Evaluation of Deep Neural Operator Models toward Ocean Forecasting

Ellery Rajagopal[b], Anantha N. S. Babu[a], Tony Ryu[a], Patrick J. Haley, Jr.[a], Chris Mirabito[a], Pierre F. J. Lermusiaux[a, †]

[a] *Department of Mechanical Engineering, Center for Computational Science and Engineering,*
[b] *Department of Electrical Engineering and Computer Science,*
Massachusetts Institute of Technology, Cambridge, MA
[†]Corresponding author: pierrel@mit.edu

Abstract—Data-driven, deep-learning modeling frameworks have been recently developed for forecasting time series data. Such machine learning models may be useful in multiple domains including the atmospheric and oceanic ones, and in general, the larger fluids community. The present work investigates the possible effectiveness of such deep neural operator models for reproducing and predicting classic fluid flows and simulations of realistic ocean dynamics. We first briefly evaluate the capabilities of such deep neural operator models when trained on a simulated two-dimensional fluid flow past a cylinder. We then investigate their application to forecasting ocean surface circulation in the Middle Atlantic Bight and Massachusetts Bay, learning from high-resolution data-assimilation simulations employed for real sea experiments. We confirm that trained deep neural operator models are capable of predicting idealized periodic eddy shedding. For realistic ocean surface flows and our preliminary study, they can predict several of the features and show some skill, providing potential for future research and applications.

Index Terms—machine learning, deep learning, neural operators, transformers, ocean modeling, ocean forecasting

I. INTRODUCTION

In recent years, the fluid dynamics communities including Numerical Weather Prediction (NWP) have been introduced to a collection of data-driven, deep-learning models. These models make use of machine learning architectures such as neural networks, neural operators, and transformers to learn the mapping between one continuous function space to another, and the resulting mapping allows forecasting time series data. In general, neural operators directly learn the mapping between two infinite-dimensional function spaces. DeepONet [1] was the first neural operator to learn such mapping combining sub-networks of a branch net and a trunk net. Various modifications and extensions of DeepONet have been proposed to further improve its accuracy and efficiency [2–4]. Neural operators were also developed based on graph kernel networks with kernel integration done by message passing using graph networks [5]. However, these operators are unstable for very deep neural networks. To overcome this, another neural operator was proposed based on the Fourier transform [6]. More recently, neural operators have been developed using the wavelet transform [7, 8]. Since their introduction, vision transformers have shown superior performance over other architectures in various image recognition and generation tasks [9]. For high-resolution applications, transformers with self-attention mechanisms can capture fine-scale features with higher accuracy and efficiency than conventional architectures such as convolutional neural networks (CNNs) [10].

Two recent examples of models that combine neural nets, neural operators, and transformers are FourCastNet [11] and Latent DeepONet [4]. FourCastNet combines neural operators, specifically Fourier Neural Operators (FNOs) [6], with Vision Transformers (ViT) [9] to represent the operator that evolves a physical quantity forward in time by one unit of time. Latent DeepONet learns the operator that maps a physical quantity at an initial state to the same quantity evolved forward in time after a (predefined) number of timesteps. It does so by combining the output of two neural nets: one for encoding the input at a fixed number of points and the other for encoding the locations for the output.

While many of these models have been trained for and applied to forecasting atmospheric variables such as air temperature and wind velocity [4, 7, 11, 12], they do not restrict inputs only to atmospheric quantities. That is, one may treat these models as a "black box" and train them with appropriate tuning of the network architecture such that they take the state of a dynamical system as input and return the state evolved forward in time as output [2]. This allows applying these models to other fields such as ocean or fluid dynamics by choosing the adequate inputs and outputs during training.

Several studies focus on these models because they offer reduced computational costs during inference [13] and can be trained directly with diverse types of observational data [2]. Many deep neural operator models so far aim to reproduce or forecast the system state without the aid of governing equations [6, 12]. They also avoid the use of adjoint equations for training [6] since neural operators are differentiable and hence back-propagation does not require adjoint methods as those used in neural ODEs or DDEs [14, 15].

In this work, to establish a benchmark, we first apply deep neural operator models to a classic idealized fluid flow, the two-dimensional (2D) flow past a cylinder [16, 17]. We then investigate their application to surface ocean circulation forecasting [18, 19]. Specifically, we consider the Middle Atlantic Bight [20–22] and Massachusetts Bay [23, 24], and utilize training and verification data from high-resolution data-assimilative simulations developed for real sea experiments.

II. DEEP NEURAL OPERATOR MODELS

One is often interested in representing the time evolution of a physical quantity defined over a continuous domain. More precisely, a common goal is to learn the mapping between the quantity at one time to the quantity at another time. Despite

the continuous analytical representations, one discretizes the domain for the purpose of computation. When applied to discretized problems, the performances of neural networks were shown to vary heavily depending on the choice of discretization [5]. As the ideal discretization for a given problem or class of problems is rarely known, standard neural networks are commonly limited by the specifics of the discretization.

As opposed to learning a mapping between two finite-dimensional vector spaces, neural operators aim to learn a mapping between two infinite-dimensional function spaces [25]. A key advantage over standard neural networks is then that neural operators are discretization invariant, thus avoiding the fundamental issue encountered by standard networks. One is no longer restricted by the specific choice of discretization to learn the time evolution of the desired quantities.

A. DeepONet

DeepONet was the first neural operator to learn such a mapping, taking advantage of the universal approximation theorem of operators that suggests a sufficiently large neural network with a single hidden layer can accurately approximate any nonlinear continuous operator up to a small approximation error [1, 26]. As a result, DeepONet can be used to learn the operator which maps a physical quantity at an initial state to the same quantity evolved forward in time after a given number of timesteps. It does so by combining the output of the trunk net, which encodes the input at a fixed number of points, with the output of the branch net, which encodes the locations for the output. The trunk-branch structure is directly inspired by the universal approximation theorem of operators.

Consider an input function u and an operator G, we are interested in the output $(G(u))(y)$ where y is a spatial location in the domain of $G(u)$. DeepONet combines a branch network and a trunk network to obtain the output as follows [1]:

$$(G(u))(y) \approx \sum_{k=1}^{p} b_k(u) \otimes t_k(y) + b_0 \tag{1}$$

where p is the number of branch networks, b_k the output of each branch network, t_k the output of the trunk network, and b_0 the bias. The specific architecture of the sub-networks could be designed according to the desired application [2]. Latent DeepONet [4] is an extension of DeepONet to learn the operator from the latent representations of the input and output spaces. This is implemented utilizing a multi-layer autoencoder which is pre-trained to encode the inputs and decode the outputs of DeepONet.

In fluid dynamics, DeepONet has been used to study linear instabilities in high-speed boundary layers and their inverse problem [27, 28]. It also inferred multiscale bubble growth and dynamics [29]. Latent DeepONet had successes in fluid convection and in simulating shallow water equations [4].

