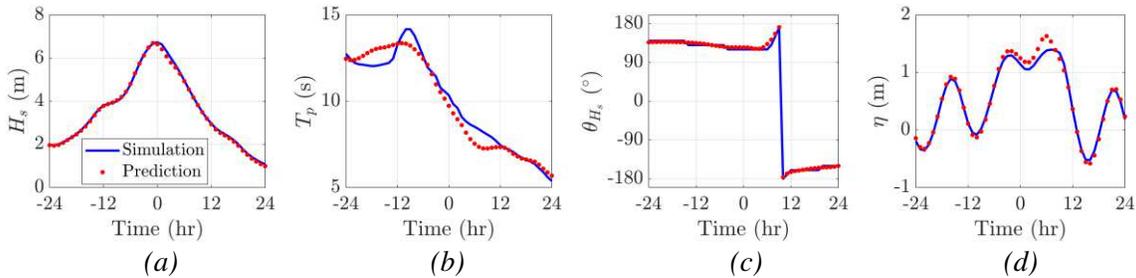
To better illustrate the level of accuracy of Model 5 on various metocean conditions, prediction results for an individual hurricane that has a similar value  $\sigma_{\text{Test}}$  as the overall value provided in Table 4 are presented in Figure 7, with  $\sigma_{\text{Test}}$  for  $H_S$ ,  $T_p$ ,  $\theta_{H_S}$ , and  $\eta$  equal to 0.04 m, 1.52 s,  $13.5^\circ$ , and 0.06 m, respectively. Note that the duration of the hurricane plotted in Figure 7 is 163 hours, but only the results for the 24 hours before and after the maximum  $H_S$  are shown here. The prediction of  $H_S$  matches the details of the simulation, while the prediction of  $T_p$  captures only the

385 overall trend of the simulation. Because  $\theta_{H_s}$  is decomposed into sinusoidal and cosinoidal values  
 386 during model training, the wrapping effect is captured (i.e., values higher than  $180^\circ$  reset to -  
 387  $180^\circ$ ) clearly. Note that Figure 7(c) shows a lower prediction error compared to the overall  $\sigma_{\theta_{H_s}}$   
 388 of  $13.5^\circ$ . This is because of some large prediction errors when the hurricane is far away from the  
 389 site of interest. An opposite situation is observed in Figure 7(d), where an overestimation at the  
 390 peak surge is not reflected in the relatively low value of  $\sigma_\eta$  for this hurricane. These situations could  
 391 be clarified with an alternative metric for performance, such as  $\sigma_{\text{Test}}$  for the time instances when  
 392 the hurricane is close to the site of interest.

393 *Table 5. The best prediction performance of Model 5 on the testing dataset and the*  
 394 *corresponding network structures.*

	Network structure	$\sigma_{\text{Test}}$
$T_p$	3×256	1.38 s
$\theta_{H_s}$	5×512	15.4°
$\eta$	7×128	0.05 m

395



396

397

398 *Figure 7. Time history prediction of a)  $H_s$ , b)  $T_p$ , c)  $\theta_{H_s}$ , and d)  $\eta$  for a testing hurricane. Hour*  
 399 *zero is selected as the time of maximum  $H_s$ .*

400

401

## 5.2 Detailed comparisons

402

Some key aspects of the five prediction models are summarized as follows to clarify the validity of  
 403 comparisons among the models.

404

- Model 1 and Model 2 both use the Kriging model and are trained using 5,000 training samples. Model 1 is trained based on hurricanes (i.e., one training sample per hurricane), but Model 2 is trained based on hurricane hours (i.e., multiple training samples per hurricane).

408

- Model 2 and Model 3 use the same number of training samples and time-dependent structure. Model 2 uses the Kriging model, but Model 3 uses the MLP model, which is the most basic form of Neural Networks.

411

- Model 3 and Model 4 use the MLP model and the same time-dependent structure. Model 3 is trained using 5,000 training samples, while Model 4 is trained using ~660,000 training samples.

414

- Model 4 and Model 5 are trained using the same ~660,000 training samples. Model 4 uses the MLP model, which cannot inherently represent time-dependent behavior and relies on stacking input features at multiple time instances to represent such behavior. Model 5 uses the RNN-GRU model, which predicts the time-dependence of metocean conditions using its inherent recurrent structure.

415

416

417

418

### 419 5.2.1 Effect of time-dependent structure

420 The Kriging model is used to implement two structures of time-dependence as represented by  
421 Models 1 and 2, which can be expressed as  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_j, \dots, t_{j+l})$  and  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow$   
422  $\mathbf{Y}(t_i)$ . As revealed in Figure 6, these two models perform similarly for  $H_s > 2$  m. However, Model  
423 2 outperforms Model 1 in two ways. First, Model 2 is more flexible in predicting hurricanes with  
424 varying duration, as Model 1 only provides predictions with a fixed duration. Second, Model 2 is  
425 more efficient in terms of how the simulated hurricanes are used for training. Even though the same  
426 number of training samples are used, Model 1 uses all 5,000 simulated hurricanes, while Model 2  
427 uses only 5,000 hurricane hours (contributed by  $\sim 3,000$  hurricanes), which is a small fraction of  
428 the  $\sim 800,000$  total hurricane hours from the 5,000 hurricanes. The efficiency of these two strategies  
429 reflects the difference in what the models are learning from the training data. For Model 1, one set  
430 of input features related to the time instance of maximum local wind speed is used to predict the  
431 corresponding time history output, while for Model 2, each set of input features is used to predict  
432 the output at a corresponding time instance. As such, Model 1 learns how to represent metocean  
433 conditions during the *maximum* local wind input of a hurricane, while Model 2 learns to represent  
434 metocean conditions during *any* wind input of a hurricane.

