



RESEARCH ARTICLE

10.1029/2025MS005667

Special Collection:
The CliMA Earth System Model

NORi: An ML-Augmented Ocean Boundary Layer Parameterization

Xin Kai Lee^{1,2,3} , Ali Ramadhan^{1,2,4} , Andre Souza¹ , Gregory LeClaire Wagner^{1,5} ,
Simone Silvestri^{1,6} , John Marshall^{1,2} , and Raffaele Ferrari^{1,2} 

¹Department of Earth, Atmospheric and Planetary Sciences, Massachusetts Institute of Technology, Cambridge, MA, USA,

²Center for Computational Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA,

³Department of Physics, Imperial College London, London, UK, ⁴Now at atdepth, Cambridge, MA, USA, ⁵Now at Aeolus Labs, San Francisco, CA, USA, ⁶Now at Department of Environment, Land and Infrastructure Engineering, Politecnico di Torino, Torino, Italy

Key Points:

- NORi uses neural networks to augment a Richardson number-based closure of turbulent vertical mixing in the ocean boundary layer
- A posteriori calibration, where loss depends on variable trajectories rather than instantaneous fluxes, ensures robust numerical stability
- NORi outperforms baselines on idealized simulations, closely tracks column observations, and is numerically stable in an idealized 3D setup

Correspondence to:

X. K. Lee,
me@xinkailee.com

Citation:

Lee, X. K., Ramadhan, A., Souza, A., Wagner, G. L., Silvestri, S., Marshall, J., & Ferrari, R. (2026). NORi: An ML-augmented ocean boundary layer parameterization. *Journal of Advances in Modeling Earth Systems*, 18, e2025MS005667. <https://doi.org/10.1029/2025MS005667>

Received 3 DEC 2025
Accepted 6 AUG 2026

Author Contributions:

Conceptualization: Xin Kai Lee,

Ali Ramadhan, John Marshall

Data curation: Xin Kai Lee

Formal analysis: Xin Kai Lee

Funding acquisition: John Marshall,
Raffaele Ferrari

Investigation: Xin Kai Lee

Methodology: Xin Kai Lee,

Ali Ramadhan, Andre Souza,

Gregory LeClaire Wagner,

Simone Silvestri, John Marshall,

Raffaele Ferrari

Abstract NORi is a machine learning (ML) parameterization of ocean boundary layer (BL) turbulence that is physics-based and augmented with neural networks. NORi stands for neural ordinary differential equations Richardson number (Ri) closure. The physical parameterization is controlled by a Richardson number-dependent diffusivity and viscosity. The neural ODEs are trained to capture the entrainment through the base of the BL, which cannot be represented with a local diffusive closure. The parameterization is trained using large-eddy simulations in an a posteriori fashion, where parameters are calibrated with a loss function that explicitly depends on the actual time-integrated variables of interest rather than the instantaneous subgrid fluxes, which are inherently noisy. NORi conserves tracers by design, uses realistic nonlinear thermodynamics, and demonstrates excellent prediction and generalization capabilities in capturing entrainment dynamics under different convective strengths, background stratifications, rotation, and wind forcings. NORi is shown to simulate the seasonal evolution of the BL at Ocean Weather Station Papa with similar performance to the state-of-the-art two-equation $k - \epsilon$ closure. When implemented in a double-gyre simulation, it is numerically stable for at least 100 years, despite only being trained on 2-day horizons, and can be run with time steps as long as 1 hr. Combining highly expressive neural networks with a physically grounded base closure proves to be a robust paradigm for designing parameterizations for climate models: data required and training cost are drastically reduced, inference performance can be directly optimized as a primary objective, and numerical stability is implicitly promoted through training.

Plain Language Summary Climate models cannot represent small-scale ocean turbulence at scales of 1–100 m because the immense computational cost means they can, at best, resolve 10-km features. These mixing processes, driven by winds, evaporation, and surface cooling, affect how heat, salinity, and other properties are exchanged between the atmosphere and ocean interior, influencing climate predictions. We introduce NORi, a new approach that combines traditional physics-based equations with modern machine learning (ML) to better represent these processes. Traditional methods often sacrifice accuracy for speed or vice versa. NORi achieves both by using neural networks to enhance simpler physics-based models where they fall short. What makes NORi effective is our training approach: using high-resolution simulations as “ground truth” and focusing on predicting evolution over time rather than matching instantaneous snapshots. This produced a model that remains stable when run for 100 years, despite being trained on two-day-long data sets, and compares favorably to observations from a long-term monitoring site in the Northeast Pacific. NORi has a lower computational cost than traditional models, making it practical for long-term climate simulations. By bridging physics and ML in this way, NORi represents a new paradigm for fast and accurate models that require less training.

1. Introduction

State-of-the-art global ocean models used in climate studies solve the equations that govern ocean dynamics and thermodynamics on grids with resolutions of 10 km or coarser (Hewitt et al., 2020; Silvestri et al., 2025). At this resolution, many subgrid-scale processes cannot be explicitly resolved. In the upper ocean boundary layers (BLs), these include turbulent mixing driven by cooling, evaporation, and wind stresses on scales from a few hundred meters down to centimeters. Despite occurring at small spatial scales, this mixing has a fundamental impact on the

Resources: John Marshall, Raffaele Ferrari
Software: Xin Kai Lee, Ali Ramadhan, Simone Silvestri
Supervision: John Marshall, Raffaele Ferrari
Validation: Xin Kai Lee, Raffaele Ferrari
Visualization: Xin Kai Lee, Raffaele Ferrari
Writing – original draft: Xin Kai Lee
Writing – review & editing: Xin Kai Lee, Ali Ramadhan, Andre Souza, Gregory LeClaire Wagner, Simone Silvestri, John Marshall, Raffaele Ferrari

large-scale structure of the ocean by regulating the exchange of heat, carbon, and other climatically important tracers between the atmosphere and the ocean interior. Subgrid-scale parameterizations are therefore introduced in climate models to represent the impact of these processes on large-scale variables.

Traditional parameterizations of turbulent mixing in the upper ocean rely on known physics, typically encoded in scaling laws with a number of free parameters that are determined with empirical data. The most parsimonious parameterizations are zero-equation first-order diffusive closures, where vertical diffusivities are increased when surface winds, cooling, and/or evaporation are strong enough to trigger turbulent mixing. The Pacanowski–Philander parameterization (Pacanowski & Philander, 1981) is one such example where the vertical diffusivity is a function of local stratification and vertical shear via the Richardson number. Although this diffusive closure smooths local gradients of momentum, temperature, and salinity, it cannot accurately represent the entrainment at the base of the BL. Entrainment is driven by convective plumes formed by surface cooling and/or evaporation that penetrate beneath the mixed layer and draw denser water from the thermocline into it. This process sharpens the gradient at the BL base—an inherently anti-diffusive, nonlocal mechanism, as illustrated in Figure 1. To represent processes that are not described purely by downgradient diffusive first-order closures, the widely used K-Profile Parameterization (KPP, Large et al., 1994) adds a nonlocal flux term to the diffusive flux. This nonlocal term does not depend on local gradients and can capture nonlocal entrainment fluxes at the BL base (Noh et al., 2003). However, bulk parameterizations, originally proposed by Niiler (1977) and recently adopted for use in large-scale ocean models (ePBL, Reichl & Hallberg, 2018), are formulated only in terms of vertically averaged properties of the BL through bulk energetic arguments. This can be thought of as a fully nonlocal model, since the behavior at the BL base depends explicitly on the bulk properties of the entire mixed layer column above it, including surface fluxes. Two-equation first-order models, such as the celebrated $k - \epsilon$ parameterizations and related generic length-scale (GLS) closures (Umlauf & Burchard, 2003), address the nonlocality of turbulent mixing by formulating equations that predict the evolution and transport of quantities such as turbulent kinetic energy (TKE) and energy dissipation. In these models, the tracer and momentum fluxes are based on local eddy diffusivities while the TKE and dissipation equations include nonlocal transport terms. Although more accurate than zero-equation closure models, they are not widely used in global climate models, as they require shorter time steps than the equations for large-scale ocean dynamics and therefore add substantial computational cost (Reffray et al., 2015; Reichl & Hallberg, 2018). Most recently, the Convective Adjustment TKE parameterization (CATKE, Wagner, Hillier et al., 2025) has been introduced to leverage the benefits of two-equation models while reducing their computational cost. CATKE is a 1.5-equation closure that uses a prognostic equation for TKE to compute the diffusivity. Its skill compares favorably with the $k - \epsilon$ parameterization at a computational cost comparable to that of KPP.

Despite much progress in the formulation of BL parameterizations, substantial biases persist in the simulation of the upper ocean in large-scale ocean models, especially in the Southern Ocean (DuVivier et al., 2018; Sallée et al., 2013; Treguier et al., 2023) and the tropics (Li & Xie, 2014). In particular, the BL depth bias is often of the same order as the depth itself (DuVivier et al., 2018; Treguier et al., 2023).

Machine learning (ML) methods offer a potential route to reduce these biases and are being increasingly used in the parameterization of climate models for subgrid-scale processes thanks to advances in computing hardware and ML software (Bolton & Zanna, 2019; Zanna et al., 2025). For upper ocean BL turbulence, a Bayesian approach has been used to tune and uncover structural deficiencies of KPP (Souza et al., 2020). Neural networks have been used to predict the evolution of temperature and salinity in the Northeast Pacific Ocean (Liang et al., 2022), improve the ePBL parameterization (Sane et al., 2023), and calibrate the vertical diffusivity profile in KPP (Yuan et al., 2024). Equation-discovery approaches have been investigated as a means to enhance the ePBL parameterization while preserving its physical interpretability (Sane et al., 2026). While all these approaches have reduced some biases, none has been completely successful. For example, Sane et al. (2023, 2026) show that ML-enhanced ocean BL parameterizations in Ocean Model Intercomparison Project configurations reduce mixed layer depth biases in the summer but not in the winter.

One of the most important challenges facing neural network-based parameterizations applied to climate modeling is the loss of numerical stability and physical consistency over long integration times (Chattopadhyay et al., 2023). The causes of such numerical instabilities are manifold, including the spectral bias of neural networks, which prioritize learning low-frequency dynamics at the expense of accurate high-frequency dynamics, essential to represent turbulence and chaotic processes (Chattopadhyay et al., 2023), as well as the use of a priori

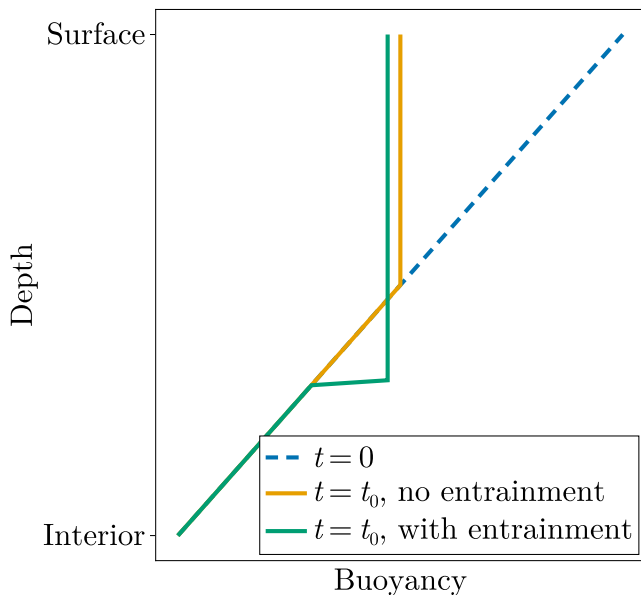


Figure 1. Schematic of the vertical profile of buoyancy as a function of depth below the ocean surface—buoyancy decreases linearly with the density anomaly generated by a change in temperature and salinity. At time $t = 0$, the buoyancy decreases linearly with depth. After some time, in response to buoyancy loss at the surface, a well-mixed layer develops in the upper ocean. In the absence of entrainment through the base of the mixed layer, the buoyancy profile smoothly connects to the stratified interior. Entrainment results in a further deepening of the boundary layer and the development of a sharp buoyancy jump at its base.

calibration to train the neural network (Frezat et al., 2022). In a priori calibration, the neural network parameterization is trained offline to match the diagnosed, unresolved subgrid-scale fluxes as a function of large-scale variables. Since no parameterization is perfect, once implemented in the dynamical equations of the climate model, it will introduce some random errors in addition to the desired fluxes. On top of that, there are unavoidable numerical errors due to the spatial and temporal discretization of the PDEs. Over time, these errors accumulate and amplify, leading to the climate variables diverging from the correct trajectories, not unlike how trajectories of chaotic systems diverge in response to infinitesimal perturbations. This challenge has been well documented in data-driven neural network-based weather forecast models, which can be considered the state-of-the-art in the application of ML tools to atmospheric models, including FourCastNet (Pathak et al., 2022), GraphCast (Lam et al., 2023), and Pangu (Bi et al., 2023). These models can develop instabilities or accumulate errors when run autoregressively over long times (Chattopadhyay et al., 2023).

A recently suggested alternative to obtain stable solutions with neural networks is a posteriori training, albeit with an increase in training cost. In a posteriori training, the parameterization is trained to match the temporal evolution of coarse-grained variables rather than the instantaneous subgrid-scale fluxes. The parameterized fluxes are not required to exactly match those in the training data set, as the training data are noisy and the fluxes required to produce the correct tendencies are not unique. Instead, the parameterization is trained end-to-end by evolving the coarse-grained variables over time and including the entire solution's evolution in the loss function. A key advantage of this approach is that it aligns the training data distribution with the conditions expected during inference, unlike a priori training, where the data distribution at inference might differ from that during

training due to drifts in prognostic variables. In addition, a posteriori training naturally accounts for numerical discretization errors, as models are trained on trajectories of coarse-grained variables over time. Examples of parameterizations calibrated a posteriori include closures for 2D turbulence (Kochkov et al., 2021), quasi-geostrophic turbulence (Frezat et al., 2022), 3D turbulence (Sirignano et al., 2020; Stachenfeld et al., 2021), the viscous Burgers' equation (Kang & Constantinescu, 2023), as well as for BL turbulence (Wagner, Hillier, et al., 2025). Another notable example of a posteriori calibration in earth system modeling is NeuralGCM (Kochkov et al., 2024), where the model variable tendencies were used to train the neural networks, rather than the more traditional approach of training a priori on eddy fluxes.

A challenge of a posteriori calibration is that it requires the forward model to be fully automatic-differentiable and is serial in time, resulting in a high computational and memory cost. For physically based models, this can be mitigated with ensemble-based methods such as Ensemble Kalman Inversion (EKI) (Dunbar et al., 2022; Iglesias et al., 2013) that do not require differentiability, but these methods perform well only when parameter counts are small and priors are informative (Gjini et al., 2025). Neural network models have large numbers of free parameters and informative priors are difficult to find because the parameters do not represent interpretable physical processes. The differentiability bottleneck has been addressed through modern languages that offer libraries for automatic differentiation (AD), such as JAX (Frostig et al., 2018) in Python and Lux.jl (Pal, 2023) and Enzyme.jl (Moses & Churavy, 2020) in Julia.