B. Kernel-Based Neural Operator Models

Neural operator mappings were also learned through the use of graph kernel networks [5], which rely on an underlying quadrature-like or Nyström approximation formulation [30]

linking multiple grid points to one set of parameters for the network. A key difference between neural networks and these neural operators is the inclusion of kernel integration in the formulation. With $\vec{v}_0$ serving as the original input, the output $\vec{v}_N$ of a standard neural network is defined as the N-th iteration of the recursion [5],

$$\vec{v}_{i+1} = \sigma(W_i \vec{v}_i + b_i) \tag{2}$$

In this formulation, W is a linear transformation, b the bias, and σ the nonlinear activation function. The output of a graph kernel network, on the other hand, follows a similar but different recursion, switching the vector $\vec{v}_i$ for the function $v_i(x)$ where x is any spatial location in the domain of interest. With $v_0(x)$ serving as the original input, the output $v_N(x)$ is defined as the N-th iteration of the recursion

$$v_{i+1}(x) = \sigma\left(W_i v_i(x) + (K(a;\Phi)v_i)(x)\right) \tag{3}$$

where W is again a linear transformation and σ a nonlinear activation function. $K(a;\Phi)$ is the kernel integral transformation parameterized by Φ and defined as

$$(K(a;\Phi)v_i)(x) = \int_D \kappa_\Phi(x,y,a(x),a(y);\phi)v_i(y)dy \tag{4}$$

where κ_Φ is a neural network function with parameters Φ, a is a function of space, and x,y are spatial locations in the domain of interest.

The now well-known FNO architecture was developed by expanding upon the idea of graph kernel networks and learning parameters directly in Fourier space through the use of what is known as Fourier layers. These layers are discretization-invariant because they can learn from and evaluate functions that are discretized arbitrarily [5]. The FNO architecture imposes that the kernel function κ_Φ not depend on the function a, along with $\kappa_\Phi(x,y) = \kappa_\Phi(x-y)$. This turns the kernel integration operator into a convolution operator, allowing for the use of the FFT for a boost in computational speed [6].

In fluid dynamics applications, FNOs have been used for predicting turbulent flows [31, 32]. Their accuracy in predicting cylindrical wakes for a range of subcritical Reynolds numbers has been studied [33]. FNOs have also been used in modeling multi-fidelity and multi-phase flows [34, 35].

FNOs were extended to Adaptive FNOs (AFNOs) that make use of the self-attention mechanism of the ViT architecture to address the shortcomings of using images as input for FNOs [10]. This improves the scalability and robustness of the resulting predictions by imposing a block diagonal structure on the weight tensor, sharing weights adaptively across tokens, and sparsifying tokens using soft-thresholding and shrinkage operations. The AFNO model is the backbone of FourCastNet, which was originally designed to predict global weather patterns using atmospheric state variables such as surface wind speed, precipitation, and atmospheric water vapor [11]. For this work, we investigate the preliminary use of FourCastNet and Latent DeepONet within an oceanographic context.

C. Model Inputs and Outputs, and Training

While both FourCastNet (FCN) and DeepONet (L-DoN) are used for operator learning, the structure of the operators

for both models is different. For a collection of images $\{x_1, \ldots, x_n\}$, FCN learns the operator which maps $x_i \mapsto x_{i+1} \; \forall i \in \{1, \ldots, n-1\}$. That is, we have

$$x_{i+1} = FCN(x_i) \; \forall i \in \{1, \ldots, n-1\}. \qquad (5)$$

Each training point for FCN is thus the pair

$$(x_i, x_{i+1}) \; \forall \; i \in \{1, \ldots, n-1\}. \qquad (6)$$

For the same set of images, L-DoN learns the mapping from $x_1 \mapsto \{x_1, \ldots, x_N\}$. That is, it maps the first image to all N images, allowing for the prediction of N images in "one shot". This means that

$$\{x_1, \ldots, x_N\} = DON(x_1). \qquad (7)$$

To train FCN, we first partition our training data into predefined batches of equal size, with each batch containing contiguous subsets of the training time series. We then loop through our collection of batches and perform a forward pass of the model with each batch. We evaluate the loss function on the output of each forward pass. Using PyTorch's automatic differentiation, we use stochastic gradient descents to adjust the model weights. This process constitutes one epoch of training, and we train for a predefined number of epochs. With L-DoN, input values are mapped to many data values. Inputs are selected such that the union of their outputs covers the training set. The training is otherwise similar to that of FCN.

III. FLOW PAST A CYLINDER

We first briefly evaluate the properties and accuracy of the two deep neural operator architectures when trained on the 2D velocity of a fluid flowing past a cylinder (FpC) or idealized island. The key parameter is the non-dimensional Reynolds number, $\mathrm{Re} = \mathrm{V}_\infty \mathrm{L}/\nu$ [17], where V_∞ is the fluid velocity at infinity from the obstacle, L the projected width of the obstacle, and ν the dynamic viscosity of the fluid. For Re within approximately 5 and 40, the flow downstream of the obstacle is symmetric. For Re greater than 40, the flows exhibit periodic, asymmetric vortex shedding known as von Kármán vortex streets. For Re increasing beyond 200, more complex aperiodic patterns appear, up to fully turbulent flows.

To create the data sets for these idealized flow cases, we employ our MIT-MSEAS 2.29 finite-volume (FV) framework [36, 37]. It contains a finite volume solver for 2D nonhydrostatic fluid and ocean dynamics. It has been very useful for diverse ocean dynamics process studies and for the incubation of advanced schemes and methodologies.

A. Test Case Description

Many evaluations were completed, but we only show results for a single set-up. The domain is a rectangular channel with a height of 3 m and a width of 20 m. The inlet velocity is set to 2 m/s. The diameter of the cylinder is set at 1 m, meaning with our kinematic viscosity of $0.01 \; \mathrm{m}^2/\mathrm{s}$, the Re is 200.

The MIT-MSEAS 2.29 numerical simulation uses 50×100 second-order finite volumes and a time-step of 0.005 s. The diffusion operator is discretized using central boundary fluxes, a total variation diminishing (TVD) scheme with a monotonized central (MC) symmetric flux limiter is used for

advection, and a rotational incremental pressure correction scheme is used for time integration. We simulate the flow for a total of 500 s. With the above parameters, we observe fully developed periodic, asymmetric vortex shedding after 50 s. An eddy sheds approximately every 2 s, so we find a shedding period of about 4 s. For the machine learning evaluation, the horizontal and vertical velocities, u and v, respectively, are sub-sampled every 0.01 s and saved for the whole 50×100 numerical domain, but only starting at time $t = 50$ s after the flow has developed. That is, these 450 seconds provide 45000 snapshots of dimension $50 \times 100 \times 2$ (each snapshot is a 50×100 array for each of u and v in m/s).