### 435 5.2.2 Kriging vs. Neural Networks

436 The time-dependent structure  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$  is highly efficient, resulting in a large number  
437 of training samples, as the number of samples is approximately equal to the total hurricane hours  
438 instead of the number of hurricanes. Kriging models must memorize the entire training dataset, and  
439 thus the training process becomes intractable for large training datasets. There are some approaches  
440 (such as the adaptive Kriging combined with importance sampling method [33]) to improve  
441 efficiency in designing the training dataset, however, for a given training database, the comparison  
442 between Model 2 and Model 3 clearly shows how the type of surrogate model affects the prediction  
443 performance. The same number of 5,000 training samples are used in Model 2 and Model 3, and  
444 Model 3 using MLP lowers  $\sigma_{H_s, \text{Test}}$  by 32% compared to Model 2 using Kriging. The improvement  
445 is even more pronounced for high values of  $H_s$ , as shown in Figure 6. The MLP model outperforms  
446 Kriging because its capacity to approximate a nonlinear process can be easily adjusted through the  
447 network structure, while Kriging models provide limited hyperparameters (i.e., the selection of  
448 regression and correlation functions) to control its ability to approximate the nonlinear behavior of  
449 a process.

450 Another important difference between the Kriging models and the Neural Networks is that for the  
451 input included in the training dataset, Kriging models always provide an exact prediction for the  
452 corresponding training output, much like interpolation; while Neural Networks make predictions  
453 with some deviation from the corresponding training output, much like regression. For predictions  
454 of a spatial variable, where Kriging models are widely used, this characteristic is attractive because  
455 the input is well defined by the spatial coordinates. For hurricane metocean conditions, however, a  
456 regression-type behavior is more attractive, because the characteristics of a hurricane can rarely be  
457 well defined by several parameters at several time instances, i.e., the same hurricane input features  
458 can be used to describe different hurricanes, leading to variation in metocean conditions. Though  
459 there are ways to account for such variation in Kriging models [34], additional hyperparameters  
460 are required.

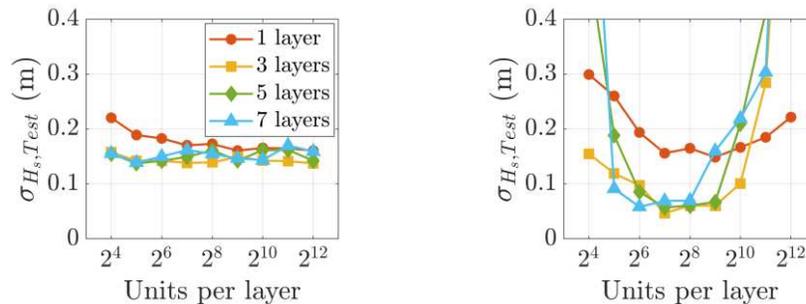
### 461 5.2.3 Size of the training dataset

462 One advantage of Neural Networks compared to Kriging models is that model training is no longer  
 463 constrained by the size of the training dataset. Model 4 uses the same training process as Model 3,  
 464 except that the entire ~660,000 training samples are used. The resulting overall value of  $\sigma_{H_s, \text{Test}}$  is  
 465 reduced by 26%, and  $\sigma_{H_s, \text{Test}}$  is almost reduced by half for relatively high values of  $H_s$  (see Figure  
 466 6).

467 The training of Neural Networks effectively minimizes the training error. Due to the non-convex  
 468 characteristics of the loss function, the mini-batch gradient descent algorithm is used as a standard  
 469 approach in Deep Learning, which is also used in this paper. This algorithm uses a small number  
 470 of training samples to estimate the gradient for minimizing the loss function and is more efficient  
 471 and suitable when the size of the training data is large [35]. This also allows the model to be easily  
 472 updated when new training samples are available, instead of re-training a model from scratch.

