Robust Ocean Subgrid-Scale Parameterizations Using Fourier Neural Operators

Victor Mangeleer
University of Liège
vmangeleer@uliege.be

Gilles Louppe
University of Liège
g.louppe@uliege.be

Abstract

In climate simulations, small-scale processes shape ocean dynamics but remain computationally expensive to resolve directly. For this reason, their contributions are commonly approximated using empirical parameterizations, which lead to significant errors in long-term projections. In this work, we develop parameterizations based on Fourier Neural Operators, showcasing their accuracy and generalizability in comparison to other approaches. Finally, we discuss the potential and limitations of neural networks operating in the frequency domain, paving the way for future investigation.

1 Introduction

Ocean simulations Conducting ocean simulations has become increasingly important, particularly to understand and fight climate change. Despite growing computing power, a five-year Earth-scale simulation of our oceans at a 1 km resolution still takes a month worth of computation [1]. One way to address this issue is by reducing the simulation resolution to 15 km which makes it run 25 times faster. However, this approach comes at the expense of neglecting a significant portion of subgrid-scale processes contributions, i.e., the (sub)mesoscale (1-10 km) physical phenomena occurring at a scale below the simulation resolution, which highly influence the ocean dynamics.

Parameterizations The challenge of achieving both speed and accuracy in ocean simulations has been approached through the development of parameterizations. The idea is to conduct a low-resolution simulation to take advantage of the fast computational time and correct it at each timestep by adding the missing subgrid-scale contributions using approximate parameterizations. Their development in simulation applications such as turbulent flows [2, 3] and, in our case, geophysical flows [4–7] has been a major research effort. Nevertheless, these parameterizations all encounter the same challenges: the closure problem with extra equations, the restricted working range and the introduction of an artificial viscosity, which disturbs the turbulent nature of the flow. In response to these challenges, the exploration of deep learning-based parameterizations started. For instance, Fully-Convolutional Neural Networks [8] show spatially accurate dynamics correction but have generalization and interpretability limitations. On the other hand, Fourier Neural Operators [9] (FNO) have demonstrated notable improvements in terms of generalization and scale invariance [10–12]. To address the challenge of generalization, attention pivoted towards model discovery techniques [13] to extract analytical parameterizations from raw data even if their results often lag behind neural networks [14], indicating the need for improvements.

Contributions We show the effectiveness of Fourier Neural Operators-based (FNO) parameterizations in addressing generalization issues and accurately correcting simulations. Furthermore, we discuss extensively their advantages and limitations compared to well-established geophysical-flow and deep learning-based parameterizations. All code, datasets, results, and analysis are publicly available at `https://github.com/VikVador/Ocean_FNO`

Machine Learning and the Physical Sciences Workshop, NeurIPS 2023.

2 Background

Ocean Flow Dynamics We focus on developing parameterizations for mid-latitude ocean flows simulations. In this region, the ocean flow dynamics can be simulated using the quasi-geostrophic equations of motion [15]. At each step of the simulation, all flow variables (e.g. velocity fields) are retrieved by solving the prognostic equation for the potential vorticity field q. In a numerical simulation, the actual equation solved is the averaged (denoted $\bar{\cdot}$) prognostic equation given by

$$\frac{\partial \bar{q}}{\partial t} + (\bar{\mathbf{u}} \cdot \nabla)\bar{q} = \bar{\mathcal{F}} + \bar{\mathcal{D}} + \bar{\mathcal{S}}, \tag{1}$$

where $\mathbf{u}$ represents the velocity field vector, $\mathcal{F}$ denotes the forcing terms and $\mathcal{D}$ stands for the dissipation terms. The neglected subgrid-scale processes contributions are taken into account by adding the quantity $\bar{\mathcal{S}}$ at each simulation timestep. Our objective is to develop an accurate parameterization that quickly approximates $\bar{\mathcal{S}}$ to eliminate energy deficiencies.

Fourier Neural Operators FNOs are neural network architectures that represent and process data in the frequency domain through Fourier transforms. Depending on the problem physics, the frequency domain representation is more suited to extract intricate input-output relationships, such as demonstrated in the case of fluid mechanics where FNOs have shown strong performance [10, 12, 16]. An interesting property of this representation is that space-time convolution becomes a simple multiplication in the frequency domain, which leads to faster convolution operations. The FNO architecture faces limitations when scaled to deep architectures, leading to the development of the Factorized Fourier Neural Operator (FFNO) [17]. By rethinking the architecture and enabling shared use of convolution matrices at different computation points, the FFNO achieves higher computational speed, reduces trainable parameters, and facilitates more extensive scaling.