B. Learning Results

We find that both FCN and L-DoN are capable of predicting the present idealized FpC flows with periodic, asymmetric vortex shedding. They can do this for long periods of time compared to the shedding frequency, but the phase and details of the structures can be lost over time. Varying the training procedures, network parameters, and hyperparameters can significantly alter the performance.

For good FCN and L-DoN performance, we trained on the first 25,000 snapshots, validated on the next 10,000, and predicted on the remaining 10,000. For FCN, we selected a global batch size of 2, size of image patches of 2×2, number of AFNO layers (depth) of 3, and embedded channel size of 96. The loss function employed was the L2-norm. Training was executed for 150 to 200 epochs. For L-DoN, we utilize similar hyperparameters as in [4] but with latent dimension of 16, batch size of 2, and size of the final hidden layer (p) of 4. The loss function employed was the L2-norm. Training was executed for 1000 epochs.

IV. REALISTIC OCEAN SURFACE FLOWS

We now investigate the use of deep neural operator models to represent and forecast realistic surface ocean circulations in two ocean regions and dynamical regimes. We first examine shelf-to-deep dynamics in the Middle Atlantic Bight region with the Gulf Stream, shelfbreak front, internal and external tides, and large storms. We then consider shallow coastal dynamics in Massachusetts Bay with coastal currents and jets, strong tidal flows, and intermittent atmospheric forcing.

For these two applications, the learning data consist of high-resolution data-assimilative simulations computed for real sea experiments [18, 20, 24] by our MIT Multidisciplinary Simulation, Estimation, and Assimilation Systems primitive equation (MSEAS-PE) modeling system. The MSEAS-PE has been used for fundamental research and for realistic simulations in varied regions of the World Ocean [38–45]. It is able to simulate sub-mesoscale processes over regional domains with complex geometries and varied interactions using an implicit two-way nesting/tiling scheme [18, 19]. Other capabilities include ensemble forecasting and data assimilation [22, 46–49], tidal prediction and inversion [50], coastal objective analysis [51], subgrid-scale models [23, 52], reduced-order
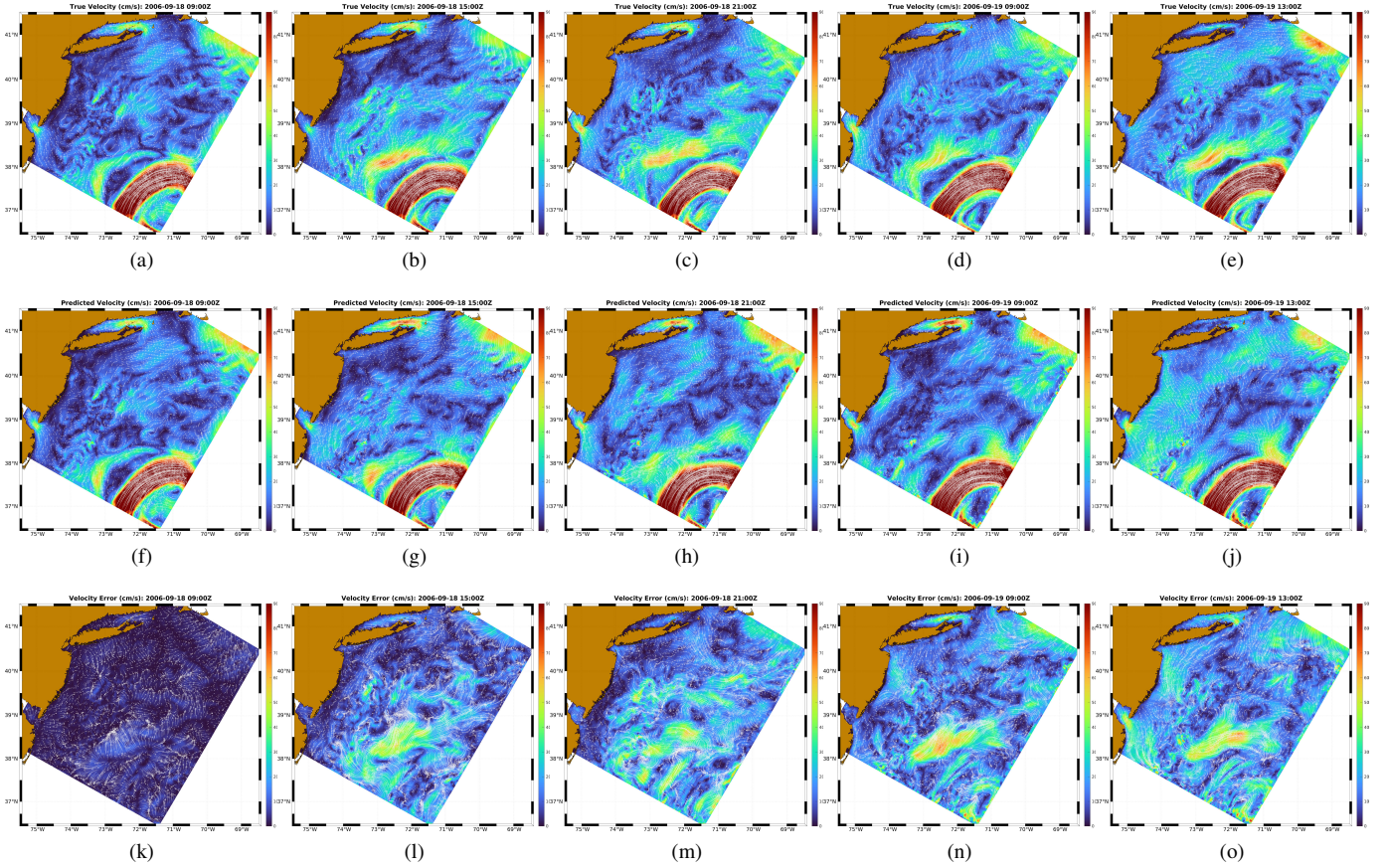
Fig. 1: *Applying FCN in the MAB*. Surface velocity field amplitudes overlaid with curved velocity vectors. Ground truth from the MSEAS-PE forecast (top row), FCN predictions for training-run 246 (middle row), and error fields (bottom row) for five different forecast times: 2006 Sep 18 09Z and 6, 12, 24, and 28 hours later.

modeling [53–57], and path planning and adaptive sampling [40, 49, 58–60].

A. Middle Atlantic Bight (MAB)

We first evaluate the use of the FCN architecture based on a real ocean experiment in the MAB. The MAB is a broad shelf with strong tides, near-inertial wave activity, freshwater inputs, and a shelfbreak with a front. It is influenced by the Gulf Stream, the warm-core rings, and a recirculating deep circulation, and is also affected by tropical storms [61, 62].