#### 473 5.2.4 Neural Networks

474 Both Models 4 and 5 can be represented with the schematic in Figure 1. Model 4 uses a simple  
 475 operation for each hidden layer as expressed in Eq. 2 and represents time-dependent processes by  
 476 including hurricane features from previous time instances in the input layer. As such, the network  
 477 learns the relationship between metocean conditions at each time instance and hurricane features  
 478 at multiple time instances. Model 5 uses a much more complex operation for hidden layers as  
 479 expressed in Eqs. 3~6 so that each hidden layer manages its own memory vector  $\tilde{\mathbf{h}}_t$  representing  
 480 the prediction history. The overall value of  $\sigma_{H_s, \text{Test}}$  for Model 5 is lowered by 64% compared to  
 481 Model 4, and Figure 6 reveals that the values of  $\sigma_{H_s, \text{Test}}$  for Model 5 are about half of Model 4 for  
 482 relatively high values of  $H_s$ . This suggests that using a network with an intrinsic time-dependent  
 483 structure performs better than manually stacking input features. However, the better performance  
 484 of Model 5 comes with a price. As shown in Figure 8, the performance of Model 5 is more sensitive  
 485 to the network structure compared to Model 4. A U-shaped behavior is observed in Figure 8(b) for  
 486 results based on networks with constant depth and varying width. The reason for this behavior is  
 487 complex and is mainly due to the non-convex loss function within Neural Networks. Based on a  
 488 study of the training and validation details, which is not presented here, some deep and narrow  
 489 networks (e.g., a  $5 \times 16$  network) converge to a local minimum of the loss function (i.e., a global  
 490 minimum exists but is missed by the training algorithm) and thus perform worse compared to a  
 491 shallower network with the same width. For a wide network (e.g., one with 4,096 units per hidden  
 492 layer), when the number of hidden layers is large (starting from 3 hidden layers in this case), the  
 493 training process becomes intractable due to the network complexity. For intermediate numbers of  
 494 units per layer, a relatively clear pattern is observed: the prediction performance is insensitive to  
 495 the number of hidden layers as long as more than one hidden layer is used, and the optimal number  
 496 of units per layer is around 128. As such, it takes significantly more effort to test the network  
 497 structure of the RNN-GRU model to reach its optimal performance compared to MLP.



498

499  
500  
501

(a)

(b)

Figure 8. The impact of the network width on model performance for a) Model 4 and b) Model 5.

## 502 6. Conclusions

503 Multiple approaches to apply surrogate models to predict time-dependent metocean conditions  
504 during hurricanes are discussed and implemented in this paper. A hurricane database composed of  
505 numerical metocean simulations for synthetic hurricanes is used for training and testing the  
506 surrogate models.

507 Some surrogate models, such as Kriging and Multilayer Perceptron, do not have the inherent ability  
508 to model time-dependent processes. However, by stacking inputs and/or outputs at various time  
509 instances, these models can be adapted to model time-dependent processes. Two adaptations are  
510 evaluated in this paper:  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_j, \dots, t_{j+l})$  and  $\mathbf{X}(t_{i-k}, \dots, t_i) \rightarrow \mathbf{Y}(t_i)$ . The latter is  
511 found to utilize training data more efficiently as multiple training samples are produced by each  
512 hurricane. In addition, for these adapted models, larger sizes of the training dataset are found to  
513 improve prediction performance, but also increase computational demands, an issue that is  
514 especially pronounced for memory-based models such as Kriging.

515 Neural Networks are demonstrated to have great potential for time-dependent surrogate modeling  
516 for metocean conditions. They outperform the Kriging model in several respects. First, the RNN-  
517 GRU model illustrates extraordinary ability in predicting four metocean variables (significant wave  
518 height, peak wave period, peak wave direction, and sea surface elevation) during hurricanes (see  
519 Table 5 and Figure 7), an ability that can lower uncertainty when applying surrogate modeling in  
520 risk analysis and other tasks. Second, the complexity of Neural Networks can be adjusted easily  
521 through the network structure, which enables the models to learn the complex behavior of metocean  
522 conditions accurately. Lastly, the optimization algorithm for Neural Network can consider large  
523 training datasets efficiently, facilitating the processes of model training and updating. The  
524 flexibility of Neural Networks, however, makes their performance sensitive to the hyperparameters,  
525 and the complex RNN-GRU model implemented in this paper is shown to be especially sensitive.

526 The results of this paper strongly demonstrate the potential of Neural Networks and Deep Learning  
527 to represent complex metocean conditions during hurricanes, a potential that could transform the  
528 way hurricane risk is assessed.

529

## 530 Acknowledgment

531 The authors wish to thank Dr. Weichiang Pang for providing the synthetic hurricane catalog. This  
532 manuscript is based upon work supported by the National Science Foundation under Grant No.  
533 CMMI-1552559 and Northeastern University. Any opinions, findings, and conclusions expressed  
534 in this material are those of the authors and do not necessarily reflect the views of the National  
535 Science Foundation or other sponsors.