3 Experimental Setup

Data In mid-latitude regions, ocean flows are categorized into two types: *jet-driven flow*, characterized by predominant motion in a single direction, and *eddy-driven flow*, marked by chaotic swirling motion. To simulate these ocean dynamics, we use the two-layer quasigeostrophic model from PyQG [18]. This model offers the necessary complexity to faithfully represent both types of flow. Specifically, the upper layer simulates interactions with the atmosphere, while the lower layer represents interactions with the ocean bottom. It is important to note that these layers also interact with each other. For training, the parameterization takes as inputs $(u,\, v,\, q) \in \mathbb{R}^{H \times H}$ where u and v are the ocean flow longitudinal and latitudinal velocity fields, and H the grid resolution. The parameterization output is the missing subgrid-scale contributions denoted as $\bar{\mathcal{S}}$. The loss function is the mean squared error between the ground truth and the value predicted by the parameterization.

Protocol We assess the quality of our parameterizations through a standardized benchmarking framework [14] and compare against several baselines. These baselines include a Fully-Convolutional Neural Networks (FCNN) [8], U-NET [19], and common empirical geophysical parameterizations such as Smagorinsky [5], Backscattering and Biharmonic [6], and one derived through symbolic regression on raw data [13, 14]. For the neural networks, they use the same inputs, predict identical quantities, and each consists of approximately 300,000 trainable parameters. Furthermore, the FCNN is used in its optimal configuration presented in [14], and the hyperparameters of the U-NET have been adjusted to make its capacity match the one of the others. These two models will not undergo any further fine-tuning. The primary objective is to assess, at a fixed capacity, whether a model exhibits superior performance compared to others. Afterwards, the model demonstrating the most promising capabilities, in this case FFNO, will undergo further tuning for optimization. We conducted two tests: offline and online. **Offline** testing means evaluating how effectively the parameterization predicts $\bar{\mathcal{S}}$ on a dataset composed of 5000 samples drawn from ten distinct, previously unobserved jet-driven simulations. For each fluid layer, we compute the coefficient of determination R^2, yielding a value of 1 for perfect predictions, 0 when predictions are no better than guessing the mean, and negative values when predictions are worse than guessing the mean. **Online** testing means evaluating the parameterization performance at making long-term projections. This involves analyzing whether the parameterized low-resolution simulation dynamics align better with those in the corresponding high-resolution simulation, in contrast to the low-resolution simulation.

To do so, we examine the power spectrum of energy-related flow properties. If the observed spectrum closely matches the one from the high-resolution simulation, it confirms the parameterization effectiveness in redistributing and addressing energy deficiencies across various scales during the simulation.

4 Results

Offline The results are summarized in Tab.1. In the first row, we establish a baseline using the training conditions outlined in [14], which used a dataset of 5000 samples from a single 10-year-long eddy-driven simulation. This allowed us to evaluate both accuracy and generalization across different flow types. Several key observations emerge: both FCNN and U-NET yield negative R^2 results in both layers. FNOs exhibit a positive score in the upper layer. Only FFNO achieves a positive score in both layers. In summary, the FNOs demonstrate superior generalization abilities. In the second row, we investigate the impact of mixing different eddy-driven simulations while maintaining a constant total number of samples. This led to improvements in all R^2 values. In the third row, we increased the number of samples. Surprisingly, we observe a decrease in overall R^2 values, with one exception: FFNO. This suggests that all models may have overfit the eddy dynamics during training. However, we speculate that **the FNOs ability to work with spectral data representations allows it to extract more complex interactions relationships between large-scale and subgrid-scale processes thus leading to better generalization performance**. In the fourth row of results, we explore the impact of training on both ocean flow dynamics. FCNN and U-NET models achieve positive R^2 values in the upper layer but struggle with negative values in the lower layer. In contrast, both FNOs consistently achieve positive R^2 values, highlighting once again the effectiveness of spectral representation computations. As our last experiment, we recognized FFNO as a promising parameterization and we proceeded to fine-tune it across various configurations, with detailed results summarized in the appendix. This led to the development of FFNO*, the best configuration obtained with a substantially larger capacity, i.e. nearly 20 times larger than the one of FFNO. For this last experiment, we observe that **FFNO* achieves unparalleled accuracy, sparking the possibility of extracting a more complex and refined analytical expression for the parameterization through symbolic regression**.