1) Test Case Description: The MAB simulation data are an outgrowth of the real-time MSEAS-PE forecasts issued during the Shallow Water 2006 (SW06) and Autonomous Wide Aperture Cluster for Surveillance (AWACS) 2006 experiments [18, 20, 21, 61, 63, 64]. The MSEAS-PE was configured in a two-way nested mode with a 3 km resolution regional domain (150×175 horizontal grid) and a 1 km resolution shelfbreak domain (173×156 horizontal grid). Both domains employ 100 terrain-following vertical levels optimized for the background T/S structure of the slope/shelf region. Surface forcing was a combination of the 6 km Weather Research & Forecasting (WRF) [65] and the $1°$ Navy Operational Global Atmospheric Prediction System (NOGAPS) [66] atmospheric

fluxes. The nested simulations were forced with tides from the high resolution TPXO8-Atlas [67, 68] adjusted for our higher resolution bathymetry and coastlines [50]. The final data-assimilative reanalysis spanned Aug. 14 to Sep. 24, 2006, and was saved at hourly resolution providing a total of 1007 snapshots. Additional details can be found in [18, 69, 70].

The FCN results presented in this paper are for the surface circulation in the regional (3 km) domain. The first 48 snapshots are not used (dynamical adjustments) and 959 snapshots of size 2x150x174 are available (each snapshot is an array for u and v in cm/s). The snapshots contain both land and ocean regions with 4,433 land cells and 21,667 ocean cells.

ML data usage, NN architecture, loss function, parameters, and sensitivity studies. For what we show, we employ the first 600 snapshots of surface velocity u and v for training, the next 200 for validation, and the remaining 159 for verification/prediction. We completed sensitivity studies for these periods, as well as for several of the FCN architectural choices and key hyperparameters such as the loss function, number of epochs, global batch size, size of image patches (p×p), number of AFNO layers (depth), and embedded channel size. For example, we varied the global batch size from 1 to 12, the image patch size from 2x2 to 6x6, the depth from 10

4

to 25, and the embedded channel size from 384 to 768. The model was trained for about 150 to 200 epochs. For the field results shown next, the MAB-trained FCN model corresponds to MAB-training-run 246 that has a tuned global batch size of 1, size of image patches of 3x3, number of AFNO layers (depth) of 10, and embedded channel size of 640. For this training-run 246, the other hyperparameters such as learning rate, sparsity threshold, and MLP ratio were those of the original FCN model [11]. The loss function employed was the L2-norm. To capture the dynamics, we found that tuning the patch size and increasing the embedded channel size led to the best results during training, validation, and prediction.

2) Learning Results: A comparison of the simulated ground truth, an MSEAS-PE forecast of the surface velocity field for 2006 Sep 18-19 in the MAB, to an FCN prediction trained on earlier MSEAS-PE fields (training-run 246) is shown in Figure 1. The Gulf Stream is present in the southern corner of the domain, with a small crest that begins to move out of the domain. This is a period of low winds so the shelf is dominated by tides (primarily M2) especially on Nantucket Shoals (northern boundary) and internal tides. The FCN forecast captures the changes in the Gulf Stream well. It also captures predominantly M2 tidal signal on the shelf, although it slightly overestimates the tidal strength.

In Figure 2, we show the root-mean-square-errors (RMSEs) of ten FCN forecasts of MAB surface velocity fields for 0 to 29h forecast lead times. The different FCN forecasts were selected among the better ones (one of which is MAB-training-run 246 shown on Fig. 1) and correspond to ten MAB training-runs with different hyperparameter values. For these FCN forecasts, averaged errors grow linearly quickly, and then taper off or grow more slowly, with a clear effect of M2 tides. Results confirm that the FCN forecasts have accuracy, with local errors reaching about 5 to 50% of the variability after 29 h. The FCN forecast uncertainty is estimated by the spread of the RMSE curves shown.

B. Massachusetts Bay (MB)

We now evaluate FCN based on a real ocean application in MB. MB is a sub-region of the Gulf of Maine (GoM), off the northeast US coast. The circulation in MB is commonly from north to south and is remotely driven from the GoM coastal current and mean wind stress [23, 24, 71, 72]. However, the MB circulation varies both seasonally and in response to wind events. MB is also strongly impacted by tides, with intense internal waves radiating off Stellwagen Bank [23, 45, 73].

1) Test Case Description: The present MB simulation was completed as a part of our "Bayesian Intelligent Ocean Modeling and Acidification Prediction Systems" (BIOMAPS) research [24, 44, 74]. The MSEAS-PE modeling system was configured in a stand-alone mode with 333 m resolution (266×451 horizontal grid) and 100 vertical levels optimized for MB. The sub-tidal initial and boundary conditions were downscaled from the global $1/12°$ Hybrid Coordinate Ocean Model (HYCOM) analyses [75], using our optimization for our higher resolution coastlines and bathymetry [19]. Local
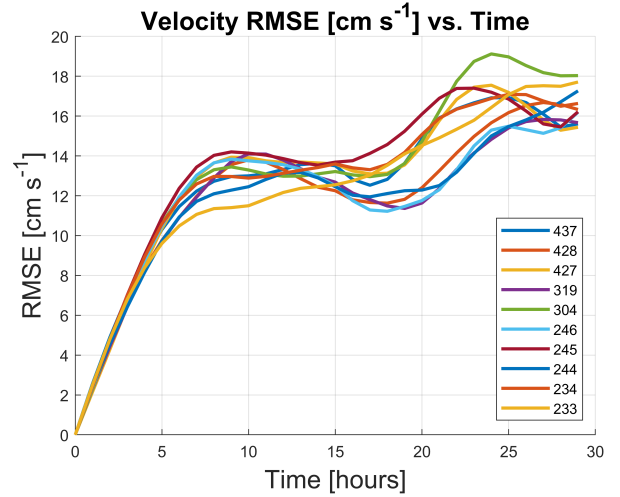

Fig. 2: *Applying FCN in the MAB*. Errors (RMSE) of FCN forecasts of surface velocity for 0 to 29h forecast lead times (2006 Sep 18 08Z to Sep 19 13Z). The FNC forecasts correspond to ten MAB training-runs with different hyperparameter values.

corrections were made using feature models and synoptic CTDs. Tidal forcing was computed from the OSU high-resolution TPXO8-Atlas, by reprocessing for our higher resolution bathymetry/coastline and quadratic bottom drag (a non-linear extension of [50]). The atmospheric forcing consisted of hourly analyses/forecasts of wind stresses, net heat flux, and surface freshwater flux from the 3 km North American Mesoscale Forecast System (NAM) [76]. The MB reanalysis spanned the period August 11 to September 13, 2019, and was saved at hourly resolution, providing 781 snapshots. Additional details can be found in [24].

The FCN results presented in this paper are for the surface currents. The first 48 snapshots are not used (dynamical adjustments) and 733 snapshots of size 2x450x264 are available (each snapshot is an array for u and v in cm/s). The snapshots contain both land and ocean regions with 39,697 land cells and 79,103 ocean cells.