We present NORi (pronounced noh-ree, a dried edible seaweed in Japanese): an ocean BL parameterization that combines the strengths of physics-based and machine-learned parameterizations. The philosophy is to start with as simple a physics-based parameterization as possible and then augment it with a data-trained neural network. The physics-based model is a first-order diffusive closure similar to the Pacanowski–Philander model (Pacanowski & Philander, 1981) where the eddy diffusivity is a simple function of the local gradient Richardson number. To capture the additional mixing physics of entrainment, the parameterization is augmented with neural networks trained a posteriori with high-resolution simulations of upper ocean turbulence. The a posteriori calibration uses a neural ordinary differential equation framework (R. T. Q. Chen et al., 2019; Rackauckas

Table 1
Overview of the NORi Framework and Training Approach

Aspect	Description
Overall approach	Augments a simple physics-based model with neural networks to capture missing physics
Physics-based component	<i>Local</i> gradient Richardson number-based eddy-diffusivity closure to capture local mixing
Neural network component	<i>Nonlocal</i> entrainment fluxes of temperature and salinity at the base of the mixed layer due to convective plumes
Neural network design	Small feedforward networks (3 hidden layers $\times$ 128 ReLU units) shared across vertical grid points
Neural network inputs	Local gradients of temperature, salinity, and potential density, the local Richardson number, and the nonlocal surface buoyancy flux
Training data	Large-eddy simulations (LES) spanning convective and shear turbulence regimes, with varying rotation, background stratification, and mean temperatures and salinities
Training strategy	Trained on the full temporal evolution of temperature and salinity rather than on instantaneous turbulent fluxes
Validation and testing	Held-out LES scenarios and the Ocean Weather Station Papa observations for accuracy; idealized 3D double-gyre simulation for numerical stability

et al., 2021) where the parameterization is integrated forward in time within fully differentiable ODE solvers. NORi considers the full nonlinear equation of state (Roquet et al., 2015) and provides a closure for the complete suite of prognostic variables for the dynamics of the ocean BL: momentum, temperature, and salinity. We show that incorporating known physics into the parameterization reduces the amount of data required for training, allows the use of a small neural network, and improves the model's generalization ability and numerical stability. We find that NORi compares favorably in accuracy and computational performance with the $k - \epsilon$ parameterization (Umlauf & Burchard, 2005), considered a gold standard for high-resolution BL simulations, and the CATKE parameterization (Wagner, Hillier, et al., 2025), a recently developed closure for use in global ocean simulations. Like all BL vertical mixing schemes, NORi parameterizes processes that deepen the BL, assuming that the large-scale model resolves lateral mesoscale and submesoscale instabilities that restratify the BL. The novelty of NORi lies in the design paradigm as well as in the training strategy, which are applicable for the design of neural network-based parameterizations in general. Instead of using a large neural network to learn the entire parameterization scheme or restricting a neural network to calibrate specific parameters in a complex physics-based model, we design a simple scheme that captures the bulk of the physical process to be parameterized and then leverage the expressivity of neural networks to learn the residual physics that the base scheme does not capture.

Section 2 introduces the high-resolution, high-fidelity, large-eddy simulations (LES) that we use to train and validate NORi. Section 3 describes the formulation of the NORi column model. Section 4 describes the local eddy-diffusivity closure which encodes the known physics. Section 5 describes the neural network formulation and the a posteriori training paradigm used in NORi. We then assess NORi's performance in single-column configurations in Section 6 and in a large-scale long-time double-gyre ocean simulation in Section 7. Finally, Section 8 presents our conclusions. Table 1 and the schematic in Figure 2 summarize the key components and design choices of NORi.

2. High-Resolution Data Set for Boundary Layer Turbulence

A parameterization must, first and foremost, accurately represent the physics of the processes it parameterizes. Therefore, a well-designed suite of “ground-truth” data is required for the training, validation, and testing of NORi. To that end, we run a wide range of high-resolution LES of upper ocean turbulence, which aim to cover the conditions under which BL turbulence occurs in the real ocean.

2.1. Large-Eddy Simulations

LES are simulations that resolve the largest turbulent eddies in the BL and use a subgrid-scale model to represent the smaller, more universal 3D eddies. LES are generated with Oceananigans.jl (Ramadhan et al., 2020; Wagner,

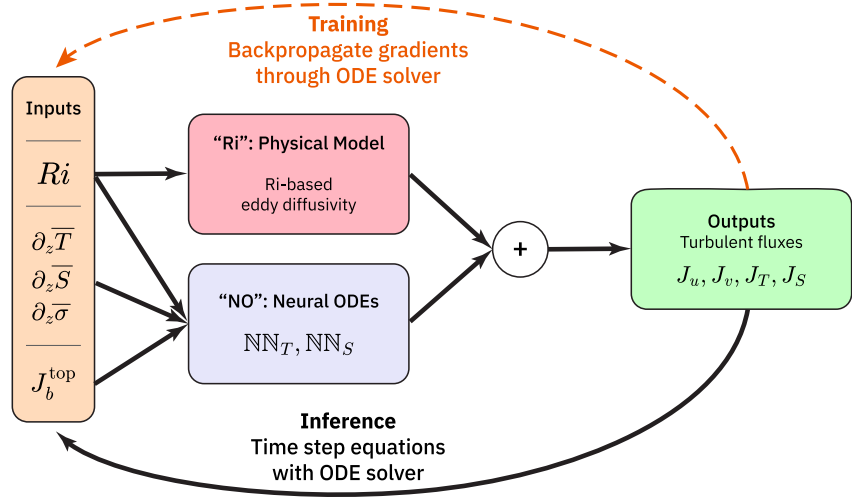


Figure 2. Schematic summary of NORi's design and training framework. NORi consists of a physics-based eddy-diffusivity closure based on the gradient Richardson number (“Ri”) augmented by neural ODEs (“NO”). The physics-based closure takes local gradient Richardson number as input and captures local mixing, while the neural networks take Richardson number, temperature, salinity, and potential density gradients, and the surface buoyancy flux as inputs and capture nonlocal entrainment fluxes at the base of the mixed layer. The base-closure outputs are summed with the neural-network entrainment fluxes for temperature and salinity (the base closure alone provides the momentum flux), producing the turbulent fluxes used to step the column-model equations forward in time. Calibration is done over the full time trajectory by backpropagating gradients through the ODE solve to update neural network parameters.

Silvestri, et al., (2025), a finite volume ocean model written in Julia (Bezanson et al., 2017) and optimized for GPUs (Silvestri et al., 2025). The LES data suite focuses on shear-driven and convection-driven deepening of the BL against a stable background stratification. Mathematically, LES solve the incompressible Boussinesq form of the Navier–Stokes equations given by

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla \cdot (\mathbf{u} \otimes \mathbf{u}) - f \hat{\mathbf{k}} \times \mathbf{u} - \nabla p + b \hat{\mathbf{k}}, \quad (1)$$

$$\frac{\partial T}{\partial t} = -\nabla \cdot (\mathbf{u} T), \quad (2)$$

$$\frac{\partial S}{\partial t} = -\nabla \cdot (\mathbf{u} S), \quad (3)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (4)$$

$$b = -g \frac{\sigma - \sigma_0}{\sigma_0}, \quad (5)$$

$$\sigma = \sigma_{\text{TEOS-10}}(T, S, z = 0), \quad (6)$$

where $\mathbf{u}$ are the velocities in the x , y , and z directions, f is the Coriolis parameter, $\hat{\mathbf{k}}$ is the unit vector in the vertical direction (normal to the fluid surface), p is the kinematic pressure, b is the buoyancy of the fluid, T and S are the temperature and salinity of the fluid, σ is the potential density, g is the gravitational acceleration, and $\sigma_0 = 1,020 \text{ kg m}^{-3}$ is the reference density. The LES are run with laterally doubly periodic boundary conditions, prescribed advective fluxes of T , S , and u at the top given by J_T^{top} , J_S^{top} , and J_u^{top} , with $J_v^{\text{top}} = 0$ in all simulations, and linear gradient boundary conditions for T and S and free-slip boundary conditions for u and v at the bottom. Additionally, we assume that the vertical velocities at the top and bottom surfaces are zero, imposing the no-penetration boundary condition given by

$$w(z = 0) = w(z = -L_z) = 0. \quad (7)$$

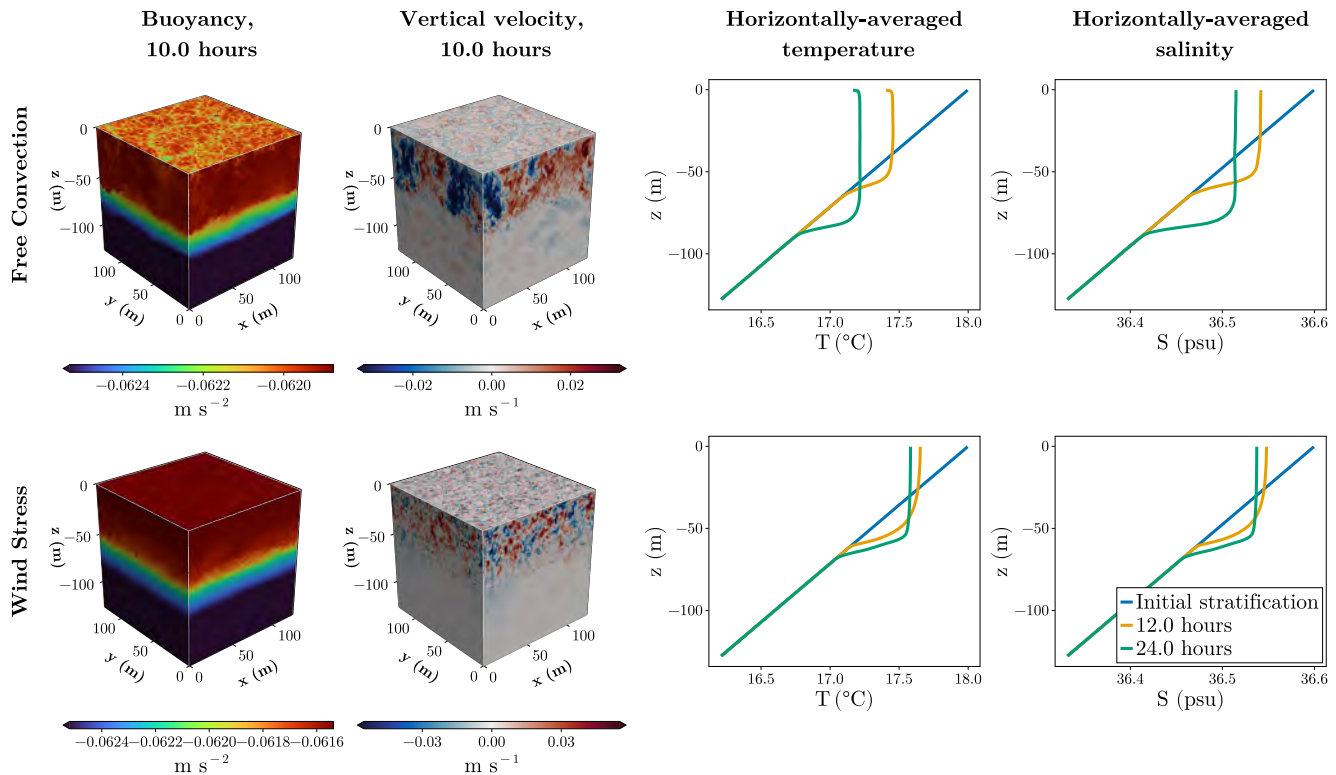


Figure 3. Large-eddy simulations (LES) for free convection and pure wind stress scenarios in a horizontally doubly periodic domain of size $(L_x, L_y, L_z) = (128, 128, 128)$ m with a grid resolution of 0.5 m and a Coriolis parameter of $f = 8 \times 10^{-5} \text{ s}^{-1}$. The top row shows a convective turbulence LES driven by surface cooling, while the bottom row shows a shear turbulence LES driven by surface wind stress. The first and second columns show snapshots of the buoyancy and vertical velocity fields. The third and fourth columns show the time evolution of the horizontally averaged temperature and salinity profiles characterized by a deepening mixed layer in response to surface forcing.

To model a realistic ocean, we use the TEOS-10 equation of state (Roquet et al., 2015), which takes into account the nonlinear dependence of buoyancy on temperature and salinity. This nonlinear dependence can lead to effects such as cabelling, which requires that temperature and salinity be modeled separately. The prognostic variables in Equations 1–4 are evolved using a third-order Runge–Kutta scheme with a pressure projection (or predictor–corrector) method that involves the solution of a three-dimensional Poisson equation for pressure (Schumann & Sweet, 1988).

The simulations do not use an explicit LES closure; instead, they employ a ninth-order weighted essentially non-oscillatory (WENO, Balsara & Shu, 2000) advection scheme. This dissipative scheme acts as an implicit viscous closure, capturing kinetic energy dissipation while maintaining high resolution through gradient-preserving properties (Silvestri et al., 2024). The implicit LES (ILES) formulation (Grinstein et al., 2007) has been shown to outperform explicit closures in ocean and atmospheric simulations where turbulence coexists with internal waves (Z. Chen et al., 2025; Pressel et al., 2017; Silvestri et al., 2024; Smolarkiewicz et al., 2007; Wagner, Hillier, et al., 2025). The ILES prevents oscillatory errors that lead to spuriously large diffusivities in explicit closures at sharp density interfaces, such as BL bases, where internal wave shears can be large without generating significant mixing. In particular, ILES with prescribed surface fluxes used in the Coastal and Regional Ocean Community model (CROCO) has been shown to give results comparable to the NCAR-LES (Fan et al., 2024) and the commonly used explicit Smagorinsky closure to simulate BL turbulence in convective and shear turbulent regimes with vertical resolutions below a few meters (Z. Chen et al., 2025).

The snapshots in Figure 3 show the simulated buoyancy and vertical velocity fields of two representative examples of upper ocean turbulence: free convection and shear turbulence acting on a fluid initialized with constant vertical stratification. These simulations, used for illustration purposes in Figure 3, are run with an isotropic resolution of 0.5 m and a Coriolis parameter of $f = 8 \times 10^{-5} \text{ s}^{-1}$. Turbulence in the free convection simulation is generated by buoyancy loss at the ocean surface, leading to static instability, and in the shear simulation by a

surface wind stress. Over time, convection- and wind-driven turbulence create a well-mixed layer at the top of the fluid, which deepens over time. Between the mixed layer and the stratified interior, a thermocline with sharp gradients is formed. This vertical structure can be seen in the horizontally averaged temperature and salinity fields in the third and fourth columns of Figure 3 for different times.

The two simulations illustrate the fundamental differences between the two scenarios of BL turbulence. In free convection, the case shown in the upper row of Figure 3, coherent plumes with strong downward velocities plunge from the surface toward and through the thermocline. As the plumes reach the base of the BL, they overshoot into the stratified ocean interior due to inertia and entrain denser fluid from the thermocline into the mixed layer, increasing the mixed layer's mean density. Entrainment mixing is nonlocal as it occurs at the BL base, driven by surface air-sea fluxes. However, in the shear-driven case in the bottom row of Figure 3, surface wind stress creates shear instability, which generates local eddies that decay away from the surface. The turbulent eddies generated by wind stress are much smaller in scale than the convective plumes created by buoyancy loss. Thus, from a parameterization perspective, the shear instability can be regarded as a local process. We will therefore develop NORi, starting with a diffusive closure that is known to represent well the local mixing induced by both convective and shear turbulence. To address the nonlocal entrainment that has proven very challenging to parameterize, we will add a machine-learned neural network component to NORi.

2.2. Training and Validation Data Generation

An LES training and validation suite is run to calibrate NORi. All LES are initialized with uniform temperature and salinity profiles in the horizontal direction and with a linear gradient in the vertical direction. The simulations are run under a variety of surface forcing scenarios: winds only, cooling or evaporation only, cooling + evaporation, winds + cooling and/or evaporation, and winds + heating + precipitation. All of these forcings produce destabilizing fluxes that mix away the ocean stratification and result in a deepening mixed layer in the upper ocean.

A convection-resolving simulation requires a resolution of $\mathcal{O}(1\text{ cm})$ to capture the homogenization of turbulent filaments at molecular scales. LES allow us to simulate the same process at a resolution of $\mathcal{O}(1\text{ m})$. Through experimentation, we found that our LES converged at an isotropic resolution of 2 m in domains of $(L_x, L_y, L_z) = (512, 512, 256)\text{ m}$. On an NVIDIA V100 GPU we achieved about 12 simulated days per day. This is the setup used for all LES in the NORi training suite.

The initial temperature and salinity stratifications are based on linear fits to summertime profiles from the ISAS reanalysis product of Argo profiles from 2002 to 2020 (Gaillard et al., 2016) spanning a representative range of oceanic stratification and equation of state regimes—we initialize with summertime profiles because we are interested in simulating the deepening of the mixed layer in fall and winter.

By referencing Argo profiles, we ensure that we cover a representative and realistic range of oceanic stratifications that we expect at inference time, including regimes with nonlinear thermodynamics. To that end, we consider profiles from three latitude bands: midlatitude Atlantic (strong temperature and salinity gradient), equatorial Pacific (strong temperature but weak salinity gradient), and Southern Ocean (salinity-dominated stratification, nonlinear thermodynamics). We complement these observationally inspired initial profiles with additional ones designed to span a broader parameter space, which is key to ensure generalizability and avoid overfitting. In particular, some training cases lack temperature or salinity gradients, others have inverse (stabilizing) salinity stratification, and others operate in the more thermodynamically nonlinear regime at lower temperatures. The actual values of the physical parameters used in the LES suite can be found in Appendix C.