536

## 537 Reference

- 538 [1] I. R. Young, "Parametric Hurricane Wave Prediction Model," *J Waterw Port C-Asce*, vol.  
539 114, no. 5, pp. 637-652, 1988. [Online]. Available: <Go to ISI>://WOS:A1988P956200006.
- 540 [2] H. L. Tolman, *User manual and system documentation of WAVEWATCH III v4.18*. 2014.
- 541 [3] DHI, *MIKE21 SW Scientific Documentary*. Denmark: DHI Water and Environment, 2014.
- 542 [4] J. Mahjoobi and A. Etemad-Shahidi, "An alternative approach for the prediction of  
543 significant wave heights based on classification and regression trees," *Appl Ocean Res*, vol.  
544 30, no. 3, pp. 172-177, 2008.
- 545 [5] A. A. Taflanidis, G. Jia, A. B. Kennedy, and J. M. Smith, "Implementation/optimization of  
546 moving least squares response surfaces for approximation of hurricane/storm surge and  
547 wave responses," *Natural Hazards*, vol. 66, no. 2, pp. 955-983, 2012, doi: 10.1007/s11069-  
548 012-0520-y.
- 549 [6] G. F. Jia and A. A. Taflanidis, "Kriging metamodeling for approximation of high-  
550 dimensional wave and surge responses in real-time storm/hurricane risk assessment," (in  
551 English), *Computer Methods in Applied Mechanics and Engineering*, vol. 261, pp. 24-38,  
552 Jul 15 2013, doi: 10.1016/j.cma.2013.03.012.
- 553 [7] S. C. James, Y. Zhang, and F. O'Donncha, "A machine learning framework to forecast  
554 wave conditions," *Coast Eng*, vol. 137, pp. 1-10, 2018.
- 555 [8] G. F. Jia, A. A. Taflanidis, N. C. Nadal-Caraballo, J. A. Melby, A. B. Kennedy, and J. M.  
556 Smith, "Surrogate modeling for peak or time-dependent storm surge prediction over an  
557 extended coastal region using an existing database of synthetic storms," *Natural Hazards*,  
558 vol. 81, no. 2, pp. 909-938, Mar 2016, doi: 10.1007/s11069-015-2111-1.
- 559 [9] R. van der Merwe, T. K. Leen, Z. Lu, S. Frolov, and A. M. Baptista, "Fast neural network  
560 surrogates for very high dimensional physics-based models in computational  
561 oceanography," *Neural Netw*, vol. 20, no. 4, pp. 462-78, May 2007, doi:  
562 10.1016/j.neunet.2007.04.023.
- 563 [10] S. Mandal and N. Prabakaran, "Ocean wave forecasting using recurrent neural networks,"  
564 *Ocean Eng*, vol. 33, no. 10, pp. 1401-1410, 2006.
- 565 [11] W. W. Hsieh and B. Tang, "Applying neural network models to prediction and data  
566 analysis in meteorology and oceanography," *B Am Meteorol Soc*, vol. 79, no. 9, pp. 1855-  
567 1870, 1998.
- 568 [12] P. Jain and M. Deo, "Neural networks in ocean engineering," *Ships and offshore structures*,  
569 vol. 1, no. 1, pp. 25-35, 2006.
- 570 [13] A. Herman, "NEURAL-NETWORK MODELING AND DATA ANALYSIS  
571 TECHNIQUES IN COASTAL HYDRODYNAMIC STUDIES—A REVIEW," L. L.  
572 Wright Ed.: Nova Science Publishers, 2011.
- 573 [14] S.-W. Kim, J. A. Melby, N. C. Nadal-Caraballo, and J. Ratcliff, "A time-dependent  
574 surrogate model for storm surge prediction based on an artificial neural network using high-  
575 fidelity synthetic hurricane modeling," *Natural Hazards*, vol. 76, no. 1, pp. 565-585, 2014,  
576 doi: 10.1007/s11069-014-1508-6.
- 577 [15] A. Herman, R. Kaiser, and H. D. Niemeyer, "Wind-wave variability in a shallow tidal sea—  
578 Spectral modelling combined with neural network methods," *Coast Eng*, vol. 56, no. 7, pp.  
579 759-772, 2009.
- 580 [16] M. Browne, B. Castelle, D. Strauss, R. Tomlinson, M. Blumenstein, and C. Lane, "Near-  
581 shore swell estimation from a global wind-wave model: Spectral process, linear, and  
582 artificial neural network models," *Coast Eng*, vol. 54, no. 5, pp. 445-460, 2007.
- 583 [17] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep  
584 neural networks with multitask learning," in *Proceedings of the 25th international  
585 conference on Machine learning*, 2008: ACM, pp. 160-167.
- 586 [18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception  
587 architecture for computer vision," in *Proceedings of the IEEE conference on computer  
588 vision and pattern recognition*, 2016, pp. 2818-2826.

589 [19] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical  
590 machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

591 [20] S. Lophaven, H. Nielsen, and J. Sondergaard, "DACE: A Matlab Kriging Toolbox.,"  
592 *Technical University of Denmark*, 2002.

593 [21] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines,"  
594 in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010,  
595 pp. 807-814.

596 [22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint*  
597 *arXiv:1412.6980*, 2014.

598 [23] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning  
599 by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

600 [24] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop, coursera: Neural networks for machine  
601 learning," *University of Toronto, Technical Report*, 2012.

602 [25] F. Liu, "Projections of Future US Design Wind Speeds Due to Climate Change for  
603 Estimating Hurricane Losses," *Clemson University*, 2014.

604 [26] P. N. Georgiou, A. G. Davenport, and B. J. Vickery, "Design Wind Speeds in Regions  
605 Dominated by Tropical Cyclones," (in English), *J Wind Eng Ind Aerod*, vol. 13, no. 1-3,  
606 pp. 139-152, 1983, doi: Doi 10.1016/0167-6105(83)90136-8.

607 [27] G. J. Holland, "An Analytic Model of the Wind and Pressure Profiles in Hurricanes," (in  
608 English), *Mon Weather Rev*, vol. 108, no. 8, pp. 1212-1218, 1980, doi: 10.1175/1520-  
609 0493(1980)108<1212:aamotw>2.0.co;2.

610 [28] C. Qiao, A. Myers, and S. R. Arwade, "Validation of a 37-year Metocean Hindcast along  
611 the U.S. Atlantic Coast," presented at the 13th International Conference on Applications of  
612 Statistics and Probability in Civil Engineering, ICASP13, Seoul, South Korea, May, 2019.