ML data usage, NN architecture, loss function, parameters, and sensitivity studies. For the results we show, we utilize the first 400 snapshots for training, the next 100 for validation, and the remaining 233 for verification/prediction. We performed a sensitivity study of the FCN architectural choices and hyper-parameters similar to that for MAB (Sect. IV-A1). For the field results shown next, the MB-trained FCN model corresponds to MB-training-run 264 that has a tuned global batch size of 6, size of image patches of 3x3, number of AFNO layers (depth) of 10, and embedded channel size of 384.

2) Learning Results: A comparison of the simulated ground truth, MSEAS-PE forecasts of the surface velocity field for 2019 Sep 3-4 in MB, to FCN predictions trained on earlier MSEAS-PE fields (MB-training-run 264) is shown in Figure 3. The circulation is complex with dynamic features and rapid changes. Tides have a strong influence, especially over Stellwagen Bank and around the tip and eastern side of Cape Cod.
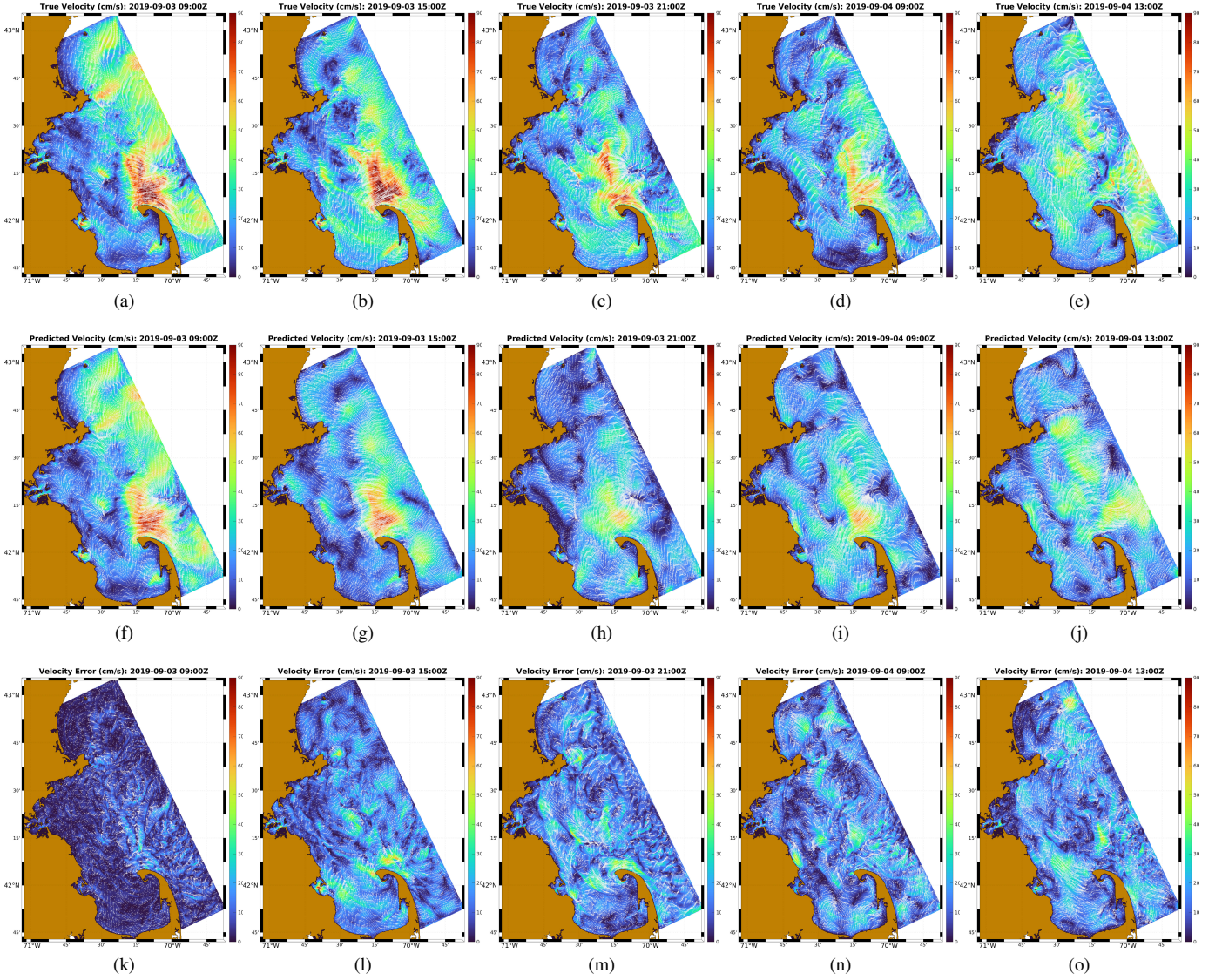
5

Fig. 3: *Applying FCN in MB*. Surface velocity field amplitudes overlaid with curved velocity vectors. Ground truth from the MSEAS-PE forecast (top row), FCN predictions for training-run 264 (middle row), and error fields (bottom row) for five different forecast times: 2019 Sep 3 09Z and 6, 12, 24, and 28 hours later.

The winds are moderate and modulate the tidal effects. We note that winds were not contained in the FCN inputs so FCN can only learn and predict their effects from indirect surface current responses. FCN does a good job capturing the tidal signal and the general sense of the local flows, but somewhat underestimates the intensification over Stellwagen Bank.

In Figure 4, we show RMSEs of ten FCN forecasts of MB surface velocity fields for 0 to 29h forecast lead times. The different FCN forecasts were selected among the better ones (one of which is MB-training-run 264 shown on Fig. 3) and correspond to ten MAB training-runs with different hyperparameter values. For these FCN forecasts, averaged errors grow linearly quickly, and then taper off, with clear effects of tides and moderate winds. Results confirm that the FCN forecasts have accuracy at many locations in MB, with most local errors

remaining within 1 to 50% of the variability after 29 h. The FCN forecast uncertainty is estimated by the spread of the RMSE curves shown.

V. CONCLUSIONS

Deep neural operator modeling frameworks have been recently developed for predicting the evolution of dynamical systems, learning from past simulation data or experimental data. The present work investigated the use of such models for reproducing and predicting the 2D flow past a cylinder and realistic ocean dynamics, learning from past simulation data. The flow past a cylinder example confirmed that FCN and L-DoN can predict idealized flows with periodic, asymmetric vortex shedding. We then illustrated the use of deep neural operator models to forecast ocean surface circulation in the Middle
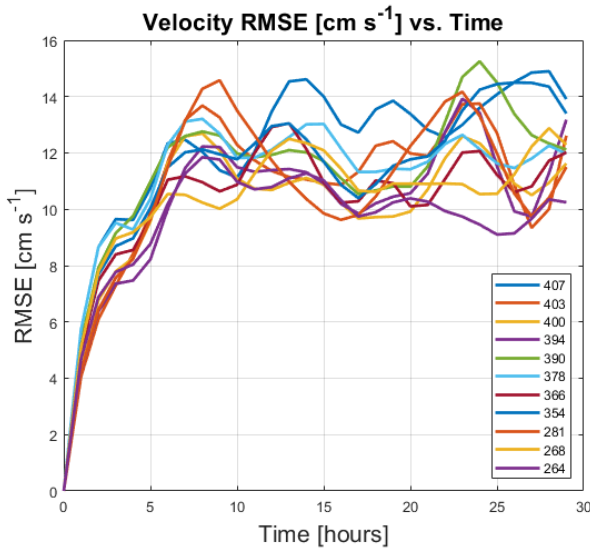
Fig. 4: *Applying FCN in MB.* Errors (RMSE) of FCN forecasts of surface velocity for 0 to 29h forecast lead times (2019 Sep 3 08Z to Sep 4 13Z). The FNC forecasts correspond to ten MAB training-runs with different hyperparameter values.