The prescribed air-sea fluxes are constant in time and space and chosen to cover the range of forcing regimes encountered in the ocean from free convection (no wind stress) to pure wind scenarios (no buoyancy flux) to mixed conditions. While the training is based on simulations forced with constant fluxes, we demonstrate in Section 6.3 that NORi remains accurate even under time-varying fluxes. The range of flux values is chosen to be broad so that NORi is as unlikely as possible to run outside the training distribution at inference time. In terms of magnitudes, cooling rates range from 0 to $2,500\text{ W m}^{-2}$, evaporation rates range from 0 to around 45 m yr^{-1} , and wind stresses range from 0 to 0.5 N m^{-2} . For context, the mean monthly cooling rate of the Gulf Stream can reach 600 W m^{-2} (The Climode Group et al., 2009), latent heat fluxes during strong wind events such as tropical cyclones can reach 500 W m^{-2} (Vreugdenhil & Gayen, 2021), the winter average evaporation rate over the Gulf

Stream is around 3 m yr^{-1} (Yu, 2007), and typical monthly averaged wind stress is around 0.2 N m^{-2} (Copernicus Climate Change Service, 2019). The main training suite consists of 62 simulations, and the LES parameters can be found in Tables C1–C4. The parameters for the 32 simulations used in the validation and testing suite are given in Tables C1 and C5. Since heating and precipitation are stabilizing fluxes that do not contribute to the deepening of the BL, we did not include them in the training suite. However, we tested NORi's performance under a winds + heating + precipitation scenario to ascertain whether it is able to capture mixing accurately under stabilizing fluxes, which we have shown in the fourth column in the bottom panel of Figure 4 (the parameters for this case are shown in row seven of Table C1). This suite of data sets is publicly available for easy download and access on Zenodo (Lee, 2025a).

All LES are run for 2 days, a compromise between simulating long enough to capture the physics of turbulent mixing under different scenarios and keeping the memory footprint of the a posteriori calibration of the 1D column model manageable. The backward propagation of gradients requires storing intermediate model states to compute parameter sensitivities over the entire simulation period. During the forward–backward AD, we do not employ checkpointing; instead, we store the model state at every time step rather than only at a limited number of checkpoints. Introducing checkpointing would reduce the memory footprint but increase computational cost, because model states at specific times would need to be recomputed from their nearest checkpoint during backpropagation.

3. Modeling Unresolved Fluxes in a Coarse-Grained Column Model

Global oceanic simulations cannot be run at LES resolutions of $\mathcal{O}(1 \text{ m})$. Our goal is thus to develop a parameterization that predicts the deepening of the BL in response to forcing without the need to resolve the small-scale horizontal motions. In practice, we want to find a closed-form 1D equation that predicts the evolution of the LES area-averaged profiles of velocity, temperature, and salinity as a function of initial conditions and air-sea fluxes.

Let us start by defining the area averaging operator of some LES variable ϕ in a doubly periodic horizontal domain Ω ,

$$\phi(x, y, z, t) = \bar{\phi}(z, t) + \phi'(x, y, z, t), \quad (8)$$

$$\bar{\phi} \equiv \frac{1}{L_x L_y} \iint_{\Omega} \phi dx dy, \quad (9)$$

$$\bar{\phi'} \equiv 0, \quad (10)$$

where $\bar{\phi}$ is the horizontal area-average of ϕ , L_x and L_y are the dimensions of the lateral domain, and ϕ' is its deviation from the horizontal average. Applying this Reynolds-like averaging as well as the no-penetration boundary conditions at the top and bottom of the LES domain according to Equation 7, the momentum, temperature, and salinity Equations 1–3 reduce to a set of 1D equations (see complete derivation in Appendix A),

$$\frac{\partial \bar{u}}{\partial t} = -\frac{\partial}{\partial z} \overline{u'w'} + f\bar{v}, \quad (11)$$

$$\frac{\partial \bar{v}}{\partial t} = -\frac{\partial}{\partial z} \overline{v'w'} - f\bar{u}, \quad (12)$$

$$\frac{\partial \bar{T}}{\partial t} = -\frac{\partial}{\partial z} \overline{w'T'}, \quad (13)$$

$$\frac{\partial \bar{S}}{\partial t} = -\frac{\partial}{\partial z} \overline{w'S'}, \quad (14)$$

where $\overline{u'w'}$, $\overline{v'w'}$, $\overline{w'T'}$, and $\overline{w'S'}$ are the total unresolved momentum, temperature, and salinity fluxes due to subgrid-scale turbulence. Their surface boundary conditions are given by

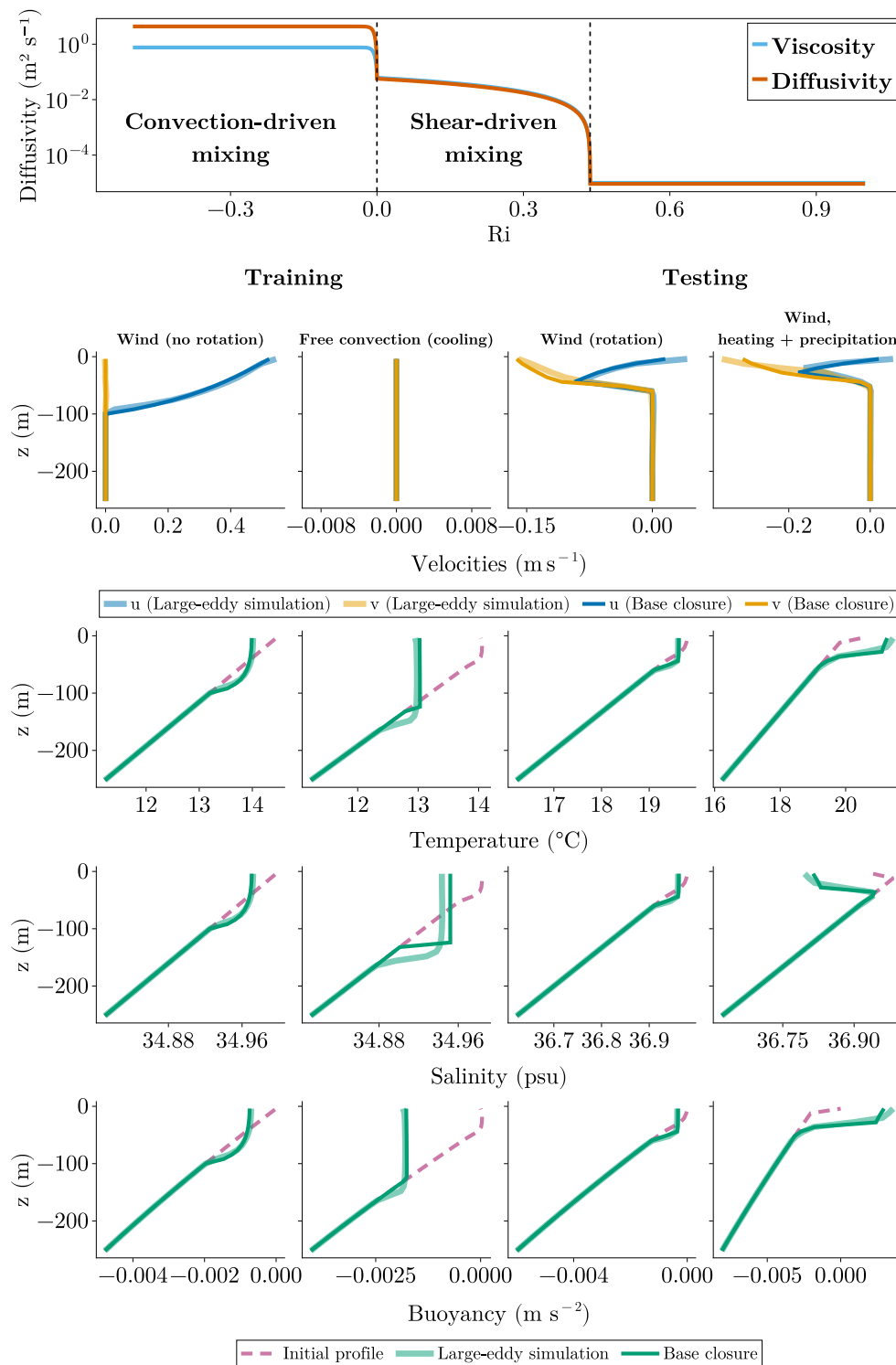


Figure 4. The top panel shows the calibrated diffusivity and viscosity values of the base closure as a function of the local gradient Richardson number Ri . In the convective range, the viscosity and diffusivity are constant, while they decrease linearly (note the logarithmic scale on the y-axis) to a background value across the shear range, with a continuous transition between regimes. The lower panels show vertical profiles of momentum, temperature, salinity, and buoyancy. Solid colored lines with a lighter shade are large-eddy simulation results forced with four different air-sea fluxes; the corresponding predictions from the base closure are overlaid with a darker shade, while the dashed lines represent the initial profiles. From left to right: a wind-forced example with no rotation and a free convection example with rotation used to train the base closure; a wind-forced example with rotation and a wind-forced example with heating and precipitation used for testing.

$$\overline{u'w'}(z=0) = J_u^{\text{top}}, \quad (15)$$

$$\overline{v'w'}(z=0) = J_v^{\text{top}}, \quad (16)$$

$$\overline{w'T'}(z=0) = J_T^{\text{top}}, \quad (17)$$

$$\overline{w'S'}(z=0) = J_S^{\text{top}}. \quad (18)$$

At the bottom $z = -L_z$ we impose free-slip boundary conditions on momentum and a gradient condition on temperature and salinity matched to the background stratification,

$$\overline{u'w'}(z = -L_z) = \overline{v'w'}(z = -L_z) = 0, \quad (19)$$

$$\partial_z \overline{T}|_{z=-L_z} = \frac{\partial T_0}{\partial z}, \quad (20)$$

$$\partial_z \overline{S}|_{z=-L_z} = \frac{\partial S_0}{\partial z}, \quad (21)$$

so that the initial mean gradients $\frac{\partial T_0}{\partial z}$ and $\frac{\partial S_0}{\partial z}$ are preserved at the lower boundary. We write equations for temperature and salinity separately because we use the full nonlinear equation of state of seawater, and we cannot combine the two equations into a single one for buoyancy.

4. “Ri”: Eddy-Diffusivity Closure Based on Local Gradient Richardson Number

Following Kochkov et al. (2024), we first attempted to develop a parameterization of the BL using a purely data-driven approach by modeling the full *fluxes* of $\overline{u'w'}$, $\overline{v'w'}$, $\overline{w'T'}$, and $\overline{w'S'}$ with neural networks (Note that Kochkov et al. (2024) modeled the *tendencies* instead of the fluxes). However, we found that neural networks were not able to learn that vertical profiles of temperature and salinity must be statically stable in simulations that parameterize rather than resolve turbulence, that is, $N^2 = \frac{\partial b}{\partial z} \geq 0$ where N^2 is the buoyancy frequency. Instead, neural networks produced a BL that was nearly well-mixed near the surface, but with small, random fluctuations in z resulting in negative N^2 patches. In nonhydrostatic formulations, this stratification inversion would trigger convective instabilities that would rapidly remove the negative N^2 . However, large-scale ocean models use the hydrostatic approximation to lower computational costs, which is inconsistent with the physics of density overturns. This deficiency could likely be addressed by increasing the size of the training data or the expressivity of the neural networks (Subel et al., 2021), but both incur significant computational and memory costs given the serial-in-time nature of a posteriori calibration.

In view of these challenges, we instead start by formulating a simple physics-based parameterization that captures most of the BL physics and then augment it with a neural network. We will refer to this physics-based component as the *base closure*. Training a neural network to learn the physics missing in the base closure rather than all BL physics reduces the data requirement and model size, dramatically improves parameterization performance, and reduces numerical instability during training and validation. Mathematically, the base closure represents all subgrid-scale fluxes as downgradient fluxes of momentum, temperature, and salinity,

$$\overline{u'w'} = J_{u,\text{local}} = -\nu \frac{\partial \overline{u}}{\partial z}, \quad (22)$$

$$\overline{v'w'} = J_{v,\text{local}} = -\nu \frac{\partial \overline{v}}{\partial z}, \quad (23)$$

$$\overline{w'T'} = J_{T,\text{local}} = -\kappa \frac{\partial \overline{T}}{\partial z}, \quad (24)$$

$$\overline{w'S'} = J_{S,\text{local}} = -\kappa \frac{\partial \bar{S}}{\partial z}, \quad (25)$$

where ν and κ are the (positive) eddy viscosity and diffusivity, which depend on the local gradient Richardson number Ri , given by

$$Ri = -\frac{g}{\sigma_0} \frac{\frac{\partial \bar{\sigma}}{\partial z}}{\left(\frac{\partial \bar{u}}{\partial z}\right)^2 + \left(\frac{\partial \bar{v}}{\partial z}\right)^2}. \quad (26)$$

The Ri -dependence is well supported by theory (Drazin & Reid, 2004) and ocean observations (e.g., Price et al., 1986), and it forms the basis of many ocean BL parameterizations (e.g., Large & Gent, 1999; Pacanowski & Philander, 1981; Wagner, Hillier, et al., 2025). When $Ri < 0$, the BL stratification is convectively unstable. When the BL is statically stable, but the shear is large enough so that Ri is below some critical value Ri^c , the BL is shear unstable. The specific functional dependence of diffusivity and viscosity on Ri is illustrated in the upper row of Figure 4 and takes the form,

$$\nu = \begin{cases} (\nu_{\text{shear}} - \nu_{\text{conv}}) \tanh \frac{Ri}{\Delta Ri} + \nu_{\text{shear}}, & \text{for } Ri < 0 \\ (\nu_0 - \nu_{\text{shear}}) \frac{Ri}{Ri^c} + \nu_{\text{shear}}, & \text{for } 0 \leq Ri < Ri^c \\ \nu_0, & \text{for } Ri \geq Ri^c \end{cases} \quad (27)$$

$$\kappa = \begin{cases} (\kappa_{\text{shear}} - \kappa_{\text{conv}}) \tanh \frac{Ri}{\Delta Ri} + \kappa_{\text{shear}}, & \text{for } Ri < 0 \\ (\kappa_0 - \kappa_{\text{shear}}) \frac{Ri}{Ri^c} + \kappa_{\text{shear}}, & \text{for } 0 \leq Ri < Ri^c \\ \kappa_0, & \text{for } Ri \geq Ri^c \end{cases} \quad (28)$$

$$\kappa_{\text{shear}} = \frac{\nu_{\text{shear}}}{Pr_{\text{shear}}}, \quad (29)$$

$$\kappa_{\text{conv}} = \frac{\nu_{\text{conv}}}{Pr_{\text{conv}}}, \quad (30)$$

where ν_{conv} , ν_{shear} , Ri^c , ΔRi , Pr_{conv} , and Pr_{shear} are parameters that are trained with the LES suite. ν_{conv} and ν_{shear} represent the viscosity under convective and shear turbulence, while Pr_{conv} and Pr_{shear} are their respective turbulent Prandtl numbers. Ri^c is the critical Richardson number above which the flow is stable, and ΔRi sets the width of the transition between the convective and shear regimes. The other parameters are instead fixed: $\sigma_0 = 1,020 \text{ kg m}^{-3}$ is the reference potential density, $g = 9.80665 \text{ m s}^{-2}$ is the gravitational acceleration, $\nu_0 = 1 \times 10^{-5} \text{ m}^2 \text{ s}^{-1}$ is the background viscosity whose value is chosen to be small enough to have no impact on the training and validation tests, while κ_0 is calculated using the shear Prandtl number $\kappa_0 = \frac{\nu_0}{Pr_{\text{shear}}}$.

Physically, when Ri drops below its critical value Ri^c but remains positive (i.e., $0 \leq Ri < Ri^c$), the flow becomes shear unstable (but convectively stable), and the viscosity and diffusivity increase from their background values ν_0 and κ_0 linearly to their shear values ν_{shear} and κ_{shear} as Ri decreases from Ri^c to 0. When Ri drops below zero the flow becomes convectively unstable and the viscosity and diffusivity increase to their convective values ν_{conv} and κ_{conv} over a transition width set by ΔRi .

This formulation is inspired by the Pacanowski–Philander model (Pacanowski & Philander, 1981), but differs from it by using different functional forms and accounting for different mixing rates in the convective and shear regimes. Regardless, a purely Ri -based parameterization cannot fully capture the entire range of small-scale mixing in ocean BLs, as it does not account for nonlocal dynamics such as entrainment (Zaron & Moum, 2009). Indeed, the Pacanowski–Philander model is not used in modern coarse-resolution ocean models.