Online As shown in Fig.2, while the FFNO (green) demonstrated good generalization abilities (see Tab.1, third row), it failed to reproduce the high-resolution (blue) simulation spectral profiles. This discrepancy arises from its limitations in accurately predicting the lower layer dynamics, which significantly impacts the long-term simulation and once again highlight the importance of finding good parameterizations. In contrast, FFNO* (red) displayed superior and consistent results across all flow variables (except APEflux) compared to the baselines (grey). Indeed, the baseline models exhibited inconsistencies, either causing spectral explosion or flattening. Despite the fact that FFNO* achieves superior results with comparable capacity to other models, it faces the drawback of using twenty-four Fourier layers, as opposed to the original FNO. Consequently, **the speed-up due to utilizing a deep-learning-based parameterization for low-resolution simulations is compromised**. This is shown in Tab.2, where the increased computational time, attributed to the frequent computation of Fourier transform and inverse Fourier transform, surpasses the one of the high-resolution simulation.

In conclusion, leveraging the frequency representation of data allows Fourier Neural Operators to exhibit superior generalization by capturing multiscale interactions. Nevertheless, there is still a need to enhance the FFNO architecture. This improvement is crucial not only for refining parameterization generalization capabilities but also for reducing simulation computational time, ultimately making the development of Fourier Neural Operator-based parameterizations worthwhile.

5 Conclusion

Our exploration of Fourier Neural Operators (FNOs) as subgrid-scale parameterizations demonstrates their superior ability to generalize when predicting missing subgrid-scale contributions in new ocean dynamics. However, further refinement is needed for the learned parameterization since its predictions lack the necessary accuracy for long-term projections. When learning a parameterization across all possible ocean dynamics, FNOs again outperform the baselines in terms of generalization. This reinforces the notion that FNOs are better suited for this task. Moreover, after optimization, FNOs achieved remarkable accuracy levels. Yet, we must acknowledge that the computational time for the parameterized low-resolution model is three times longer than that of the high-resolution counterpart thus making it unusable. Furthermore, while the parameterization effectively corrected energy deficiencies, spatial accuracy remained a concern. Even if the predictions were accurate, their incorrect placement had a noticeable impact on flow dynamics.

Table 1: Analysis of our parameterizations accuracy by calculating the coefficient of determination (R^2) for both ocean model fluid layers. A R^2 value close to 1 indicates better results. The *Training Datasets* column provides information on training sample sizes and the number of simulations used. Testing is performed using a dataset of jets-driven flow simulations. The key observations are: FFNO generalizes better in every situations (row 1 to 3). When trained on both ocean dynamics, FCNN and U-NET perform poorly on the lower layer whereas the FNOs obtained positive score suggesting again better generalization capabilities. When the training and architecture are optimized, FFNO* reaches new levels of accuracy suggesting that extracting the analytical expression of the learned parameterization is a path worth to explore.

Training Datasets			Upper layer R^2 / Lower layer R^2				
Samples [10^3]	Eddies	Jets	FCNN	U-NET	FNO	FFNO	FFNO*
5	1	0	-0.21 / -10.60	-0.69 / -16.30	0.18 / -0.92	0.76 / 0.12	-
5	10	0	-0.23 / -11.80	-0.40 / -12.34	0.42 / -0.23	0.78 / 0.24	-
20	40	0	-0.34 / -15.20	-0.43 / -14.40	0.41 / -0.34	**0.86** / **0.37**	-
5	5	5	0.40 / -5.75	0.32 / -4.91	0.49 / 0.13	0.83 / 0.49	**0.99** / **0.93**

Table 2: Analysis of the computational time for a 10-year jet-driven flow scenario simulation reveals the following: The first two columns display the baseline times for high and low-resolution simulations, while the next four columns show the total time needed for a parameterized low-resolution simulation. The speed-up metric quantifies the improvement in computational time compared to a high-resolution simulation. A speed-up above 1 indicates a positive gain—meaning the simulation is faster, with higher values being better—while a speed-up below 1 signifies a negative gain, indicating slower performance than the high-resolution simulation. Key findings indicate that only the FCNN, U-NET, and FNO parameterizations yield positive speed-ups, despite their worst predictive accuracy. On the other hand, while FFNO and FFNO* exhibit superior correction capabilities, their simulation times surpass that of a high-resolution simulation, negating the advantage of using parameterizations. This discrepancy is attributed to the FFNO's twenty-four Fourier layers compared to the FNO, resulting in significant computational overhead due to the numerous Fourier transformations and inverse Fourier transforms required between layers.