Atlantic Bight and in Massachusetts Bay, using training and verification data from our multi-resolution, data-assimilative MSEAS-PE simulations that we employed during real-time sea experiments in 2006 and 2019, respectively. We completed some sensitivity studies on the ML model data, architecture, and parameters. We compared sets of ML forecasts to the ground-truth MSEAS-PE simulations, and we found that the trained deep neural operator models predicted several of the features and showed some skill.

The results show promise for future ocean research and applications. This includes targeted repetitive forecasting, data assimilation, path planning, adaptive sampling, monitoring, sustainability, protection, risk management, and climate.

REFERENCES

[1] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, "Learning nonlinear operators via deeponet based on the universal approximation theorem of operators," *Nature machine intelligence*, vol. 3, no. 3, pp. 218–229, 2021.

[2] L. Lu, X. Meng, S. Cai, Z. Mao, S. Goswami, Z. Zhang, and G. E. Karniadakis, "A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data," *Computer Methods in Applied Mechanics and Engineering*, vol. 393, p. 114778, 2022.

[3] S. Goswami, M. Yin, Y. Yu, and G. E. Karniadakis, "A physics-informed variational deeponet for predicting crack path in quasi-brittle materials," *Computer Methods in Applied Mechanics and Engineering*, vol. 391, p. 114587, 2022.

[4] K. Kontolati, S. Goswami, G. E. Karniadakis, and M. D. Shields, "Learning in latent spaces improves the predictive accuracy of deep neural operators," *arXiv preprint arXiv:2304.07599*, 2023.

[5] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Neural operator: Graph kernel network for partial differential equations," *arXiv preprint arXiv:2003.03485*, 2020.

[6] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Fourier neural operator for parametric partial differential equations," *arXiv preprint arXiv:2010.08895*, 2020.

[7] T. Tripura and S. Chakraborty, "Wavelet neural operator: a neural operator for parametric partial differential equations," *arXiv preprint arXiv:2205.02191*, 2022.

[8] G. Gupta, X. Xiao, and P. Bogdan, "Multiwavelet-based operator learning for differential equations," *Advances in neural information processing sys.*, vol. 34, pp. 24048–24062, 2021.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[10] J. Guibas, M. Mardani, Z. Li, A. Tao, A. Anandkumar, and B. Catanzaro, "Adaptive fourier neural operators: Efficient token mixers for transformers," *arXiv preprint arXiv:2111.13587*, 2021.

[11] J. Pathak, S. Subramanian, P. Harrington, S. Raja, A. Chattopadhyay, M. Mardani, T. Kurth, D. Hall, Z. Li, K. Azizzadenesheli, P. Hassanzadeh, K. Kashinath, and A. Anandkumar, "Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators," 2022.

[12] B. Bonev, T. Kurth, C. Hundt, J. Pathak, M. Baust, K. Kashinath, and A. Anandkumar, "Spherical fourier neural operators: Learning stable dynamics on the sphere," *arXiv preprint arXiv:2306.03838*, 2023.

[13] K. Michałowska, S. Goswami, G. E. Karniadakis, and S. Riemer-Sørensen, "Neural operator learning for long-time integration in dynamical systems with recurrent neural networks," *arXiv preprint arXiv:2303.02243*, 2023.

[14] A. Gupta and P. F. J. Lermusiaux, "Generalized neural closure models with interpretability," *Scientific Reports*, 2023. In press.

[15] A. Gupta and P. F. J. Lermusiaux, "Neural closure models for dynamical systems," *Proceedings of The Royal Society A*, vol. 477, pp. 1–29, Aug. 2021.

[16] M. Zdravkovich, *Flow Around Circular Cylinders Volume 1: Fundamentals*. Cambridge University Press, 1997.

[17] P. Kundu and I. Cohen, *Fluid Mechanics*. Elsevier, 2004.

[18] P. J. Haley, Jr. and P. F. J. Lermusiaux, "Multiscale two-way embedding schemes for free-surface primitive equations in the "Multidisciplinary Simulation, Estimation and Assimilation System"," *Ocean Dynamics*, vol. 60, pp. 1497–1537, Dec. 2010.

[19] P. J. Haley, Jr., A. Agarwal, and P. F. J. Lermusiaux, "Optimizing velocities and transports for complex coastal regions and archipelagos," *Ocean Modeling*, vol. 89, pp. 1–28, 2015.

[20] P. F. J. Lermusiaux, P. J. Haley, Jr., W. G. Leslie, O. Logoutov, and A. R. Robinson, "Autonomous Wide Aperture Cluster for Surveillance (AWACS): Adaptive sampling and search using predictive models with coupled data assimilation and feedback," 2006.

[21] Y.-T. Lin, A. E. Newhall, T. F. Duda, P. F. J. Lermusiaux, and P. J. Haley, "Merging multiple-partial-depth data time series using objective empirical orthogonal function fitting," *IEEE Journal of Oceanic Engineering*, vol. 35, pp. 710–721, Oct. 2010.

[22] P. F. J. Lermusiaux, C. Mirabito, P. J. Haley, Jr., W. H. Ali, A. Gupta, S. Jana, E. Dorfman, A. Laferriere, A. Kofford, G. Shepard, M. Goldsmith, K. Heaney, E. Coelho, J. Boyle, J. Murray, L. Freitag, and A. Morozov, "Real-time probabilistic coupled ocean physics-acoustics forecasting and data assimilation for underwater GPS," in *OCEANS 2020 IEEE/MTS*, pp. 1–9, IEEE, Oct. 2020.

[23] P. F. J. Lermusiaux, "Evolving the subspace of the three-dimensional multiscale ocean variability: Massachusetts Bay," *Journal of Marine Systems*, vol. 29, no. 1, pp. 385–422, 2001.

[24] P. J. Haley, Jr., A. Gupta, C. Mirabito, and P. F. J. Lermusiaux, "Towards Bayesian ocean physical-biogeochemical-acidification prediction and learning systems for Massachusetts Bay," in *OCEANS 2020 IEEE/MTS*, pp. 1–9, IEEE, Oct. 2020.