613 [29] DHI, *MIKE21-FM HD Scientific Documentary*. Denmark: DHI Water and Environment,  
614 2014.

615 [30] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Machine Learning*  
616 *Proceedings 1992*: Elsevier, 1992, pp. 249-256.

617 [31] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine  
618 learning," *Artificial intelligence*, vol. 97, no. 1-2, pp. 245-271, 1997.

619 [32] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th*  
620 *{USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*,  
621 2016, pp. 265-283.

622 [33] W. Yun, Z. Lu, X. J. S. Jiang, and M. Optimization, "An efficient reliability analysis  
623 method combining adaptive Kriging and modified importance sampling for small failure  
624 probability," vol. 58, no. 4, pp. 1383-1393, 2018.

625 [34] R. B. Gramacy, *Surrogates: Gaussian Process Modeling, Design, and Optimization for the*  
626 *Applied Sciences*. CRC Press, 2020.

627 [35] M. Li, T. Zhang, Y. Chen, and A. J. Smola, "Efficient mini-batch training for stochastic  
628 optimization," in *Proceedings of the 20th ACM SIGKDD international conference on*  
629 *Knowledge discovery and data mining*, 2014: ACM, pp. 661-670.

630

## Figures

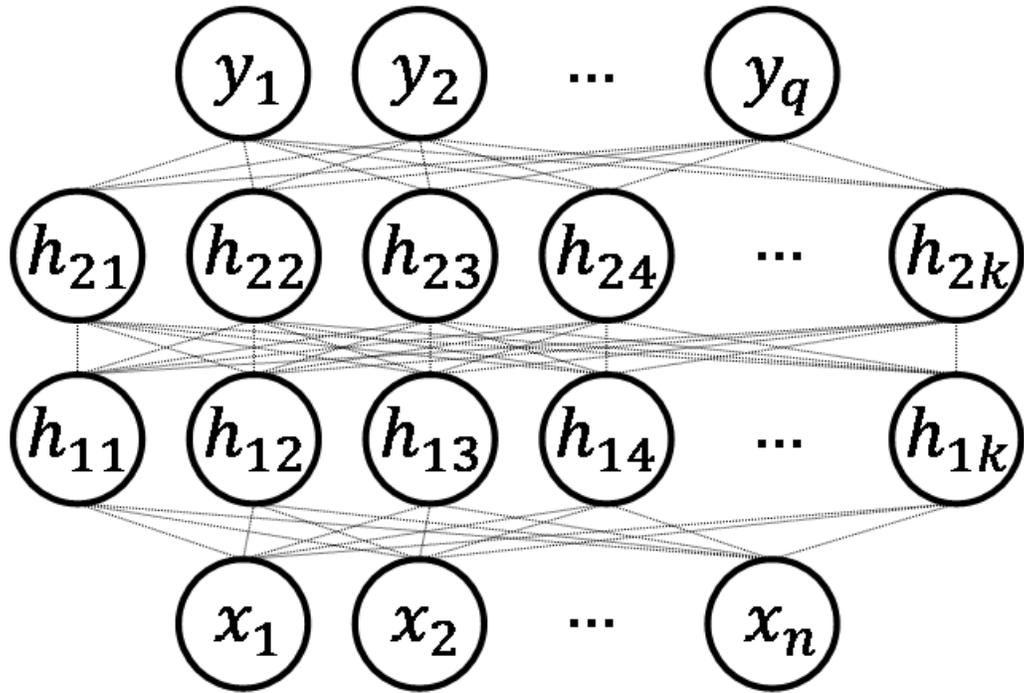


Figure 1

An example MLP model structure with  $l = 2$  hidden layers and  $k$  units per hidden layer.