	Baselines [min]		Parameterized low-resolution (LR) [min]				
	High-resolution	Low-resolution	FCNN	U-NET	FNO	FFNO	FFNO*
	22.25	2.10	11.09	13.05	11.397	46.89	61.68
Speed-up [-]	1.00	10.58	2.00	1.70	**1.95**	**0.47**	**0.36**

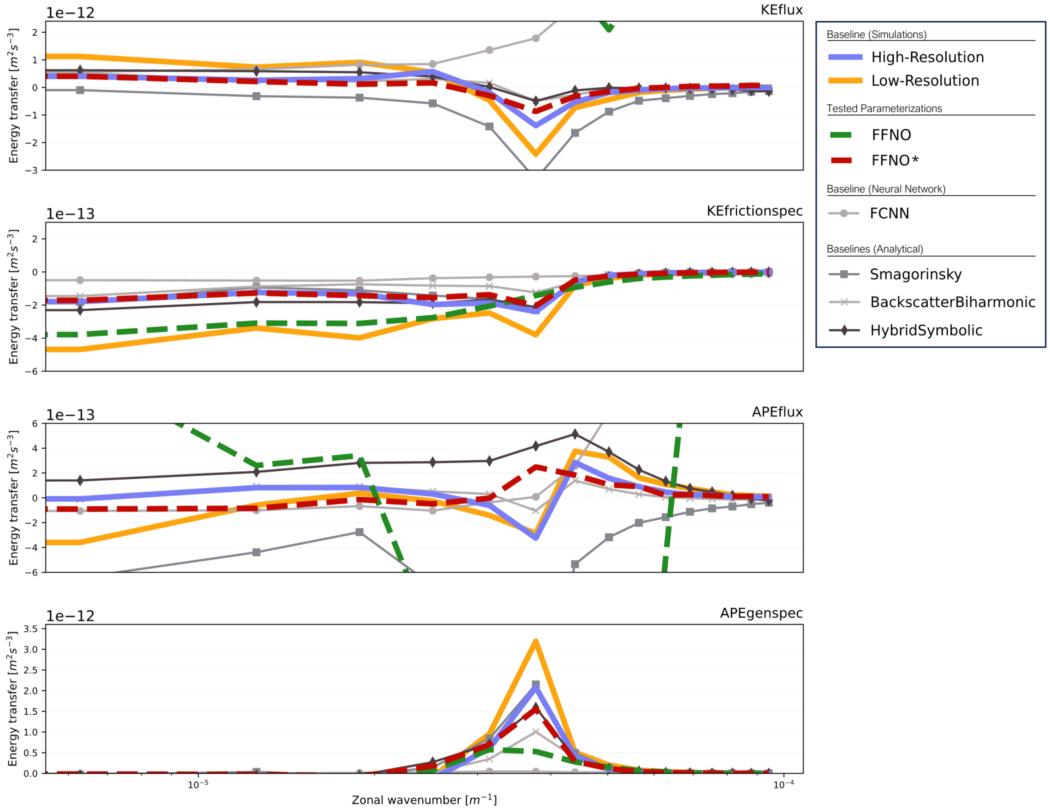

Figure 1: Analysis of energy transfer contributions (y-axis) across various length scales (x-axis) for different flow quantities. The objective is to align the power spectrum with that of the high-resolution (blue). The observed flow quantities are: *KEflux* is the kinetic energy transfer, while *KEfrictionspec* assesses energy dissipation due to friction with the bottom of the ocean. *APEflux* measures available potential energy for transfer, and *APEgenspec* tracks newly generated potential energy at each scale. Observations reveal that while FFNO (green) exhibits good offline generalization (see Tab.1, third row), its predictions lack the accuracy necessary for a stable long-term simulation, resulting in spectra divergence from the high-resolution. Conversely, the optimized FFNO* (red), trained on both flow dynamics types, demonstrates improved matching with the high-resolution spectra (except for APEflux). Baselines (grey) do not achieve the same consistency as FFNO*, they either excessively amplify the spectra, worsening them compared to the low-resolution (orange), or significantly dampen them (indicated by the flat lines for KEfrictionspec).