[25] N. B. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. M. Stuart, and A. Anandkumar, "Neural operator: Learning maps between function spaces with applications to pdes.," *J. Mach. Learn. Res.*, vol. 24, no. 89, pp. 1–97, 2023.

[26] T. Chen and H. Chen, "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems," *IEEE transactions on neural networks*, vol. 6, no. 4, pp. 911–917, 1995.

[27] P. C. Di Leoni, L. Lu, C. Meneveau, G. E. Karniadakis, and T. A. Zaki, "Neural operator prediction of linear instability waves in high-speed boundary layers," *Journal of Computational Physics*, vol. 474, p. 111793, 2023.

[28] Y. Hao, P. C. Di Leoni, O. Marxen, C. Meneveau, G. E. Karniadakis, and T. A. Zaki, "Instability-wave prediction in hypersonic boundary layers with physics-informed neural operators," *Journal of Computational Science*, p. 102120, 2023.

[29] C. Lin, M. Maxey, Z. Li, and G. E. Karniadakis, "A seamless multiscale operator neural network for inferring bubble dynamics," *Journal of Fluid Mechanics*, vol. 929, p. A18, 2021.

[30] E. J. Nyström, "Über die praktische auflösung von integralgleichungen mit anwendungen auf randwertaufgaben," *Acta Mathematica*, vol. 54, pp. 185–204, 1930.

[31] W. Peng, Z. Yuan, and J. Wang, "Attention-enhanced neural network models for turbulence simulation," *Physics of Fluids*, vol. 34, no. 2, 2022.

[32] Z. Li, W. Peng, Z. Yuan, and J. Wang, "Long-term predictions of turbulence by implicit u-net enhanced fourier neural operator," *Physics of Fluids*, vol. 35, no. 7, 2023.

[33] P. I. Renn, C. Wang, S. Lale, Z. Li, A. Anandkumar, and M. Gharib, "Forecasting subcritical cylinder wakes with fourier neural operators," *arXiv preprint arXiv:2301.08290*, 2023.

[34] Y. Lyu, X. Zhao, Z. Gong, X. Kang, and W. Yao, "Multi-fidelity prediction of fluid flow based on transfer learning using fourier neural operator," *Physics of Fluids*, vol. 35, no. 7, 2023.

[35] G. Wen, Z. Li, K. Azizzadenesheli, A. Anandkumar, and S. M. Benson, "U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow," *Advances in Water Resources*, vol. 163, p. 104180, 2022.

[36] P. F. J. Lermusiaux, "Numerical fluid mechanics." MIT OpenCourseWare, May 2015.

[37] M. P. Ueckermann and P. F. J. Lermusiaux, "2.29 Finite Volume MATLAB Framework Documentation," MSEAS Report 14, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 2012.

[38] W. G. Leslie, A. R. Robinson, P. J. Haley, Jr, O. Logutov, P. A. Moreno, P. F. J. Lermusiaux, and E. Coelho, "Verification and training of real-time forecasting of multi-scale ocean dynamics for maritime rapid environmental assessment," *Journal of Marine Systems*, vol. 69, no. 1, pp. 3–16, 2008.

[39] P. J. Haley, Jr., P. F. J. Lermusiaux, A. R. Robinson, W. G. Leslie, O. Logoutov, G. Cossarini, X. S. Liang, P. Moreno, S. R. Ramp, J. D. Doyle, J. Bellingham, F. Chavez, and S. Johnston, "Forecasting and reanalysis in the Monterey Bay/California Current region for the Autonomous Ocean Sampling Network-II experiment," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 56, pp. 127–148, Feb. 2009.

[40] P. F. J. Lermusiaux, D. N. Subramani, J. Lin, C. S. Kulkarni, A. Gupta, A. Dutt, T. Lolla, P. J. Haley, Jr., W. H. Ali, C. Mirabito, and S. Jana, "A future for intelligent autonomous ocean observing systems," *Journal of Marine Research*, vol. 75, pp. 765–813, Nov. 2017. The Sea. Volume 17, The Science of Ocean Prediction, Part 2.

[41] D. N. Subramani, P. J. Haley, Jr., and P. F. J. Lermusiaux, "Energy-optimal path planning in the coastal ocean," *Journal of Geophysical Research: Oceans*, vol. 122, pp. 3981–4003, 2017.

[42] C. S. Kulkarni, P. J. Haley, Jr., P. F. J. Lermusiaux, A. Dutt, A. Gupta, C. Mirabito, D. N. Subramani, S. Jana, W. H. Ali, T. Peacock, C. M. Royo, A. Rzeznik, and R. Supekar, "Real-time sediment plume modeling in the Southern California Bight," in *OCEANS Conference 2018*, (Charleston, SC), IEEE, Oct. 2018.

[43] A. Gupta, P. J. Haley, D. N. Subramani, and P. F. J. Lermusiaux, "Fish modeling and Bayesian learning for the Lakshadweep Islands," in *OCEANS 2019 MTS/IEEE SEATTLE*, (Seattle), pp. 1–10, IEEE, Oct. 2019.

[44] P. F. J. Lermusiaux, M. Doshi, C. S. Kulkarni, A. Gupta, P. J. Haley, Jr., C. Mirabito, F. Trotta, S. J. Levang, G. R. Flierl, J. Marshall, T. Peacock, and C. Noble, "Plastic pollution in the coastal oceans: Characterization and modeling," in *OCEANS 2019 MTS/IEEE SEATTLE*, (Seattle), pp. 1–10, IEEE, Oct. 2019.

[45] W. H. Ali, A. Charous, C. Mirabito, P. J. Haley, Jr., and P. F. J. Lermusiaux, "MSEAS-ParEq for ocean-acoustic modeling around the globe," in *OCEANS 2023 IEEE/MTS Gulf Coast*, (Biloxi, MS), IEEE, Sept. 2023. In press.

[46] P. F. J. Lermusiaux, D. G. M. Anderson, and C. J. Lozano, "On the mapping of multivariate geophysical fields: Error and variability subspace estimates," *Quarterly Journal of the Royal Meteorological Society*, vol. 126, no. 565, pp. 1387–1429, 2000.

[47] P. F. J. Lermusiaux, "On the mapping of multivariate geophysical fields: Sensitivities to size, scales, and dynamics," *Journal of Atmospheric and Oceanic Technology*, vol. 19, no. 10, pp. 1602–1637, 2002.

[48] P. F. J. Lermusiaux, "Estimation and study of mesoscale variability in the Strait of Sicily," *Dynamics of Atmospheres and Oceans*, vol. 29, no. 2, pp. 255–303, 1999.

[49] P. F. J. Lermusiaux, "Adaptive modeling, adaptive data assimilation and adaptive sampling," *Physica D: Nonlinear Phenomena*, vol. 230, no. 1, pp. 172–196, 2007.