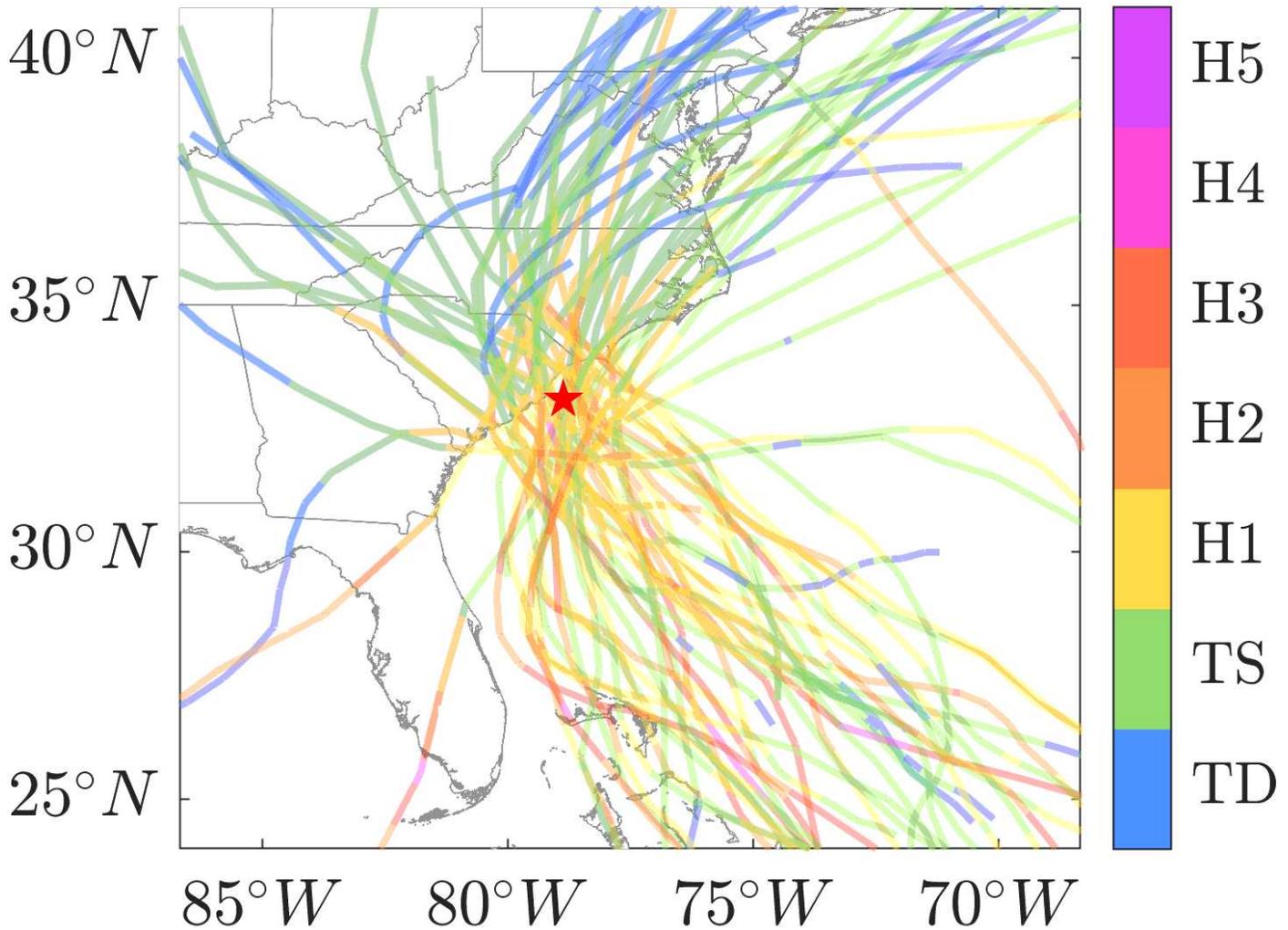


Figure 2

The location of the offshore site near South Carolina (red star) and the trajectories of 50 hurricanes randomly sampled from a total of 5,881 synthetic hurricanes included in the hurricane database. The color of the trajectory indicates the hurricane intensity per the Saffir-Simpson scale (TD stands for tropical depression and TS stands for tropical storm).

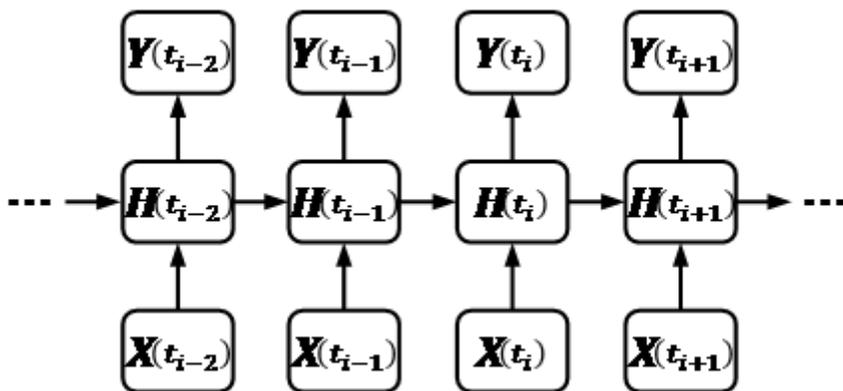


Figure 3

Schematic of the time-dependent structure of a sequence-to-sequence model, where subscripts indicate the time instance, X represents the input vector, H represents a vector of hidden variables, and Y represents the output vector

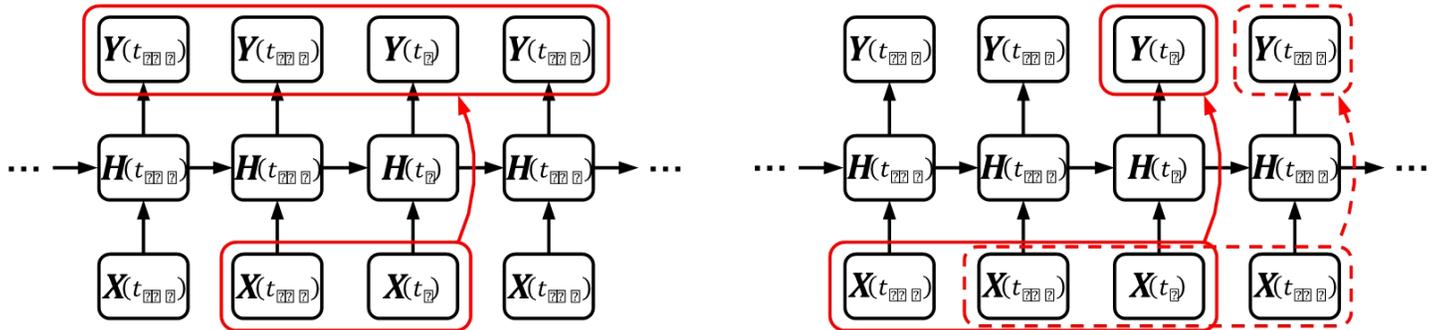


Figure 4

Two structures to predict time-dependent processes using surrogate models without inherent time-dependent characteristics: (a) prediction of one output vector composed of various time instances using input features at multiple time instances, and (b) prediction of output at each time instance using input features at multiple time instances. Red rectangles indicate the inputs and outputs selected to form the input and output vectors.