6 Future work

For future work, we propose investigating novel architectures based on Fourier Neural Operators. To address spatial challenges, we suggest exploring a hybrid approach by combining space-time convolution with spectral convolution. In terms of computational efficiency, there is potential for improvement in the design of Fourier layers to reduce their number. Finally, we find it intriguing to explore the application of symbolic regression to extract learned parameterizations.

Acknowledgments and Disclosure of Funding

This work was supported by Service Public de Wallonie Recherche under Grant No. 2010235 - ARIAC by DIGITALWALLONIA4.AI. In addition to that, the authors would like to thank François Rozet, Gerome Andry, Omer Rochman and Sacha Lewin for their support and insightful feedback.

References

[1] Oliver Fuhrer et al. “Near-global climate simulation at 1 km resolution: establishing a performance baseline on 4888 GPUs with COSMO 5.0”. In *Geoscientific Model Development* 11.4 (2018), pp. 1665–1681.

[2] AK Singhal and DB Spalding. “Predictions of two-dimensional boundary layers with the aid of the k-epsilon model of turbulence”. In *Computer Methods in Applied Mechanics and Engineering* 25.3 (1981), pp. 365–383.

[3] Florian R Menter. *Improved two-equation k-omega turbulence models for aerodynamic flows*. Tech. rep. 1992.

[4] Ludwig Prandtl. “7. Bericht über Untersuchungen zur ausgebildeten Turbulenz”. In *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 5.2 (1925), pp. 136–139.

[5] Joseph Smagorinsky. “General circulation experiments with the primitive equations: I. The basic experiment”. In *Monthly weather review* 91.3 (1963), pp. 99–164.

[6] Malte F Jansen and Isaac M Held. “Parameterizing subgrid-scale eddy effects using energetically consistent backscatter”. In *Ocean Modelling* 80 (2014), pp. 36–48.

[7] Malte F Jansen et al. “Energy budget-based backscatter in an eddy permitting primitive equation model”. In *Ocean Modelling* 94 (2015), pp. 15–26.

[8] Thomas Bolton and Laure Zanna. “Applications of deep learning to ocean data inference and subgrid parameterization”. In *Journal of Advances in Modeling Earth Systems* 11.1 (2019), pp. 376–399.

[9] Zongyi Li et al. “Fourier Neural Operator for Parametric Partial Differential Equations”. In *International Conference on Learning Representations*. 2021.

[10] Jaideep Pathak et al. “Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators”. In *arXiv preprint arXiv:2202.11214* (2022).

[11] Boris Bonev et al. “Spherical Fourier Neural Operators: Learning Stable Dynamics on the Sphere”. In *arXiv preprint arXiv:2306.03838* (2023).

[12] Gege Wen et al. “U-FNO—An enhanced Fourier neural operator-based deep-learning model for multiphase flow”. In *Advances in Water Resources* 163 (2022), p. 104180.

[13] Laure Zanna and Thomas Bolton. “Data-driven equation discovery of ocean mesoscale closures”. In *Geophysical Research Letters* 47.17 (2020), e2020GL088376.

[14] Andrew Ross et al. “Benchmarking of machine learning ocean subgrid parameterizations in an idealized model”. In *Journal of Advances in Modeling Earth Systems* 15.1 (2023), e2022MS003258.

[15] Stephen Griffies. *Fundamentals of ocean climate models*. Princeton university press, 2018.

[16] Thorsten Kurth et al. “Fourcastnet: Accelerating global high-resolution weather forecasting using adaptive fourier neural operators”. In *Proceedings of the Platform for Advanced Scientific Computing Conference*. 2023, pp. 1–11.

[17] Alasdair Tran et al. “Factorized Fourier Neural Operators”. In *The Eleventh International Conference on Learning Representations*. 2023.

[18] Nora Loose et al. “GCM-filters: A Python package for diffusion-based spatial filtering of gridded data”. In *Journal of Open Source Software* 7.70 (2022).

[19] Olaf Ronneberger et al. “U-net: Convolutional networks for biomedical image segmentation”. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer. 2015, pp. 234–241.