[50] O. G. Logutov and P. F. J. Lermusiaux, "Inverse barotropic tidal estimation for regional ocean applications," *Ocean Modelling*, vol. 25, no. 1–2, pp. 17–34, 2008.

[51] A. Agarwal and P. F. J. Lermusiaux, "Statistical field estimation for complex coastal regions and archipelagos," *Ocean Modelling*, vol. 40, no. 2, pp. 164–189, 2011.

[52] P. F. J. Lermusiaux, "Uncertainty estimation and prediction for interdisciplinary ocean dynamics," *Journal of Computational Physics*, vol. 217, no. 1, pp. 176–199, 2006.

[53] F. Feppon and P. F. J. Lermusiaux, "Dynamically orthogonal numerical schemes for efficient stochastic advection and Lagrangian transport," *SIAM Review*, vol. 60, no. 3, pp. 595–625, 2018.

[54] A. Charous and P. F. J. Lermusiaux, "Dynamically orthogonal differential equations for stochastic and deterministic reduced-order modeling of ocean acoustic wave propagation,"

in *OCEANS 2021 IEEE/MTS*, pp. 1–7, IEEE, Sept. 2021.

[55] J. P. Heuss, P. J. Haley, Jr., C. Mirabito, E. Coelho, M. C. Schönau, K. Heaney, and P. F. J. Lermusiaux, "Reduced order modeling for stochastic prediction onboard autonomous platforms at sea," in *OCEANS 2020 IEEE/MTS*, pp. 1–10, IEEE, Oct. 2020.

[56] T. Ryu, J. P. Heuss, P. J. Haley, Jr., C. Mirabito, E. Coelho, P. Hursky, M. C. Schönau, K. Heaney, and P. F. J. Lermusiaux, "Adaptive stochastic reduced order modeling for autonomous ocean platforms," in *OCEANS 2021 IEEE/MTS*, pp. 1–9, IEEE, Sept. 2021.

[57] T. Ryu, W. H. Ali, P. J. Haley, Jr., C. Mirabito, A. Charous, and P. F. J. Lermusiaux, "Incremental low-rank dynamic mode decomposition model for efficient dynamic forecast dissemination and onboard forecasting," in *OCEANS 2022 IEEE/MTS*, (Hampton Roads, VA), pp. 1–8, IEEE, Oct. 2022.

[58] P. F. J. Lermusiaux, T. Lolla, P. J. Haley, Jr., K. Yigit, M. P. Ueckermann, T. Sondergaard, and W. G. Leslie, "Science of autonomy: Time-optimal path planning and adaptive sampling for swarms of ocean vehicles," in *Springer Handbook of Ocean Engineering: Autonomous Ocean Vehicles, Subsystems and Control* (T. Curtin, ed.), ch. 21, pp. 481–498, Springer, 2016.

[59] P. F. J. Lermusiaux, P. J. Haley, Jr, and N. K. Yilmaz, "Environmental prediction, path planning and adaptive sampling: sensing and modeling for efficient ocean monitoring, management and pollution control," *Sea Technology*, vol. 48, no. 9, pp. 35–38, 2007.

[60] M. M. Doshi, M. S. Bhabra, and P. F. J. Lermusiaux, "Energy-time optimal path planning in dynamic flows: Theory and schemes," *Computer Methods in Applied Mechanics and Engineering*, vol. 405, p. 115865, Feb. 2023.

[61] D. Tang, J. N. Moum, J. F. Lynch, P. Abbot, R. Chapman, P. H. Dahl, T. F. Duda, G. Gawarkiewicz, S. Glenn, J. A. Goff, H. Graber, J. Kemp, A. Maffei, J. D. Nash, and A. Newhall, "Shallow water '06: A joint acoustic propagation/nonlinear internal wave physics exp.," *Oceanography*, vol. 20, Dec 2007.

[62] A. Gangopadhyay, A. R. Robinson, and H. G. Arango, "Circulation and dynamics of the western north atlantic. part i: Multiscale feature models," *Journal of Atmospheric and Oceanic Technology*, vol. 14, no. 6, pp. 1314–1332, 1997.

[63] N. R. Chapman and J. F. Lynch, "Editorial: Special issue on the 2006 shallow water experiment," *IEEE Journal of Oceanic Engineering*, vol. 35, no. 1, pp. 1–2, 2010.

[64] M. E. G. D. Colin, T. F. Duda, L. A. te Raa, T. van Zon, P. J. Haley, Jr., P. F. J. Lermusiaux, W. G. Leslie, C. Mirabito, F. P. A. Lam, A. E. Newhall, Y.-T. Lin, and J. F. Lynch, "Time-evolving acoustic propagation modeling in a complex ocean environment," in *OCEANS - Bergen, 2013 MTS/IEEE*, pp. 1–9, 2013.

[65] Rutgers Center for Ocean Observing Leadership, "RUWRF Mesoscale Meteorological Model," 2020.

[66] National Oceanic and Atmospheric Administration, "Navy Operational Global Atmospheric Prediction System (NOGAPS)," 2020.

[67] G. D. Egbert and S. Y. Erofeeva, "Efficient inverse modeling of barotropic ocean tides.," *Journal of Atmospheric and Oceanic Technology*, vol. 19, no. 2, pp. 183–204, 2002.

[68] G. D. Egbert and S. Y. Erofeeva, "TPXO8-ATLAS," 2013.

[69] S. M. Kelly and P. F. J. Lermusiaux, "Internal-tide interactions with Gulf Stream and Middle Atlantic Bight shelfbreak front," *Journal of Geophysical Research: Oceans*, vol. 121, pp. 6271–6294, 2016.

[70] Y. Pan, P. J. Haley, Jr., and P. F. J. Lermusiaux, "Interactions of internal tides with a heterogeneous and rotational ocean," *Journal of Fluid Mechanics*, vol. 920, p. A18, Aug. 2021.

[71] A. F. Blumberg, R. P. Signell, and H. L. Jenter, "Modeling transport processes in the coastal ocean," *J. Marine Env. Engg.*, vol. 1, pp. 31–52, 1993.

[72] W. Geyer and J. Ledwell, "Boundary mixing in massachusetts bay," MWRA Enviro. Quality Dept. Tech. Rpt. 97–9, MA Water Resources Authority, Boston, MA, 1997. 20 pp.

[73] D. Halpern, *Observations of short period internal waves in Massachusetts Bay.* PhD thesis, MIT, 1969.

[74] A. Gupta and P. F. J. Lermusiaux, "Bayesian learning of coupled biogeochemical-physical models," *Progress in Oceanography*, vol. 216, p. 103050, May 2023.

[75] J. A. Cummings and O. M. Smedstad, *Variational Data Assimilation for the Global Ocean*, vol. 2, pp. 303–343. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[76] National Centers for Environmental Prediction (NCEP), "North American Mesoscale Forecast System (NAM)." https://www.emc.ncep.noaa.gov/index.php?branch=NAM, Sept. 2019.