## 5,881 hurricanes

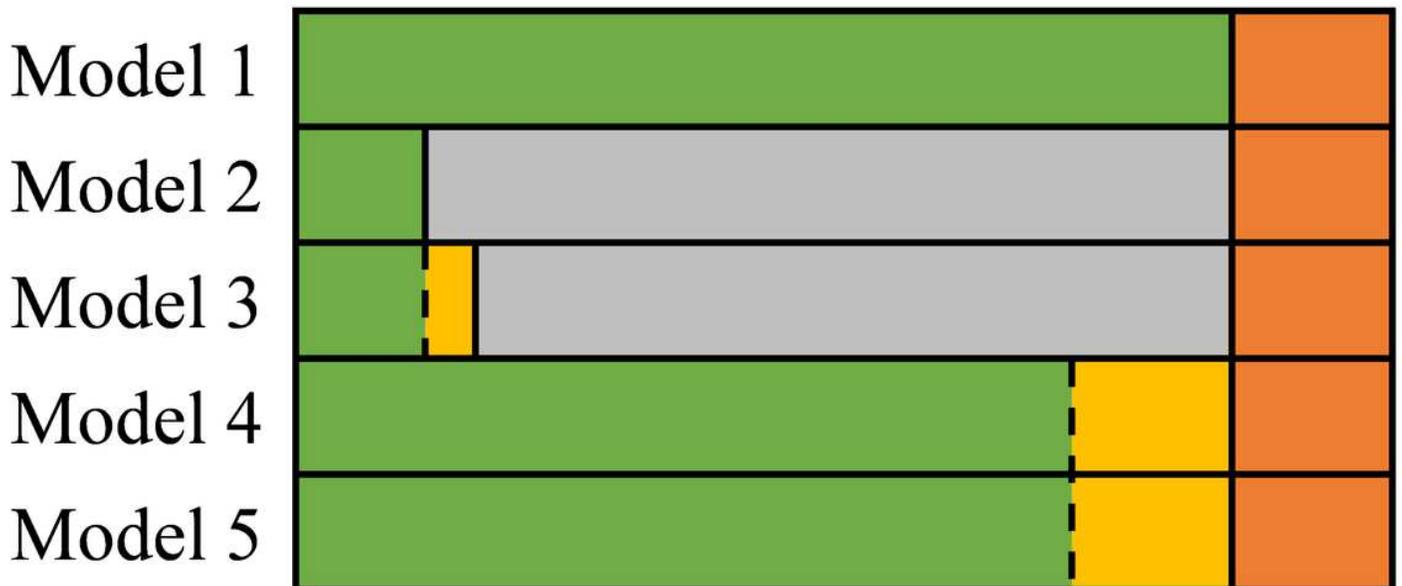


Figure 5

Division of the training (in green), validation (in yellow), and testing (in orange) datasets for each model.