However, in NORi the base closure is only the first step and nonlocal mixing processes will be captured with neural network components.

The base closure implies a change in both the magnitude of viscosity and the Prandtl numbers between the convective, shear, and stable regimes. This is supported by the LES solution and mixing length arguments. The eddy diffusivity/viscosity can be expressed as the product of a turbulent velocity and a mixing length scale. Although the turbulent velocity spans similar ranges in all regimes, the mixing length scale is significantly longer in the convective regime, spanning the entire turbulent BL and resulting in stronger mixing of tracers. In the convective regime, the turbulent Prandtl number is smaller than one because rotation slows momentum mixing by establishing a thermal wind when the vertical mixing time scale is longer than the Coriolis period (Young, 1994). This is not relevant in the shear regime, which is associated with faster mixing time scales.

The free parameters in the base closure are calibrated in an a posteriori fashion by minimizing the mean squared difference between the column profiles and vertical gradients predicted by the closure and those diagnosed from the LES, for all scenarios and times. The loss function $\mathcal{L}_{\text{base}}$ is defined as

$$\mathcal{L}_{\text{base}}(\bar{u}, \bar{v}, \bar{T}, \bar{S}; \theta_{\text{base}}) = \frac{1}{N_{\text{sim}}} \sum_{a=1}^{N_{\text{sim}}} (\mathcal{L}_u^a + \mathcal{L}_v^a + \mathcal{L}_T^a + \mathcal{L}_S^a + \mathcal{L}_\sigma^a) \quad (31)$$

with

$$\mathcal{L}_\phi^a = \frac{A_\phi^a}{N_t N_z} \sum_{i=1}^{N_z} \sum_{j=1}^{N_t} |\bar{\phi}^{a,i,j} - \bar{\phi}_{\text{LES}}^{a,i,j}|^2 + \frac{A_{\partial_z \phi}^a}{N_t N_z} \sum_{i=1}^{N_z} \sum_{j=1}^{N_t} |\partial_z \bar{\phi}^{a,i,j} - \partial_z \bar{\phi}_{\text{LES}}^{a,i,j}|^2, \quad (32)$$

where $\bar{\phi}$ and $\bar{\phi}_{\text{LES}}$ are the profiles of one of the variables (velocity, temperature, salinity) as a function of vertical level i , time j , and scenario a —the overbars indicate that we are only parameterizing the area-averaged profiles. Normalization factors A_ϕ^a are adaptively calculated based on initial conditions and losses in iteration 0 before training to ensure balanced learning in all variables and loss components. This is done through a two-stage process. First, the magnitudes of the temperature and salinity losses are reweighted to be inversely proportional to their contributions to density variations, preventing either variable from dominating the loss. The density loss function is then weighted such that

$$\frac{\mathcal{L}_\sigma^a}{\mathcal{L}_T^a + \mathcal{L}_S^a} = \frac{1}{9}. \quad (33)$$

The 1/9 ratio is chosen to ensure that the density contribution to the loss function acts as a weak constraint compared to temperature and salinity ones. This lets density guide the optimization process without overwhelming the temperature and salinity signals. The value of 1/9 is chosen through experimentation, and the calibration is insensitive to it as long as the density weighting is small enough for the density contribution to remain a weak constraint relative to the temperature and salinity losses. Momentum losses are normalized to match the sum of density, temperature, and salinity losses. The same weights are also applied to their respective gradient loss counterparts. Second, after these physical weightings are applied, the total profile losses and gradient losses are rescaled to be equal. The mathematical details of how normalization factors are computed are provided in Appendix D.

The free parameters $\theta_{\text{base}} = [\nu_{\text{conv}}, \nu_{\text{shear}}, Ri^c, \Delta Ri, Pr_{\text{conv}}, Pr_{\text{shear}}]^T$ in the base closure are calibrated with EKI using the EnsembleKalmanProcesses.jl package (Dunbar et al., 2022; Iglesias et al., 2013), a gradient-free optimization algorithm well-suited for problems with a small number of parameters. Since there are only six tunable parameters in the base closure, we can afford a large ensemble (200 members) to explore the parameter space more effectively than gradient-based methods and take advantage of EKI's robust performance when dealing with noisy loss landscapes (Gjini et al., 2025).

Free parameters in the base closure are calibrated in two stages. First, we calibrate only the shear-relevant parameters ν_{shear} , Ri^c , and Pr_{shear} using LES that are purely wind-driven as shown in Table C1, freezing the convection-relevant parameters that are never activated in these scenarios since $Ri \geq 0$. Second, we freeze the

shear-relevant parameters and calibrate the convection-relevant parameters ν_{conv} , ΔRi , and Pr_{conv} using LES that contain convective forcing as shown in Tables C2–C4. Even though the base closure is unable to capture nonlocal entrainment, free convection cases are still useful to calibrate the convection-relevant parameters which quantify the strength of local convective mixing. With this two-stage approach, we aim to minimize error compensation during the calibration, ensuring that we are capturing the physics in a structurally and parametrically correct way. The calibrated parameters of the base closure are $\nu_{\text{conv}} = 0.761 \text{ m}^2 \text{ s}^{-1}$, $\nu_{\text{shear}} = 6.16 \times 10^{-2} \text{ m}^2 \text{ s}^{-1}$, $Ri^c = 0.437$, $\Delta Ri = 9.70 \times 10^{-3}$, $Pr_{\text{conv}} = 0.175$, and $Pr_{\text{shear}} = 1.08$.

The base closure is capable of capturing the evolution of the vertical profile of the BLs under training and testing scenarios where instabilities are driven purely by surface wind stresses. As expected from a purely local parameterization, the base closure underestimates the mixed layer depth when convection is present, since entrainment is a nonlocal process (as discussed in Section 2). The bottom panels of Figure 4 show examples of vertical profiles of temperature, salinity, and potential density from LES forced with four different air-sea fluxes and the corresponding predictions from the base closure. The base closure is unable to fully capture the deepening of the BL in the free convection training case (second from left), which could be attributed to the missing nonlocal entrainment physics. The other three training and validation cases shown in Figure 4 include purely wind-driven mixing in rotational and irrotational cases, as well as (destabilizing) wind stress combined with (stabilizing) heating and precipitation. In all three cases, mixing is purely local and the base closure is able to accurately predict the depth of the mixed layer. Compared with traditional physical closures, the base closure is comparable with $k - \epsilon$ (first column of Figure 7) and CATKE (first column of Figure J1). Since the base closure performs comparably to CATKE, which has been shown to outperform KPP (Wagner, Hillier, et al., 2025), the base closure also outperforms KPP in scenarios without convection.

5. “NO”: Using Neural Networks to Capture Nonlocal Entrainment

We now use neural networks as a residual model to capture the nonlocal entrainment physics missing from the base closure.

5.1. Enforcing Physical Knowledge and Constraints on Neural Networks With Architecture Design

In NORi, the neural networks are used to predict the missing entrainment fluxes of temperature and salinity at the base of the BL. Initially, we included additional networks for u and v , but they did not improve skill despite a substantial increase in training cost. This is likely because the momentum fluxes are dominated by inertia-gravity wave signals rather than turbulent fluctuations, obscuring the entrainment signal that contributes to vertical mixing. Through our experimentation, we also found that entrainment fluxes in T and S still lead to enhanced momentum mixing at the base of the BL, because the entrainment-modified stratification lowers Ri , which raises the eddy viscosity of the base closure.

Mathematically, we add nonlocal terms J_{NN_T} and J_{NN_S} which are computed using neural networks NN_T and NN_S to Equations 24 and 25,

$$\overline{w'T'} = J_{T,\text{local}} + J_{T,\text{nonlocal}} = -\kappa \frac{\partial \overline{T}}{\partial z} + J_{\text{NN}_T} \quad (34)$$

$$\overline{w'S'} = J_{S,\text{local}} + J_{S,\text{nonlocal}} = -\kappa \frac{\partial \overline{S}}{\partial z} + J_{\text{NN}_S} \quad (35)$$

By construction, the surface and bottom fluxes of J_{NN_T} and J_{NN_S} are set to zero, ensuring that the neural network only redistributes tracers within the water column without adding or removing them from the domain. The output of each neural network NN_T and NN_S consists of only a scalar value that represents the nonlocal flux at the target grid point. To infer the missing entrainment fluxes along the column, the same neural networks are “convolved” along the vertical dimension, that is, the weights of the neural networks are shared among all vertical grid points. The inputs to NN_T and NN_S are the temperature gradient, salinity gradient, potential density gradient, and gradient Richardson number of five grid points within the neighborhood of the grid point intended for inference, as well as the surface buoyancy flux. The information of local shear is encoded in the local Richardson number Ri while information regarding nonlinear seawater thermodynamics from TEOS-10 (Roquet et al., 2015) is encoded

in the potential density gradient inputs. Location-dependent inputs (neighborhood tracer gradients and Ri) change accordingly as the neural networks are vertically “convolved.” This design makes the effect of entrainment local to the BL base, so only local information about the flow is required to predict the effects of entrainment-driven mixing. However, nonlocal information of the surface buoyancy flux is provided as inputs to predict the strength of the convective plumes, consistent with evidence that the entrainment fluxes scale nonlocally with air-sea fluxes (Deardorff et al., 1980; Van Roekel et al., 2018). More technical details on the neural network architecture are provided in Appendix E.

Under this formulation, the neural networks' behavior is independent of depth, implicitly assuming that entrainment depends only on the local flow structure and the surface buoyancy flux, not on the absolute depth of the BL base. This assumption is valid for BLs with depths up to roughly a kilometer. In very deep BLs, rotational effects partly suppress entrainment as the time scale for convective plumes to reach the BL base becomes comparable to the Coriolis time scale (Marshall & Schott, 1999). Such deep BLs are only found in regions of dense water formation at high latitudes and require a dedicated parameterization that includes both vertical mixing by convection under the influence of rotation and lateral restratification by lateral baroclinic instabilities (Boccaletti et al., 2007). NORi, like most ocean BL parameterization schemes, does not explicitly account for deep convection physics.

Taking advantage of the fact that entrainment creates upgradient fluxes near the base of the BL, the neural networks are active for inference within five grid points below and 10 grid points above the base of the BL, diagnosed as the shallowest point at which the gradient Richardson number exceeds the critical value Ri^c . Five grid points below the BL base, equivalent to 40 m, proved sufficient to capture plume-driven entrainment in all LES spanning the wide range of forcings considered. While the upgradient fluxes are located near the base of the BL, they bring tracers into the mixed layer where they are rapidly redistributed across the entire BL by the BL-filling convective plumes. Ten grid points above the BL base proved necessary to keep tracers well mixed over the BL. In effect, the neural networks have implicitly learned to detect the base of the BL through the Richardson number criterion and to apply entrainment fluxes in its vicinity. We experimented with using fewer and more grid points above and below the base of the BL and found that this configuration gave us the best results based on the LES we have run for the training suite.

5.2. Neural ODEs: Online Calibration to Promote Accuracy, Generalizability, and Stability

The 1D column model consists of a system of ODEs for two velocity components, temperature, and salinity, with neural networks embedded in the temperature and salinity equations. Two separate neural networks that represent the nonlocal turbulent fluxes of temperature and salinity are embedded in their respective equations. The neural ODEs are calibrated a posteriori to promote numerical stability and generalizability of the solutions, as explained in the introduction. This means that the loss $\mathcal{L}$ is a function of the full time history of the evolving variables rather than the subgrid fluxes. Mathematically, for some integration time from t_0 to t_1 where the variables are time-marched forward,

$$\mathcal{L}(\bar{u}, \bar{v}, \bar{T}, \bar{S}; \theta_T, \theta_S) = \mathcal{L}(\bar{u}(t), \bar{v}(t), \bar{T}(t), \bar{S}(t), \bar{\sigma}(t)), \quad (36)$$

where θ_T and θ_S are the trainable weights of the neural networks, and the variables in $\mathcal{L}$ are the full time series from t_0 to t_1 . The potential density σ is included in the loss because it is a nontrivial function of temperature and salinity due to nonlinearities in the equation of state.

The neural networks are trained with a posteriori calibration. We utilize AD to compute how loss changes with respect to the weights of the neural networks. AD propagates derivatives through the ODE solver, enabling gradient-based updates to parameters. In the context of the parameterization problem addressed in this study, the climate model acts as the forward model, implying that online calibration requires the climate model to be entirely differentiable. However, contemporary climate models are not differentiable. Additionally, due to the serial-in-time nature of online calibration, training requires substantial computational power and memory. Below, we describe how we were able to overcome both challenges. First, however, we explain why we did not resort to a priori calibration, which is a common strategy to simplify the calibration task.

A priori (or offline) calibration trains the neural networks to match the turbulent fluxes diagnosed from the LES as a function of the inputs with no information about the temporal evolution of the system. Mathematically, offline training for our problem would take the form

$$\mathcal{L}_{\text{offline}}(\bar{u}, \bar{v}, \bar{T}, \bar{S}; \theta_T, \theta_S) = \mathcal{L}(\overline{u'w'}, \overline{v'w'}, \overline{w'T'}, \overline{w'S'}). \quad (37)$$

This approach led to numerical instabilities within a few time steps when applied to our parameterization problem. The reason is that the diagnosed horizontally averaged LES turbulent fluxes contain irrelevant signal, like internal waves, in addition to that relevant to entrainment: linear internal waves generate fluctuations in all variables but have no impact on entrainment, and discretization errors are unavoidable. Although regularization techniques can be used to reduce the impact of noise contamination in training data, this is not appropriate for our purposes because the target entrainment flux signal is characterized by sharp gradients at the BL base. Any smoothing or further coarse-graining of its vertical profile would affect the stratification at the base of the BL, which is a crucial variable in setting the deepening of the BL itself. As a result, a priori calibration results in parameterizations of fluxes with a small amount of grid-scale noise. This noise inevitably accumulates as the neural ODEs are integrated forward in time, eventually leading to a finite-time blowup. We speculate that this is due to the iterative rollout of neural ODEs and the lack of explicit variance-dissipation mechanisms in Equations 11 through 14. As the neural networks receive noisy inputs (the fluxes), they generate noisy outputs. Lacking strong damping, this noise accumulates, leading to blowup. However, this approach might succeed with substantially more training data, but at considerably greater cost.

The loss function used for a posteriori calibration of the neural networks has the same form as the one used to train the base closure and given in Equation 32 except for the omission of the momentum fields, $\mathcal{L}_u^a$ and $\mathcal{L}_v^a$. Including the momentum terms in the loss function significantly deteriorates the quality of the training, because the velocity fields are dominated by inertial oscillations, which have no impact on entrainment. In an attempt to capture them, neural networks deteriorate the quality of T and S predictions. However, flow shear information is implicitly encoded in the local Richardson number Ri , which is invariant to inertial oscillations and Galilean transformations. We also found that including vertical derivatives of tracers in the loss function during optimization promotes smooth solutions free of grid-scale oscillations about the training data, acting as a form of regularization. Training is carried out over $N_{\text{sim}} = 62$ simulations outlined in Tables C1–C4. The details of how normalization factors are calculated are discussed in Section 4 and Appendix D.

At the time of NORi's development, there were no fully differentiable ocean models that provided the functionality and interface required for our a posteriori calibration. Therefore, we built a standalone implementation of the 1D BL model described in Equations 11 through 14 in the Julia programming language (Bezanson et al., 2017). At the time of writing, this training is likely to become feasible directly in Oceananigans.jl in the near future, due to recent developments in its AD capabilities (Moses et al., 2026).

The neural ODEs Equations 11 through 14 are quite stiff (Kim et al., 2021) due to the multiscale nature of the BL physics, where the diffusion terms, Coriolis terms, and neural network entrainment terms evolve on three significantly different length and time scales. Explicit time stepping schemes are thus prohibitively expensive for both the forward and backward passes. Instead, we implemented a split implicit-explicit algorithm, where the diffusion term is time stepped with the implicit Euler method and the nonlocal flux is time stepped with the explicit Euler method. The neural ODEs are time stepped by iterating over the nondimensional, discretized equations with a constant $\Delta t_{\text{train}} = 10$ min during training and $\Delta t_{\text{validation}} = 5$ min during validation (The discretized equations are provided in Appendix F and the nondimensionalization is derived in Appendix G.). As we will also show in Section 6, NORi is insensitive to time step size up to 1 hr, so with a training time step of 10 min, explicit higher-order time stepping schemes are not required.