7 Appendix

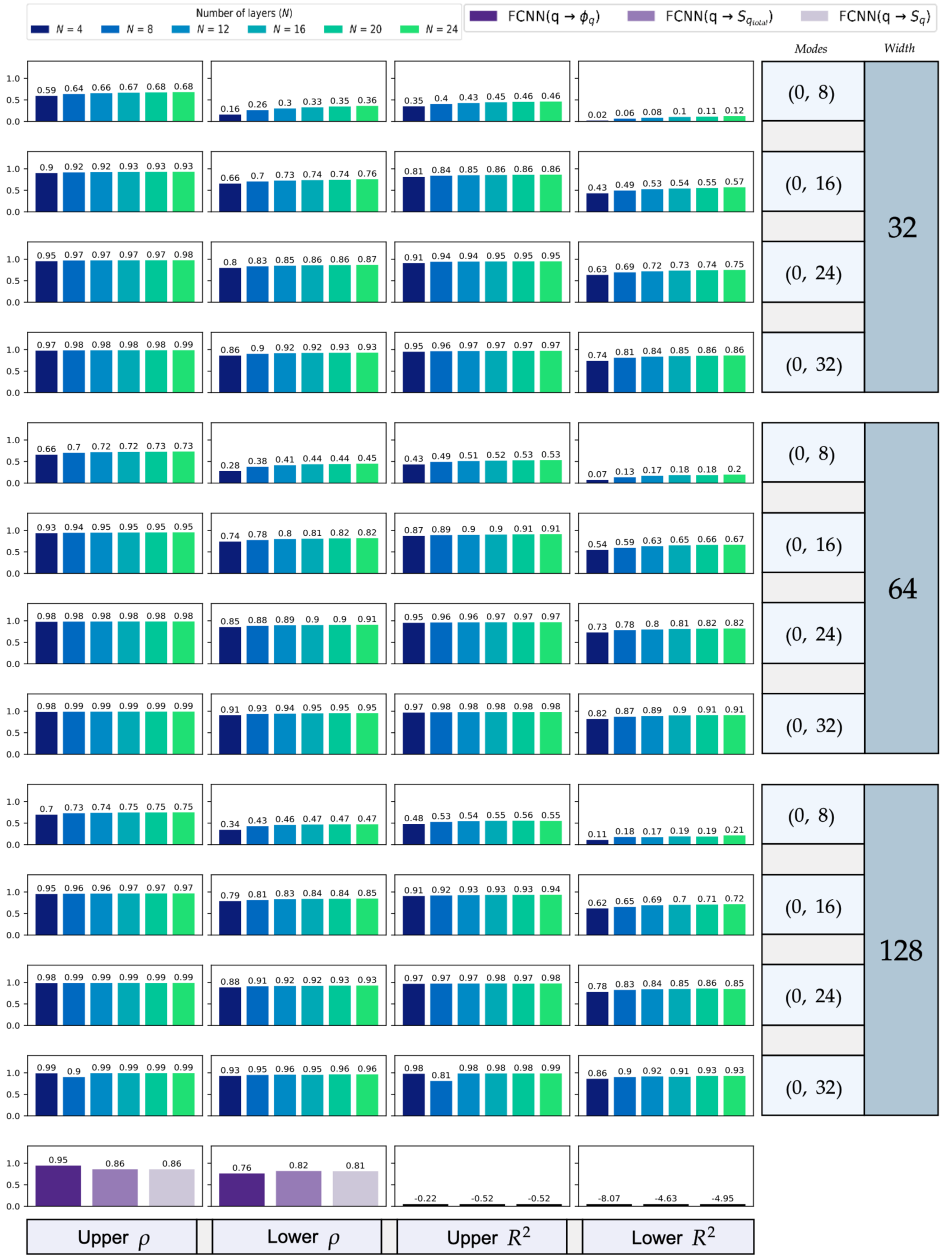

Figure 2: Sensitivity analysis of FFNO hyperparameters on jet-driven flow. Models trained on a dataset with eddies and jet-driven flows are evaluated offline. The first two columns depict correlations in the upper and lower layers, while the next two show R^2 coefficients. Hyperparameters include *width* (latent space size), the number of Fourier layers N, and the range of Fourier modes. Notably, the number of retained modes appears to have the most impact, with broader ranges yielding better results. Increased Fourier layers generally show a minor impact but significantly prolong the forward pass time due to additional Fourier transform computations. Enhanced *width* improves model capacity and results, with a lesser impact on computational time compared to an increase in depth. As a result, FFNO uses a width of 128, the first 32 modes, and 24 Fourier layers.