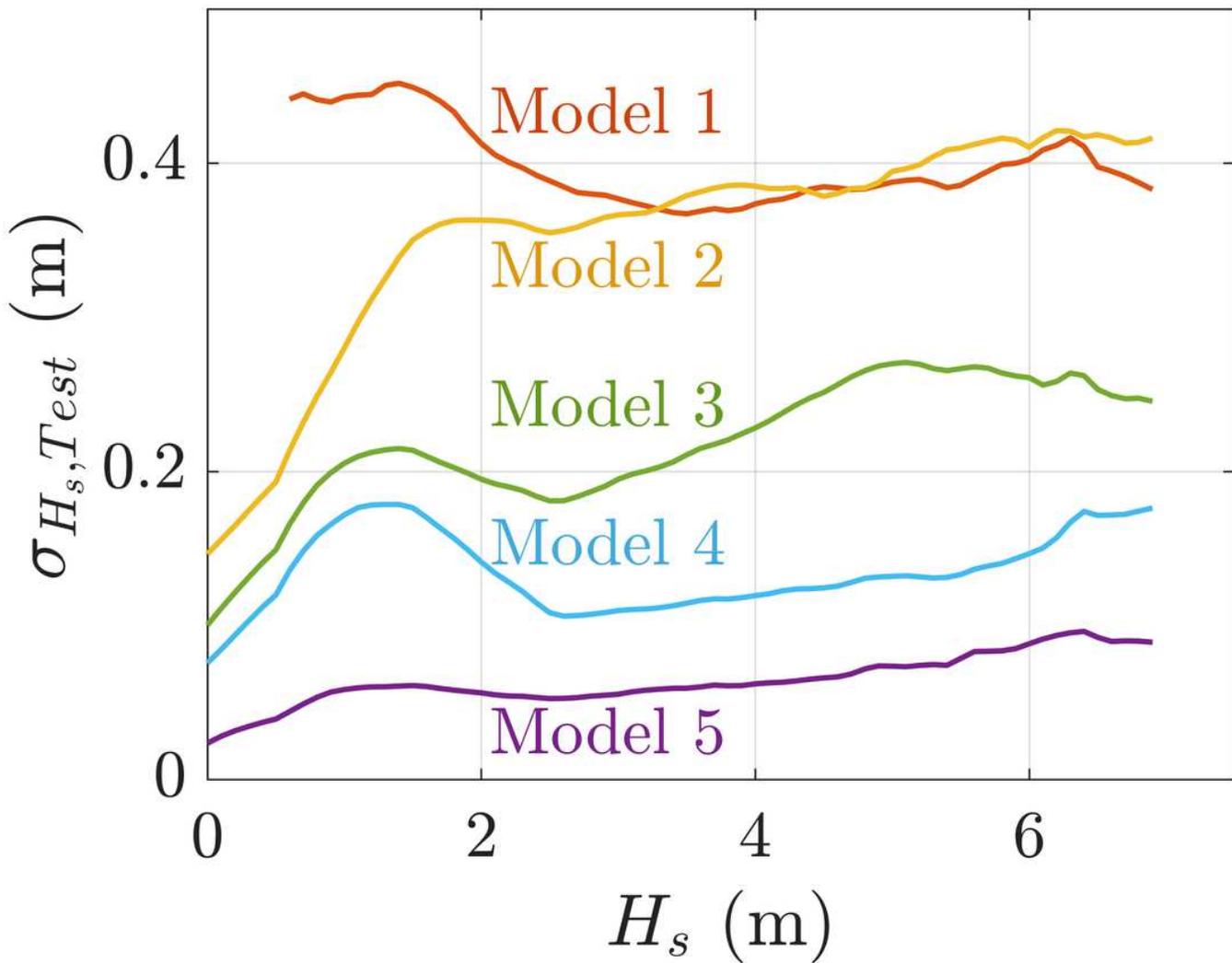
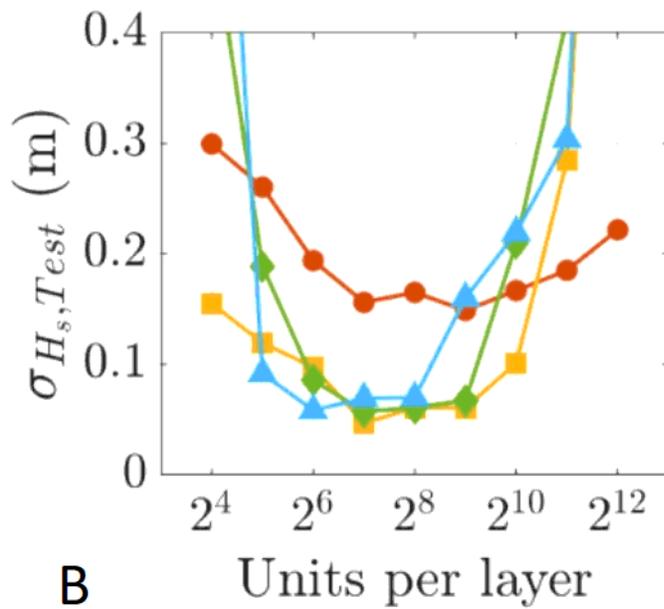
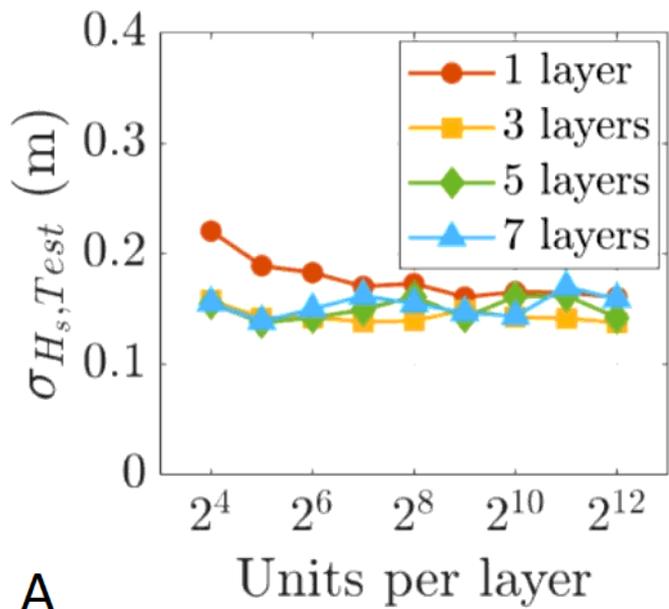


Figure 6

Prediction errors as a function of  $H_s$ .

Figure 7

Time history prediction of a)  $H_s$ , b)  $T_p$ , c)  $\theta(H_s)$ , and d)  $\eta$  for a testing hurricane. Hour zero is selected as the time of maximum  $H_s$ .



**Figure 8**

The impact of the network width on model performance for a) Model 4 and b) Model 5.