The neural networks are built employing Lux.jl (Pal, 2023), with the standard Glorot weight initialization technique (Glorot & Bengio, 2010). During training, both input and output data are normalized to a mean of zero and a variance of one across all training simulations. The backpropagation of the weight gradients of the neural network with respect to the loss function is accomplished through the Enzyme.jl AD tool (Moses & Churavy, 2020) in the Julia language; see Equations F1 through F4. The weights of the neural network are updated using the Adam optimizer (Kingma & Ba, 2017). Through a grid hyperparameter sweep over different activation functions, neural network layers, and the number of units in the hidden layers, we found that 3 hidden layers with

128 units each and rectified linear units (ReLU, Glorot et al., 2011) provided the best performance as computed following the loss function introduced above. Each neural network in the final configuration contains 35,969 parameters. To avoid local minima due to the complex loss landscape, we employ a “curriculum learning” strategy. This involves training neural ODEs to make accurate and numerically stable predictions over progressively extended integration time frames. Integration times are increased in three stages with integration windows of 15, 23.3, and 43.3 hr, with the learning rate decreased at each stage. These values carry no particular significance apart from corresponding to the 90th, 140th, and 260th time steps at a 10-min time step. In each stage, the neural ODEs are trained for 2,000 epochs, and the initial model weights of each stage are selected based on the lowest training loss in the previous stage. At the beginning of each stage, the losses are renormalized, that is, the A_ϕ^a factors in the loss function are recalculated to promote a good gradient flow toward optimal model performance. The final model weights are selected on the basis of the lowest validation loss in the final stage. We used a total of 62 training examples to train neural ODEs, which is equivalent to 62 different initial and boundary conditions integrated forward in each epoch. A “full-batch gradient descent” approach (where the loss is computed across all training simulations for each gradient descent step) is taken, as we found that performing minibatching worsens loss convergence. Total training takes approximately 90 hr wall-clock time on a single CPU core on the Supercloud computing cluster (Reuther et al., 2018). We chose CPU training because of Enzyme.jl's more mature CPU support at the time of training NORi, but in future iterations it would be useful to train natively on Oceananigans.jl with GPUs.

6. Training, Validating, and Comparing NORi in Single-Column Contexts

All single-column experiments in this section use a vertical grid resolution of 8 m, the grid resolution on which NORi is trained. We chose this resolution to match that of operational global ocean models, which is $\mathcal{O}(10\text{ m})$ in the entrainment region in the upper ocean.

6.1. Training and Validation Loss

We trained NORi on 62 cases listed in Tables C1–C4 and validated it on another 30 cases shown in Table C5 to select the final model weights. The last two testing cases in Table C1 are not used to validate the components of the neural network, as they lack the nonlocal convection-driven entrainment that the neural networks are designed to capture. Figure 5 shows that the neural network losses decreased during the training epochs, approaching convergence after about 4,000 epochs. Training losses remain small in cases that begin with small losses, indicating that neural networks augment the base closure only when there are significant residual errors. We also find that the “curriculum” learning approach, where neural networks are trained on progressively longer integration windows, reduces the loss over the final 43.3 hr more effectively than training directly on the full integration window. The bulk of loss reduction occurs in the first stage of training, which covers the first 15 hr, suggesting that once neural networks learn to capture entrainment physics early on, they tend to generalize reasonably well to the rest of the integration window. The longer integration windows in the second and third stages serve to fine-tune the model weights and enhance the numerical stability of the solutions over longer time frames. The validation losses follow a similar trend to the training losses, indicating that NORi interpolates well across the physical regimes it is trained on.

6.2. Inference Performance Against Large-Eddy Simulations

Once trained, NORi matches the LES solutions in all training cases and demonstrates strong predictive skill in the validation cases. This includes the cases where the base closure was inadequate, indicating that NORi effectively captures entrainment, which is the primary deficiency of the base closure.

We illustrate the improvement of NORi over the base closure in six representative cases: three training cases and three validation cases in Figure 6. NORi matches the LES solutions in the training cases and approximates the validation cases well, showing substantial improvement over the base closure in simulations where the entrainment is strong. The first column of Figure 6 shows the evolution of a BL subject to strong winds and cooling. The initial stratification is derived from a linear fit to summertime Argo profiles in the midlatitude Atlantic, where the gradient is strong (Gaillard et al., 2016). During the first hours, a turbulent Ekman layer driven

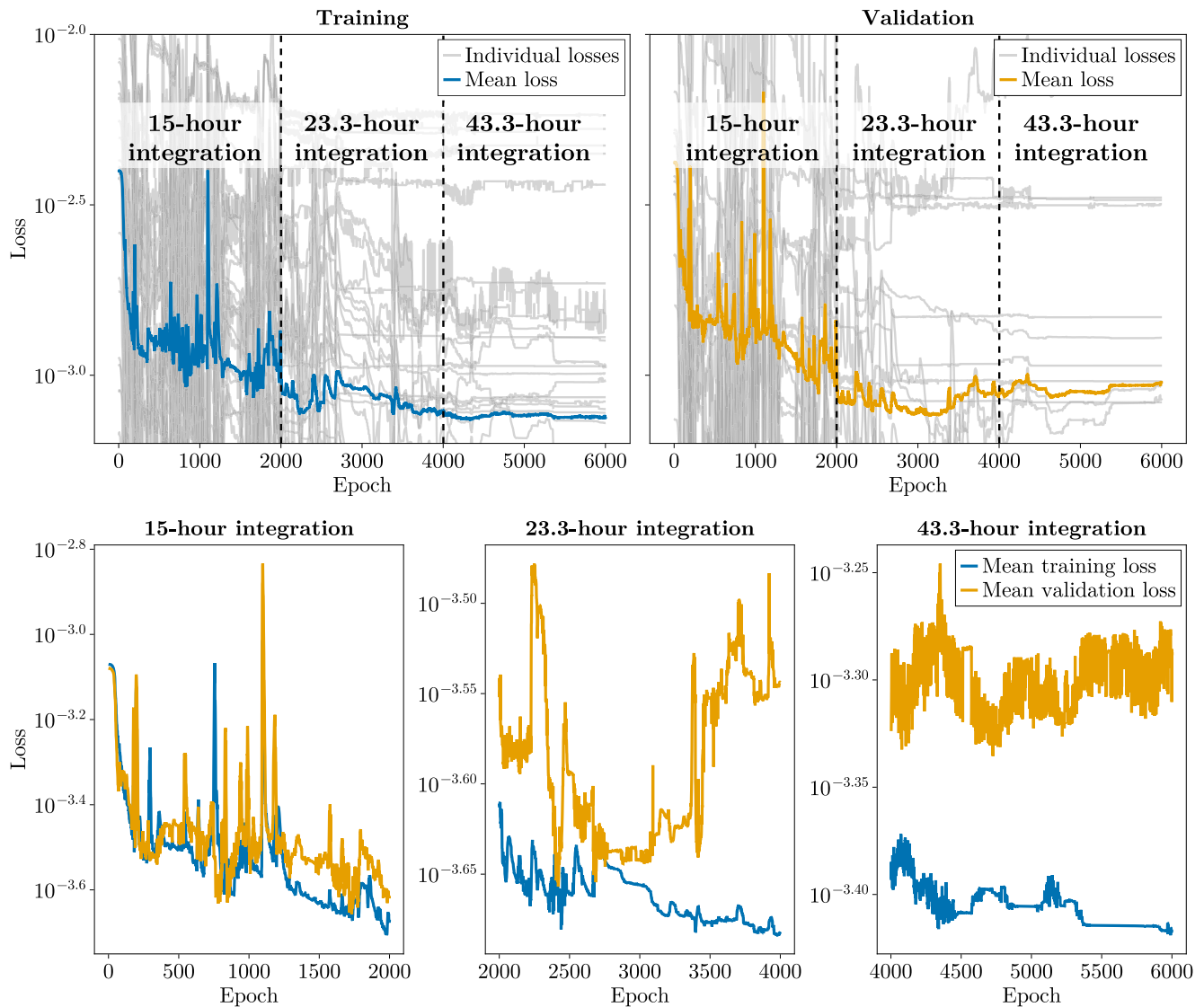


Figure 5. Training and validation losses of NORi. Top row: mean and individual losses as a function of epoch over the final integration horizon of 43.3 hr for the training suite (Tables C2–C4) in the left panel and the validation suite (Table C5) in the right panel. For the plots shown in this upper row, all losses are normalized once, at epoch 0—an epoch is defined as one iteration over the entire training suite. The gray lines are the individual losses for each training/validation case, while the colored lines are their means. Bottom row: mean training and validation losses over the three stages of “curriculum learning,” where each subsequent stage covers a longer integration period: 15, 23.3, and 43.3 hr. The training loss is re-normalized at the start of each stage, as explained in Appendix D. The epoch with the lowest training loss at each stage is used to initialize network weights for the next stage. The final model weights are selected from the model with the lowest validation loss in the final stage to avoid overfitting the model to the data.

by wind develops at the surface, with depth $L_{Ek} \approx 0.25f^{-1}\sqrt{|J_u^{top}|} \sim 70$ m (McWilliams, 2006). But soon afterward convective plumes start to penetrate deeper, resulting in substantial entrainment captured by NORi, but not by the base closure. In contrast, the third column of Figure 6 presents a simulation in which both NORi and the base closure match the LES solution. This solution is representative of the equatorial Pacific, where $f = 0$ and therefore $L_{Ek} \rightarrow \infty$, so Ekman dynamics do not apply. Despite strong evaporation, momentum shear reaches the base of the BL and dominates over convection, resulting in minimal entrainment. These two examples demonstrate that NORi correctly reduces to the base closure when shear dominates turbulence and deepening scales with Ri , while it improves on the base closure when convection drives entrainment.

The fourth through sixth columns of Figure 6 show validation examples that were not seen during training. NORi generalizes well to these unseen scenarios, confirming that the neural network effectively interpolates within the

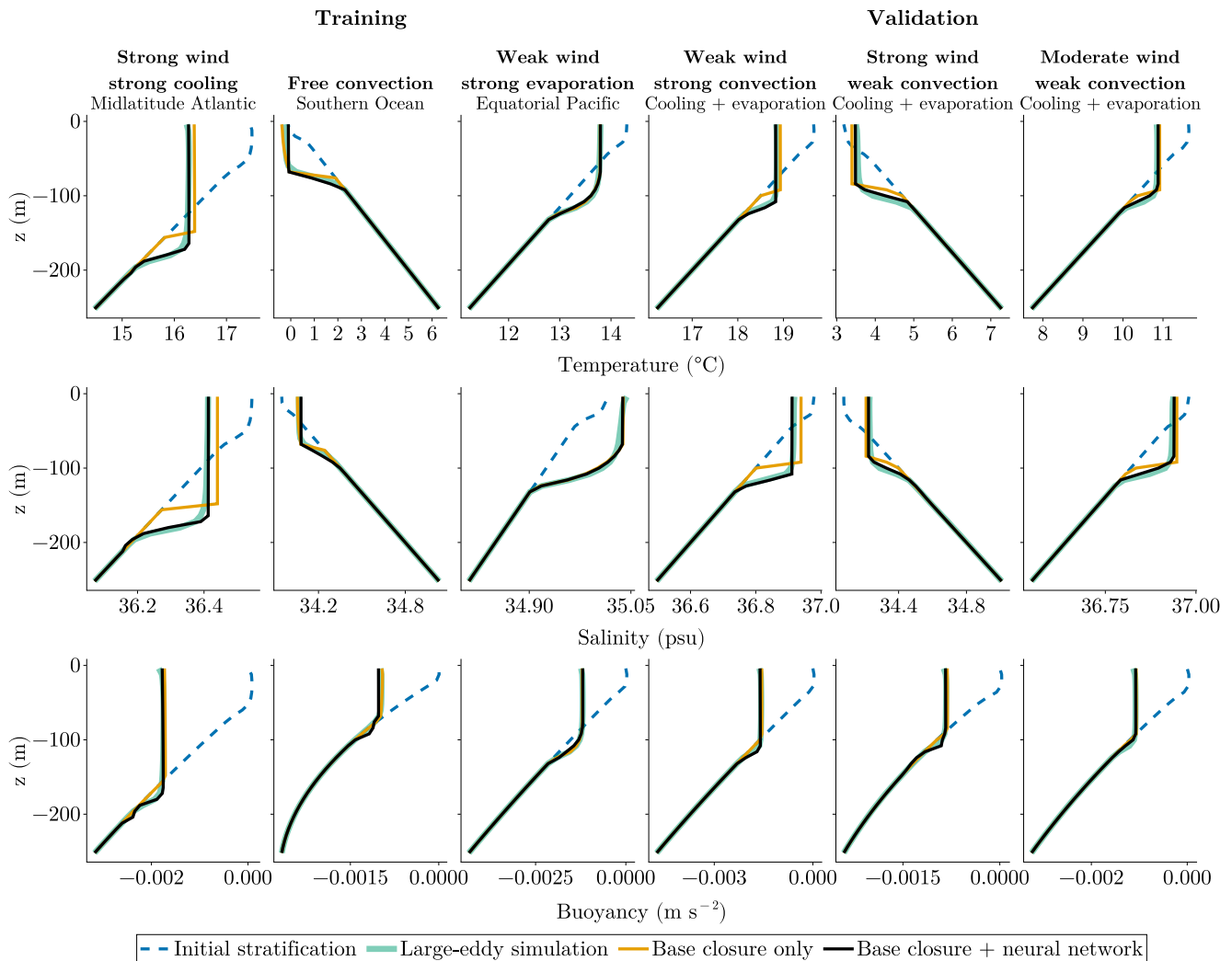


Figure 6. Temperature, salinity, and buoyancy profiles for selected training (columns 1 through 3) and validation (columns 4 through 6) examples generated with NORi (black lines), the base closure (orange lines), and area-averaged large-eddy simulation solutions (green lines). The profiles are computed 1.75 days after initialization (dashed lines).

range of air-sea fluxes and stratification values seen in training. Should new cases emerge where NORi fails to match LES or observational measurements of BL evolution, its high expressivity and flexibility give us confidence that it can be retrained through fine-tuning or transfer learning to capture the new physics.

Importantly, NORi works well in the nonlinear thermodynamics regime, as demonstrated in the second and fifth columns of Figure 6, where conditions similar to those of the Southern Ocean with low temperatures, destabilizing temperature stratifications, and salinity-dominated buoyancy structures are illustrated. The initial temperature and salinity stratifications are linear with depth, but the resulting buoyancy is not. NORi captures the entrainment of temperature, salinity, and buoyancy, which behave differently due to the nonlinear equation of state and are missed by the base closure.

It is worth remarking that the base closure matches the buoyancy LES profiles much better than those of temperature and salinity, whose errors compensate when combined into buoyancy. This supports our argument that BL parameterizations must be validated with solutions and observations of both temperature and salinity profiles, rather than just buoyancy, as is often done in the literature.

Figure 7 shows that NORi compares very favorably with $k - \epsilon$, a state-of-the-art two-equation first-order turbulence closure that solves two additional prognostic equations for the TKE k and its dissipation ϵ to predict the

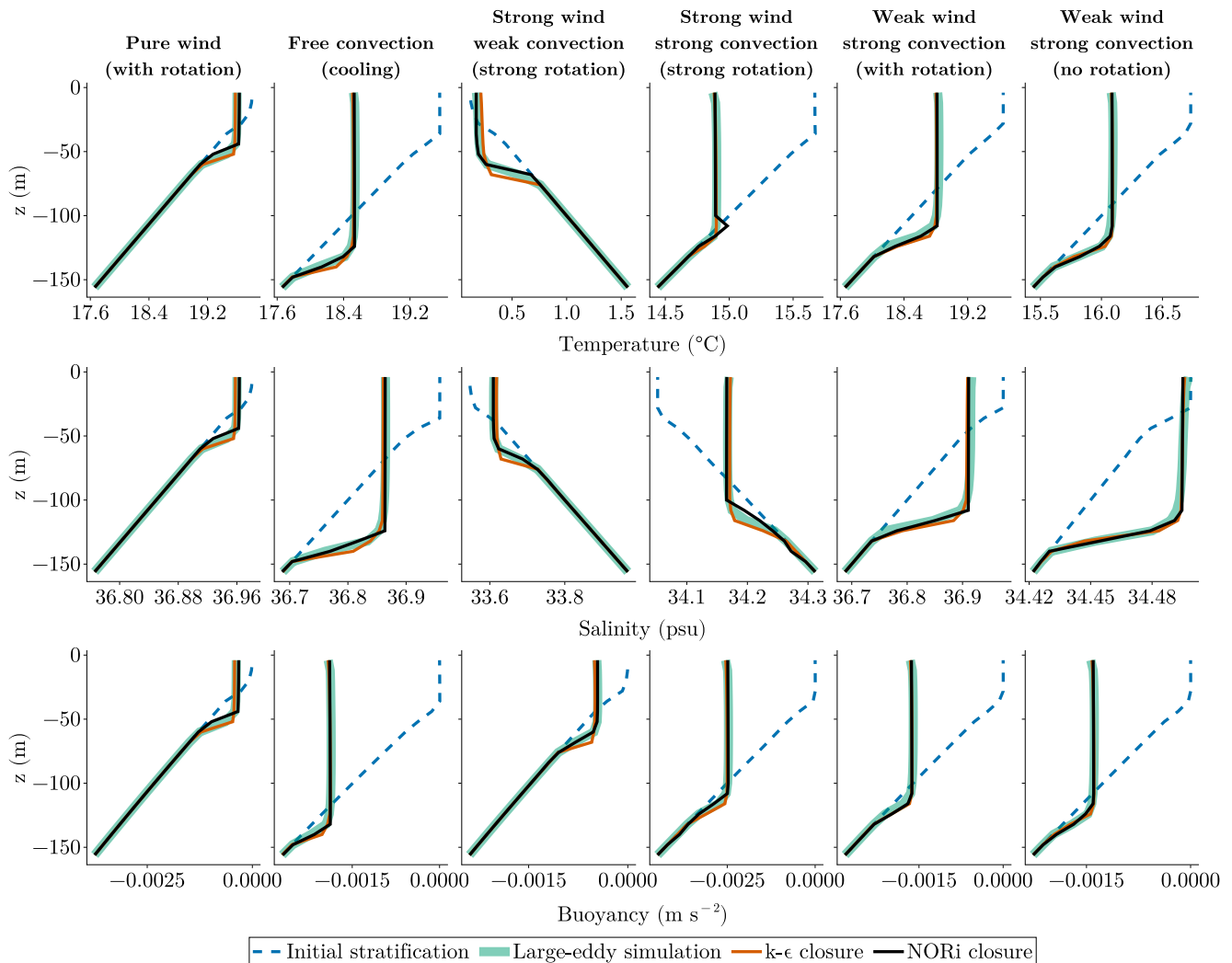


Figure 7. Comparison of NORi and $k - \epsilon$ performance in a column-model setting. The model profiles of temperature, salinity, and buoyancy are plotted together with the area-averaged large-eddy simulation profiles. The examples shown are drawn from validation cases not seen by NORi during training (see Table C5 for a complete list). The profiles are computed 1.75 days after initialization (dashed lines).

turbulent fluxes $\overline{u'w'}$, $\overline{v'w'}$, $\overline{w'T'}$, and $\overline{w'S'}$. The $k - \epsilon$ closure used in this study is the one implemented in Oceananigans.jl for oceanic applications and briefly described in Appendix B. Despite its much simpler formulation, NORi, a leaner zero-equation model, produces single-column solutions that are practically indistinguishable from both the $k - \epsilon$ and LES solutions within the validation suite used in this work.

NORi compares favorably also with CATKE (Wagner, Hillier, et al., 2025), a recently developed 1.5-equation closure that estimates turbulent diffusivities for momentum and tracers using diagnostic equations for a mixing length and a prognostic equation for TKE. The comparison is presented in Figure J1 in Appendix J. Since NORi performs comparably to CATKE, which has been shown to outperform traditional parameterizations such as KPP (Wagner, Hillier, et al., 2025), NORi would likewise outperform KPP for the scenarios examined in this study. We attribute NORi's skill to a base closure that is physically simple and realistic yet expressive enough in structure for the neural networks to capture the missing physics correctly and stably, enabled by the right calibration strategy. Therefore, we conclude that NORi is a promising candidate for coarse-resolution, long-time step simulations where few-equation closure models are preferred, though further testing in realistic configurations is needed to confirm this.

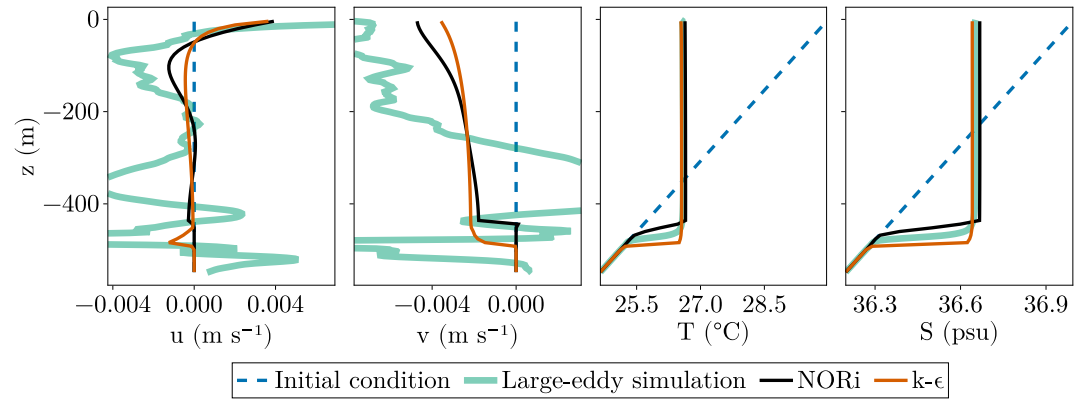


Figure 8. Velocity, temperature, and salinity profiles 60 days after initialization (dashed lines) for simulations forced with a constant wind stress ($J_u^{\text{top}} = -1 \times 10^{-4} \text{ m}^2 \text{ s}^{-2}$) and time-dependent heat and freshwater fluxes given by Equations 38 and 39. The Coriolis parameter is set to $f = 1 \times 10^{-4} \text{ s}^{-1}$. Solutions from NORi, the $k - \epsilon$ model, and the area-averaged large-eddy simulations are shown.

6.3. Numerical Stability

To assess whether NORi is numerically stable on time scales longer than the training horizon of 2 days, we run NORi for 60 days in a single 768 m deep column with a constant grid resolution of 8 m and a time step of 5 min. The simulation is initialized with a constant temperature gradient of $\frac{\partial T_0}{\partial z} = 9.77 \times 10^{-3} \text{ }^\circ\text{C m}^{-1}$ and a salinity gradient of $\frac{\partial S_0}{\partial z} = 1.46 \times 10^{-3} \text{ psu m}^{-1}$, and surface values $T_0(z = 0) = 30^\circ\text{C}$ and $S_0(z = 0) = 37 \text{ psu}$. Surface forcing is imposed through a constant momentum flux of $J_u^{\text{top}} = -1 \times 10^{-4} \text{ m}^2 \text{ s}^{-2}$ and time-dependent sinusoidal temperature and salinity fluxes

$$J_T^{\text{top}} = \left[2 \cos\left(\frac{2\pi}{1 \text{ day}} t\right) + 1 \right] \times 10^{-4} \text{ }^\circ\text{C m s}^{-1}, \quad (38)$$

$$J_S^{\text{top}} = -2 \cos\left(\frac{2\pi}{2.63158 \text{ days}} t\right) \times 10^{-5} \text{ psu ms}^{-1}, \quad (39)$$

with a Coriolis parameter of $f = 1 \times 10^{-4} \text{ s}^{-1}$. The period 2.631 58 days is chosen to be essentially incommensurate with 1 day so that the combined temperature and salinity fluxes generate irregular fluctuations in the surface buoyancy flux, primarily destabilizing, but occasionally stabilizing. Because they have different periods, the temperature and salinity fluxes are not in phase. At certain times, they act in opposite directions (one stabilizing, the other destabilizing), while at other times they are jointly stabilizing or destabilizing. Therefore, in addition to testing numerical stability, we test whether NORi can simulate the evolution of a BL under time-dependent air-sea fluxes despite being trained only in settings with constant air-sea fluxes. NORi correctly turns on the neural network to capture entrainment only when the buoyancy flux drives convection (destabilizing).

In Figure 8 we compare the NORi profiles of the velocity, temperature, and salinity fields at 60 days with the profiles of an LES and the $k - \epsilon$ model forced with the same time-dependent air-sea fluxes. We note that the momentum components u and v can be highly noisy in LES due to internal oscillations that do not contribute to entrainment, so the analysis of results against LES should focus on the temperature and salinity fields. This also illuminates why the loss function used to train NORi does not include momentum terms (see our discussion in Section 5).

The profiles are quite similar, but differences do emerge: the $k - \epsilon$ model produces a mixed layer that is around 50 m deeper than that of NORi at the end of 60 days. Comparing both models with the LES solution, we see that NORi underestimates the penetration of mixing, while $k - \epsilon$ overestimates it, resulting in mixed layer biases of around 25 m in each direction. Further training over longer integration times may be warranted to improve NORi's representation of entrainment. As for $k - \epsilon$, there is literature confirming biases in free convection scenarios characterized by slightly stable stratification in the mixed layer and an increase in stratification at the BL base,

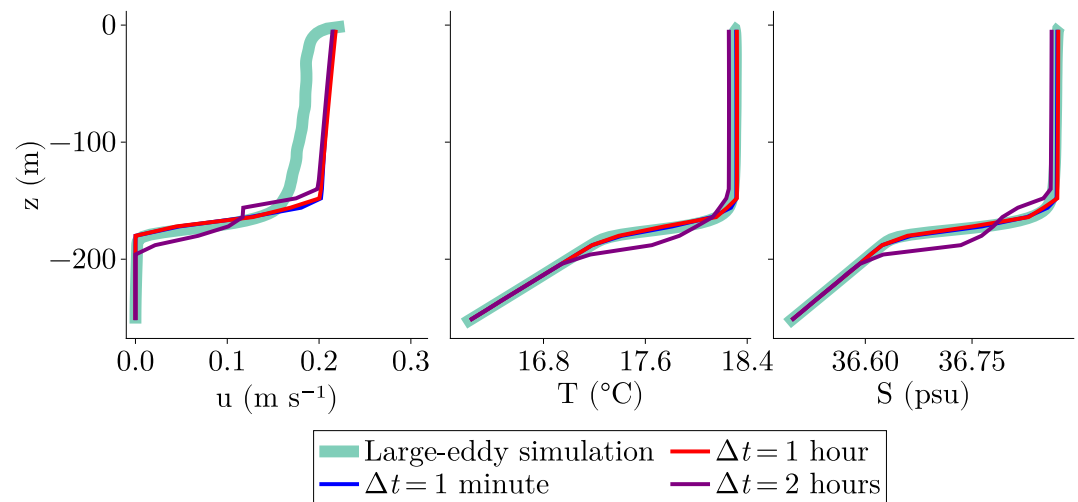


Figure 9. Velocity, temperature, and salinity profiles 4 days after initialization generated with NORi using a range of time steps. The surface forcing parameters are given in the text. The v -component is zero because the simulation is non-rotating and the surface momentum stress acts in the x -direction. The large-eddy simulation solution is also shown for reference.

both of which cannot be captured with purely diffusive closures (Deardorff, 1972; Legay et al., 2025). That said, $k - \epsilon$ parameters have not been calibrated with the same suite of LES used to train NORi, potentially affecting its performance. Regardless, the conclusion stands that NORi and $k - \epsilon$ are virtually indistinguishable over short time scales (see Figure 7), but small biases accumulate, leading to differences over longer time scales. Although these biases are minor at 60 days, their impact on long global ocean simulations may be significant.

6.4. Time Step Dependence

An important consideration in the development of parameterizations for large-scale ocean models is the time step required to get stable and accurate solutions. To address this point, we run NORi and the $k - \epsilon$ model with increasing time steps from 1 min to 2 hr. To focus on the limitations of the time steps associated with the parameterizations and not with the inertial oscillations, we set $f = 0$. The simulations are initialized with a temperature gradient of $\frac{\partial T_0}{\partial z} = 1.5 \times 10^{-2} \text{ } ^\circ\text{C m}^{-1}$ and a salinity gradient of $\frac{\partial S_0}{\partial z} = 2 \times 10^{-3} \text{ psu m}^{-1}$, with surface values $T_0(z = 0) = 20^\circ\text{C}$ and $S_0(z = 0) = 37 \text{ psu}$. The surface fluxes are set to $J_u^{\text{top}} = -1 \times 10^{-4} \text{ m}^2 \text{ s}^{-2}$, $J_T^{\text{top}} = 2 \times 10^{-4} \text{ } ^\circ\text{C m s}^{-1}$, and $J_S^{\text{top}} = -2 \times 10^{-5} \text{ psu m s}^{-1}$, respectively. The models are integrated for 4 days. Figures 9 and 10 show the final velocity, temperature, and salinity profiles for the different time steps computed using the NORi closure and the $k - \epsilon$ closure.

As shown in Figure 9, the NORi solutions are independent of time step for time steps shorter than 1 hr. The solution begins to deviate from the LES solution only when the time step is increased to 2 hr; the mixed layer becomes too deep because NORi begins to overestimate mixing. This time step constraint appears to be set by the diffusive time scale associated with the diffusivity at the base of the BL rather than by the neural networks and may be further extended by using more accurate discretization schemes. In contrast, as seen in Figure 10, the $k - \epsilon$ model achieves numerical convergence only for time steps below 30 min, while it underestimates mixing for longer time steps. It is well known that two-equation BL models, such as $k - \epsilon$, require shorter time steps than zero-equation ones. Indeed, this is the main reason why they are not more commonly used in large-scale coarse-resolution ocean models that require longer time steps (Reichl & Hallberg, 2018; Wagner, Hillier, et al., 2025). In summary, NORi can be used with time steps up to 1 hr without loss of accuracy compared to T and S profiles from LES solutions, making it a good candidate for use in coarse-resolution ocean simulations. It should also be acknowledged that, regardless of time steps, the velocity profiles produced by both NORi and $k - \epsilon$ show larger deviations from the LES solution than the tracer profiles do. However, it appears that the details of the velocity profile are not key for capturing the vertical mixing of tracers.

In a highly unoptimized, naive first implementation of NORi in Oceananigans.jl where data are copied to and from the working arrays for neural network inference, NORi is around 15% slower than $k - \epsilon$ and around 45%

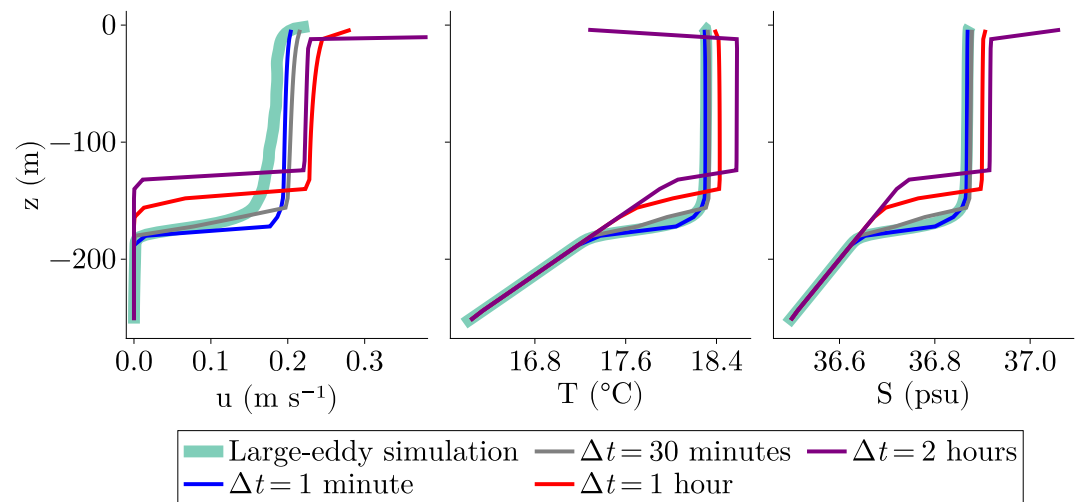


Figure 10. Same as Figure 9 but computed with the $k - \epsilon$ model.

slower than its unaugmented base closure on a per-time step basis. Given that NORi remains accurate at time steps at least twice as long as those required by $k - \epsilon$, we conclude that there are computational advantages to using NORi over two-equation closures in large-scale simulations. At the same time, we have seen in Figure 8 that this set of trained NORi weights produces biases that are of the same magnitude as $k - \epsilon$ when compared to LES in our simulation over 60 days, supporting the case for more comprehensive evaluation and testing of NORi for large-scale simulations.

6.5. Ocean Weather Station Papa: Validation Against Observations

The validation of NORi by comparing with the library of LES solutions has been predicated on two assumptions. First, the LES solutions accurately capture the physics of BL mixing. Second, the neural networks would be able to represent mixing under variable surface forcing despite being trained with LES solutions forced with constant fluxes. Both assumptions are now tested by running NORi as a column model under realistic settings based on observations from Ocean Weather Station Papa (OWS Papa). OWS Papa provides 70-year-long time series of atmospheric variables and temperature and salinity upper ocean profiles at 50.1°N, 144.9°W in the subpolar gyre of the Northeast Pacific (M. Cronin et al., 2023). OWS Papa has been widely used to test single-column BL models as it is situated in a location where lateral fluxes are weak (Burchard & Bolding, 2001), making it an ideal testbed for evaluating vertical mixing parameterizations (Kantha & Clayson, 1994; Large et al., 1994; Yuan et al., 2024).

We used OWS Papa data from 2007 onward, when a new surface mooring was deployed, providing a self-consistent data set. NORi was then run to simulate the fall to winter evolution of the BL for each one of 8 years (we excluded years with more than 72 hr of gaps in the data). The simulations were initialized with observed temperature and salinity profiles from November 1 and integrated forward for 120 days. NORi was forced with surface momentum, heat, and salinity fluxes computed from air-sea observations using the COARE3.0b algorithm (Fairall et al., 2003; M. F. Cronin et al., 2006). We focused on fall to winter observations, when the BL deepens through processes parameterized in NORi. The spring to summer evolution is instead dominated by BL restratification through mesoscale and submesoscale baroclinic instabilities that require information about lateral buoyancy gradients and are currently not included in the formulation of NORi.

The changes in vertically integrated heat and salt content do not match the surface air-sea fluxes in the OWS Papa data. The mismatch is likely due to lateral advection of different water masses. In preliminary NORi runs, this led to large discrepancies between observed and simulated temperature and salinity profiles. To partially mitigate the issue, we subtracted a constant from the time-variable surface fluxes, computed as the difference between the time-integrated fluxes over the fall/winter season and the net change in temperature and salinity over the same period, integrated over the top 100 m, which is the approximate depth of the mixed layer at the end of 120 days. This constant departs from zero because the heat and freshwater budgets at the OWS Papa station are not closed.

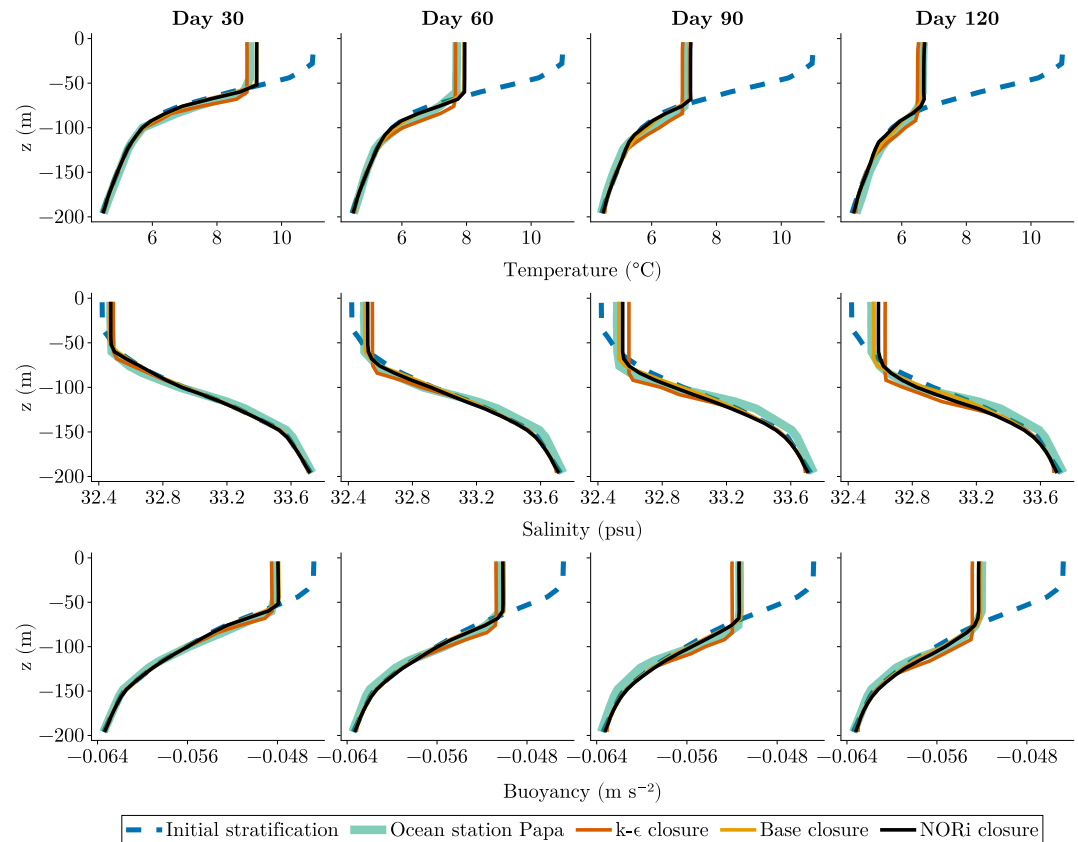


Figure 11. Vertical profiles of temperature, salinity, and buoyancy from Ocean Weather Station Papa (OWS Papa) observations and column model simulations run with NORi, its base closure, and the $k - \epsilon$ closure with prescribed surface fluxes from OWS Papa. Simulations are performed for 8 different years between 2007 and 2023 starting with observed profiles on 1 November and run for 120 days. The instantaneous profiles are the mean over eight different years at 30-day intervals. Some years are excluded because there are large gaps (more than 72 hr) in the data. A constant flux correction is applied on top of the prescribed fluxes to mitigate the drifts in the model solution due to the discrepancies between the observed temperature and salinity budgets and the prescribed surface fluxes.

This correction is not closure-specific and is not tuned to improve NORi relative to the baselines. It is used only to remove the common budget mismatch that a one-dimensional column model cannot represent. Although there are also temperature and salinity drifts deeper in the water column, we do not attempt to correct for those, as they are irrelevant to the testing of BL models.

Figure 11 shows profiles of temperature, salinity, and buoyancy from observations and BL models (NORi, base closure, and $k - \epsilon$) at four times during the 120 days, averaged over eight different years. All models produce profiles close to the observations. It is encouraging that NORi produces reasonable solutions under realistic, time-dependent air-sea fluxes, even though it was trained only on idealized LES forced with constant fluxes. The similarity between the NORi and base closure predictions may naively lead to the conclusion that adding neural networks to capture entrainment was unnecessary. However, the explanation for this similarity is that convection is very weak at OWS Papa, as evidenced by the BL depth changing by less than 50 m over 120 days. In such a scenario, entrainment is expected to be weak. It is actually encouraging that, even though NORi was not explicitly trained on such weak surface forcings, its neural networks do not produce spurious entrainment but instead remain essentially inactive. We note that $k - \epsilon$ slightly overestimates the mixed layer depth, confirming the same bias reported in the validation test cases shown in Figures 7 and 8. More details about the model-data comparison are given in Appendix H where we show the evolution of stratification and model-data differences in Figure H1 and additional comparisons with the CATKE model in Figure J2.

6.6. Summary of Single-Column Comparisons

Taken together, the validation against LES and OWS Papa observations paints a consistent picture of how NORi compares with the state-of-the-art closures $k - \epsilon$ and CATKE. In regimes where turbulence is driven primarily by shear and entrainment is weak, the base closure alone is already sufficient to reproduce the LES and observed profiles, and NORi, $k - \epsilon$, and CATKE are essentially indistinguishable. However, under convective regimes, the base closure systematically underestimates entrainment and the addition of the neural networks is the key for NORi to achieve the same skill as $k - \epsilon$ and CATKE while retaining the longer time steps afforded by a zero-equation closure.

7. Testing NORi on a Large-Scale, Centennial Double-Gyre Simulation

It is difficult to make statements about the numerical stability of neural network-based parameterizations based only on single-column simulations, since the lateral effects of advection and feedbacks between different columns are missing. To test NORi's numerical stability in large-scale simulations, an idealized double-gyre simulation is set up that includes many key components of realistic ocean simulations. This test is not intended to be an evaluation of NORi's predictive skill in realistic ocean simulations, since there is no “ground truth” in a double-gyre setup. Evaluating NORi's skill in large-scale settings would require global/regional ocean simulations and reliable observations to compare against, which is beyond the scope of this work.

Implementing NORi in Oceananigans.jl was straightforward; it is only necessary to set up input and output fields for neural network inference, as the neural network package Lux.jl easily integrates into Oceananigans.jl without the need for intermediate wrappers. NORi works flexibly with both CPU and GPU backends in Oceananigans.jl, allowing us to take advantage of hardware acceleration for large-scale simulations.

The double-gyre has dimensions of $(L_x, L_y, L_z) = (4000, 6000, 1.6)$ km with a grid size of $(N_x, N_y, N_z) = (100, 100, 200)$, which fits on a single NVIDIA V100 GPU. We do not include any other parameterizations to isolate and observe only the effects of BL schemes. The setup uses a β -plane approximation with $f = 2\Omega \sin 45^\circ \approx 1.0 \times 10^{-4} \text{ s}^{-1}$ at the center of the domain and $\beta = \frac{2\Omega \cos 45^\circ}{R_{\text{Earth}}} \approx 1.6 \times 10^{-11} \text{ m}^{-1} \text{ s}^{-1}$, where Ω is the rotation rate of the Earth and R_{Earth} is the radius of the Earth. The meridional extent of the gyre is 6,000 km and corresponds approximately to 54° in latitude. Forcing consists of restoring temperature and salinity on an 8-day time scale to linear profiles decreasing with latitude from 30°C to 0°C and from 37 to 34 psu, respectively (see Figure 12). Momentum forcing is through a sinusoidal wind stress of magnitude $1 \times 10^{-4} \text{ m}^2 \text{ s}^{-2} \approx 0.1 \text{ N m}^{-2}$ driving easterlies north and south and westerlies at the center latitude, as shown in Figure 12. A linear drag with a damping time scale of 30 days is imposed at the deepest grid level. The double-gyre simulation is run for 100 years with a constant time step of 5 min, chosen short enough that the NORi and $k - \epsilon$ BL model results can be directly compared. The simulation is initialized with a uniform temperature field in the horizontal direction and with a linear vertical gradient decreasing from 30°C at the top to 10°C at the bottom. The salinity field is initialized without vertical gradient, but with a linear meridional gradient decreasing from 37 to 34 psu with latitude. Figure 12 shows the equilibrated 3D buoyancy field, a horizontal temperature slice at a depth of 164 m, and the vertically integrated barotropic stream function Ψ averaged over 10 years, defined as

$$\frac{\partial \Psi}{\partial y} = - \int_{-L_z}^0 u dz, \quad \frac{\partial \Psi}{\partial x} = \int_{-L_z}^0 v dz, \quad (40)$$

$$\bar{\Psi} = \frac{1}{10 \text{ years}} \int_{\text{year } 90}^{\text{year } 100} \Psi dt. \quad (41)$$

The barotropic stream function is characterized by two intensified gyres on the western side of the basin. A thermocline develops in the southern half of the domain consistent with the ventilated thermocline theory (Pedlosky, 1996; Vallis, 2017). Deep convection generates a well-mixed water column at the northern edge of the domain.

We run simulations using NORi, the base closure, and the $k - \epsilon$ model. In this naive first implementation of NORi in Oceananigans.jl, we allocated additional working arrays to store the inputs and outputs of the neural networks; thus, a neural network inference is a forward pass over the input working array. Admittedly, this is a

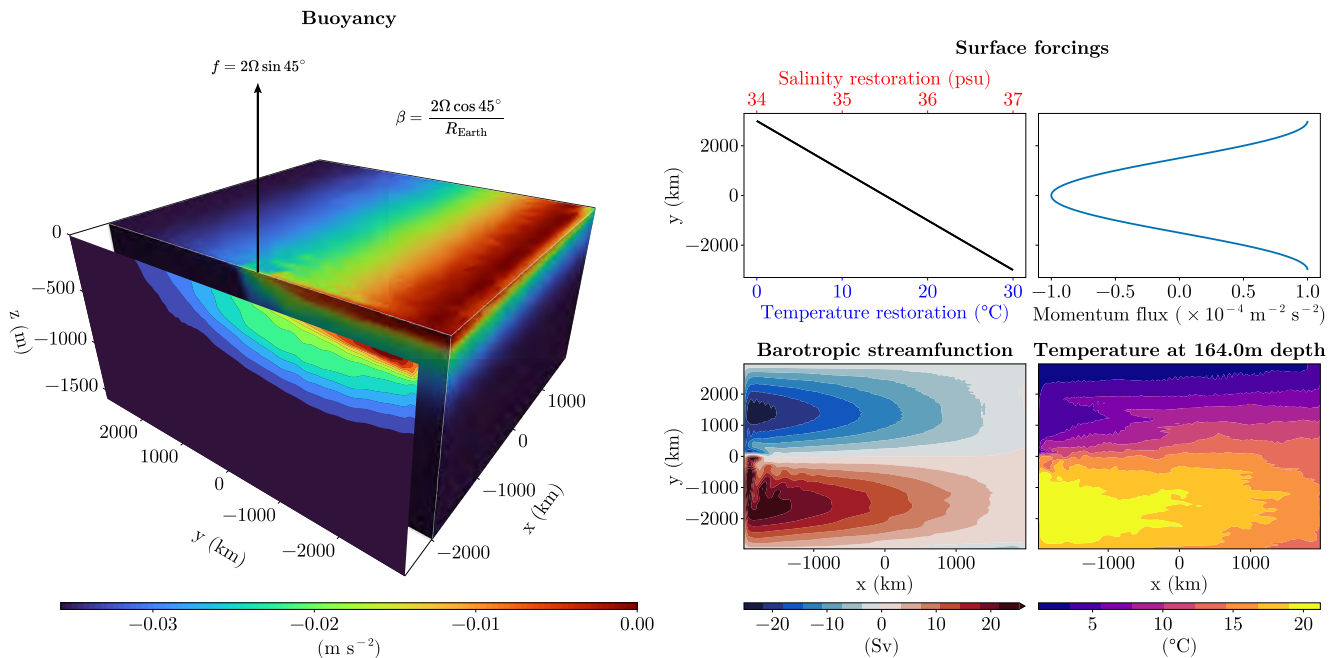


Figure 12. The configuration of the double-gyre simulation. Results from the NORi closure (implemented in Oceananigans.jl and run on GPUs) at $t = 100$ years. The subfigure on the left shows a 3D snapshot of the buoyancy field as well as the zonally averaged buoyancy of the simulation. The subfigures on the right show the temperature and salinity surface-restoring profiles (top left), the surface wind stress (top right), the barotropic stream function averaged over the last 10 years (bottom left), and a horizontal temperature slice at a depth of 164 m (bottom right). The barotropic stream function reaches a maximum of around 40 Sv in the subtropical gyre.

highly inefficient approach as it incurs severe memory overhead as well as data duplication. However, this simple implementation has already yielded a performance that is only 15% slower than $k - \epsilon$ on a per-time step basis. Combined with the larger time steps that NORi supports (see Figures 9 and 10), a more efficient implementation has the potential to be significantly faster than two-equation closures when deployed in large-scale simulations.

NORi is stable for the full 100 years of simulation despite being trained on an ensemble of LES with integration times of less than 2 days. In the many different setups we have run during experimentation, NORi has never crashed due to numerical instabilities. This is significant as numerical stability is one of the biggest challenges facing hybrid modeling of PDEs (Frezzat et al., 2022; Kochkov et al., 2024), where equations are strongly coupled spatially and temporally such that they can be unstable against noisy neural network outputs. Figure 13 compares the zonally averaged temperature at $t = 2.5$ years from double-gyre simulations using the three BL models. NORi simulates deeper mixed layers than the base closure as expected, because the base closure lacks the representation of entrainment. In contrast, NORi produces shallower mixed layers than $k - \epsilon$, especially in the northern part of the domain. Interestingly, this difference pattern is the same as what we observed in the single-column comparison at 60 days with LES in Figure 8. Although NORi and $k - \epsilon$ produce very similar solutions during the 2-day training periods, important differences can emerge over longer time periods, as also discussed in Section 6.3. However, the discrepancies in this case may also arise from interactions between the parameterizations and the large-scale ocean dynamics. Figure 11 in Appendix I confirms that these discrepancies persist in time by showing the same zonally averaged temperature after 100 years. In Appendix J we further show that NORi generates deeper mixed layers than CATKE (Wagner, Hillier, et al., 2025) at 100 years, suggesting that more studies where objective comparisons against observationally constrained data are available, such as regional/global ocean simulations, are needed to determine which model is more accurate.

The results of the double-gyre solutions highlight the importance of recalibrating BL parameterizations in global simulations with climatological observations, because the small residuals in the calibration with LES can build up over longer times and larger scales. However, calibration with short-term, high-resolution simulations or observations provides essential initial estimates for the model parameters, reducing the amount of data required for calibration in the global context.

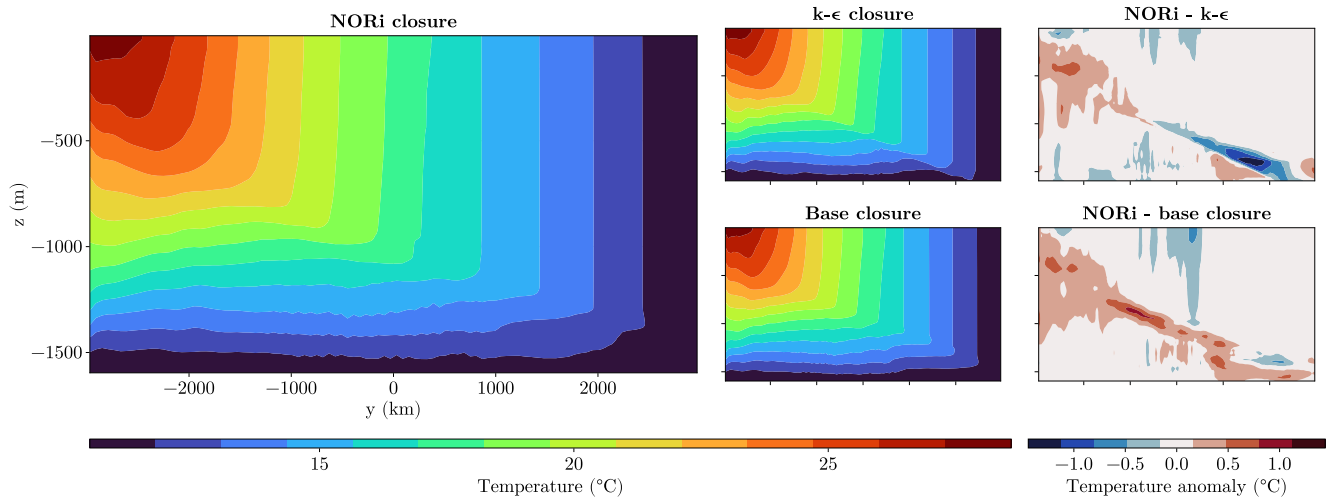


Figure 13. Zonally averaged temperature slice from the three boundary layer models in the double-gyre setup: NORi, $k - \epsilon$, and the base closure. The snapshots are taken at $t = 2.5$ years. The rightmost column shows the temperature differences between NORi and the other two models.

8. Conclusions

We introduced NORi, a parameterization of oceanic surface BL turbulence formulated with neural ODEs where neural networks augment a simple physical closure. To overcome the triple conundrum of high numerical stability, fast model inference, and relatively cheap training, the neural networks, as expressive as they are, must be constrained by our physical understanding of the vertical mixing process. Incorporating physical knowledge into model design drastically reduces the amount of data required and the size of the model (Goyal & Bengio, 2022), allowing us to arrive at a physical closure that is “simple but not any simpler.” Layered on top of this closure are highly expressive neural networks that capture more complicated processes, that is, the nonlocal entrainment fluxes that are missing from the local eddy-diffusivity closure and whose accurate representations have eluded traditional parameterizations for decades. With this approach, NORi outperforms physics-based closures while preserving physical principles that are well understood, suggesting that it is worthwhile to augment traditional parameterizations with neural networks rather than replace them with neural networks altogether.

During the NORi design process, we paid special attention to ensuring that every choice is physically motivated. Since neural networks predict fluxes instead of tendencies, NORi satisfies conservation laws to machine precision by construction. The nonlinear thermodynamics of seawater are explicitly incorporated, so NORi can be readily deployed in realistic ocean simulations. By sharing the same neural network weights across all vertical levels, we ensure that NORi’s representation of entrainment is independent of mixed layer depth. However, this also means that NORi cannot capture deep convection dynamics (see our discussion in Section 5.1), but a simple extension of including the mixed layer depth in the neural network inputs can address this limitation. To reduce inference cost, the neural networks in NORi are relatively lean, with only three hidden layers and 128 neurons per hidden layer, and further work on distillation and pruning could make them even smaller without a significant loss in accuracy. It would also be worth investigating whether evaluating the neural networks in half precision increases inference speed while retaining sufficient accuracy near the sharp gradients at the BL base. NORi generalizes well across a wide range of wind, cooling, evaporation, and rotational strengths under various stratifications and thermodynamic regimes, correctly weakens its neural network components to produce solutions at OWS Papa that are comparable to, if not better than, those obtained with the $k - \epsilon$ model, and is time step-independent below 1 hr, which is appropriate for large-scale, coarse-resolution ocean simulations.

NORi was designed to be calibrated with a posteriori training, where the loss function depends on the full time history of coarse-grained variables rather than on instantaneous turbulent fluxes alone, ensuring that we optimize the model directly to our desired target. By using the neural ODE approach, we also explicitly optimized for numerical stability, since numerically unstable models will incur a high loss and are thus avoided. As a result, NORi is numerically stable for at least 100 years despite being trained on an ensemble of LES with timespans of roughly 2 days, and in the many double-gyre configurations we have explored during experimentation, NORi has

never crashed due to numerical instabilities. However, in pursuit of numerical stability, we have made sacrifices in the scalability of the calibration process because of the serial-in-time nature of neural ODEs. Despite these drawbacks, we have found that a posteriori training is essential for building a model that is usable in practice, as training neural networks directly on instantaneous turbulent fluxes leads to models that are highly numerically unstable and exhibit finite-time blowup. Ultimately, providing more structure and physical inductive biases in the model design is essential for improving training outcomes.

We have designed NORi to be relatively straightforward to implement and deploy in both CPU- and GPU-based ocean models. Importantly, automatic differentiability is only required during training; at inference time, NORi only requires a standard forward pass through the neural networks, making it compatible with any ocean model that supports neural network evaluation. Using the Julia software stack, we integrated NORi directly into Oceananigans.jl without intermediate wrappers, and took advantage of its GPU-oriented design to avoid CPU–GPU data transfers during inference. In our first naive implementation, additional working arrays were allocated for neural network inputs and outputs, leading to memory-bound performance due to large transfers of values between working arrays and native Oceananigans.jl fields. Despite such inefficiencies, NORi's performance is only 15% slower than the state-of-the-art $k - \epsilon$ closure on a per-time step basis. A more efficient implementation that writes custom forward-pass kernels accessing native Oceananigans.jl arrays directly would eliminate the need for these working arrays and is a performance engineering task for future work.

A few important challenges remain to be tackled in future work. NORi has been trained and deployed with a fixed grid resolution of 8 m, which is comparable to that of state-of-the-art global ocean models (around 10 m near the BL base). However, NORi cannot yet be used flexibly with variable vertical grid sizes. A straightforward extension may be to interpolate between the dynamics grid and the physics grid, running NORi inference at a fixed 8 m resolution and then interpolating the fluxes back to the dynamics grid, similar to what is done in Molod (2009). A first attempt at this approach using simple linear interpolation yielded unsatisfactory results, as entrainment fluxes weaken with coarser vertical resolutions that fail to represent the sharp gradients at the BL base. More sophisticated reconstruction techniques, like making resolution an input to the neural networks, may be potential solutions. Alternatively, one could implement a resolution-independent representation of the entrainment fluxes—van Zanten et al. (1999) infers the fluxes from prognostic equations for the inversion depth and minimum buoyancy flux—though this would require a reformulation of the neural network architecture. Failing all of the above, one may simply retrain the neural networks on the desired grid, even though this would limit the utility of the neural networks except in operational applications where the grid resolution is fixed.

Once variable vertical grid support is implemented and validated, NORi could be run in a regional and/or global ocean simulation with realistic forcings—setups that can be verified against observations. This would allow us to assess whether NORi alleviates known mixed layer depth biases in large-scale climate simulations. Ultimately, we plan to calibrate NORi jointly with other ocean parameterizations, such as the Gent and McWilliams (1990) and Redi (1982) parameterizations as well as interior mixing parameterizations which also enhance vertical mixing, a posteriori within both regional and global ocean simulations, where the model is initialized, forced, and verified with observations. This would allow us to fine-tune NORi and reduce biases found in large-scale simulations, but not present in idealized column setups.

Joint calibration of different parameterizations is useful to eliminate compensating errors. A first step in that direction would be to jointly calibrate NORi's base closure and neural networks to ensure that the neural networks do not compensate for biases other than nonlocal entrainment. The neural network weights and base closure parameters learned from the separate calibration would be used as priors to reduce the parameter space to be explored for a posteriori calibration given the complexity of the forward model. The next step would be to compare NORi with current state-of-the-art BL parameterizations calibrated against a common suite of ocean scenarios to rigorously assess their respective biases. Should the comparison identify structural deficiencies, one may consider adopting a more sophisticated base closure, like $k - \epsilon$ (Umlauf & Burchard, 2003) or CATKE (Wagner, Hillier, et al., 2025), or increasing the depth of the neural networks.

Beyond free-running comparisons, Wei et al. (2026) also demonstrated an assessment approach in which mean fields in single-column models are strongly nudged toward LES values so that flux skill can be diagnosed independently of the coupled mean-field evolution, since the errors in mean fields and turbulent fluxes are

entangled. This may serve as a useful tool for evaluating NORi alongside other parameterizations in future work, revealing systematic biases NORi might possess.

Another important consideration is that in its current implementation, NORi is active for five grid points below the BL base, resulting in a 40 m layer below the BL base, which is thicker than the plume penetration depth in all the LES cases we have run, including the testing case where the BL deepened to around 500 m. In that scenario, there is a nonzero wind stress, thus the thickness of the entrainment layer is also influenced by shear. For free convection scenarios, the 1/5 rule applies (Carson, 1973), so it is possible that our current implementation holds only for BLs up to 1 km. To address such regimes, we can run LES with much longer integration times and deeper BLs and train NORi with additional grid points below the BL base. Even though this extension is straightforward, we have opted not to pursue this as this exercise is purely academic, since global ocean models have vertical resolutions that coarsen with depth. Under such circumstances, five grid points may remain sufficient.

The suite of LES should also be extended to include missing BL physics. Surface wave effects, that is, Langmuir turbulence, are not accounted for in the LES generated in this work but are known to substantially affect the evolution of the BL under very strong winds. Different LES numerical implementations, such as explicit rather than implicit LES closure or addition of wall-models, may provide additional validation of the data set used for calibration. Retraining or fine-tuning NORi with additional LES would be straightforward, since neither the architecture nor the training procedure would need to change.

We have focused on the deepening phases of the BL in the absence of lateral density gradients, when the dynamics is controlled by destabilizing surface fluxes. However, submesoscale instabilities at upper ocean buoyancy fronts affect the deepening of the BL (Taylor & Ferrari, 2010; Thomas & Ferrari, 2008) and dominate restratification when surface fluxes are stabilizing (Fox-Kemper et al., 2008; Taylor & Thompson, 2023). In the future, we aim to extend NORi to include the impact of submesoscale processes within a similar framework, thereby allowing for a more complete representation of the BL dynamics across the entire seasonal cycle.

Appendix A: Column Model Derivation of the Ocean Surface Boundary Layer

For easier reference, we first rewrite Equations 1–4 in their component form:

$$\frac{\partial u}{\partial t} = -\nabla \cdot (uu) + fv - \partial_x p, \quad (\text{A1})$$

$$\frac{\partial v}{\partial t} = -\nabla \cdot (uv) - fu - \partial_y p, \quad (\text{A2})$$

$$\frac{\partial w}{\partial t} = -\nabla \cdot (uw) - \partial_z p + b, \quad (\text{A3})$$

$$\frac{\partial T}{\partial t} = -\nabla \cdot (uT), \quad (\text{A4})$$

$$\frac{\partial S}{\partial t} = -\nabla \cdot (uS), \quad (\text{A5})$$

$$\nabla \cdot \mathbf{u} = 0. \quad (\text{A6})$$

We again note that by decomposing a field into its horizontally averaged component and perturbations away from the average $\phi = \bar{\phi}(z, t) + \phi'(x, y, z, t)$, in a laterally doubly periodic domain, the horizontal averaging operator $\bar{(\cdot)}$ has the property that

$$\frac{\partial \bar{\phi}}{\partial x} = \frac{\partial \bar{\phi}}{\partial x} = 0, \quad (\text{A7})$$

$$\frac{\partial \bar{\phi}}{\partial y} = \frac{\partial \bar{\phi}}{\partial y} = 0, \quad (\text{A8})$$

$$\overline{\phi'} \equiv 0. \quad (\text{A9})$$

Applying the horizontal averaging to Equation A6 and noting properties Equation A7 and A8, we obtain the following:

$$\frac{\partial \overline{\mu}}{\partial x} + \frac{\partial \overline{p}}{\partial y} + \frac{\partial \overline{w}}{\partial z} = 0, \quad (\text{A10})$$

$$\frac{\partial \overline{w}}{\partial z} = 0, \quad (\text{A11})$$

$$\therefore \overline{w} = C \quad (\text{A12})$$

where C is a constant. We now apply the horizontal averaging operator to the impenetrability boundary condition on the surface $z = 0$, that is, Equation 7, which yields

$$\overline{w(z=0)} = 0, \quad (\text{A13})$$

$$\overline{w} = C = 0, \quad (\text{A14})$$

since $\overline{w}$ is a constant everywhere in the column.

Expanding the terms in Equation A1 and applying the horizontal averaging operator, we have (in Cartesian coordinates)

$$\frac{\partial \overline{u}}{\partial t} = -\partial_x \overline{u u'} - \partial_y \overline{v u'} - \partial_z \overline{w u'} + f \overline{v} - \partial_x \overline{p}, \quad (\text{A15})$$

$$= -\frac{\partial}{\partial z} \left(\overline{u u' w'} + \overline{u' w' w'} + \overline{u w' w'} + \overline{u' w' w'} \right) + f \overline{v}, \quad (\text{A16})$$

$$= -\frac{\partial}{\partial z} \left(\overline{u u' w'} + \overline{u' w' w'} \right) + f \overline{v}, \quad (\text{A17})$$

$$= -\frac{\partial \overline{u' w'}}{\partial z} + f \overline{v}, \quad (\text{A18})$$

where $\overline{u' w'}$ is the vertical flux of horizontal momentum in the x -direction, yielding Equation 11. Similarly, we can apply the same procedure to Equations A2, A4, and A5 to arrive at Equations 12–14, respectively.

Appendix B: The Implementation of the $k - \epsilon$ Boundary Layer Closure

The $k - \epsilon$ implementation in Oceananigans.jl follows the generic length-scale (GLS) framework of Umlauf and Burchard (2003) tailored to oceanic applications, using the Canuto-A stability functions of Canuto et al. (2001) together with the realizability constraints of Umlauf and Burchard (2005) on the stratification and shear numbers $\alpha^N = N^2(k/\epsilon)^2$ and $\alpha^M = |S|^2(k/\epsilon)^2$ where N^2 is the buoyancy frequency and S is the shear. Because the momentum and tracer stability functions $\mathbb{S}_u(\alpha^N, \alpha^M)$ and $\mathbb{S}_c(\alpha^N, \alpha^M)$ are rational functions of the local stratification and shear that share a common denominator but differ in their numerators, the turbulent Prandtl number $\text{Pr}_t = \mathbb{S}_u/\mathbb{S}_c$ is itself a function of α^N and α^M rather than a fixed constant. The eddy viscosity and diffusivities are

reconstructed as $\nu_t = \mathbb{S}_u k^2/\epsilon$ and $\kappa_t = \mathbb{S}_c k^2/\epsilon$, with separate Schmidt-number scalings for the k and ϵ diffusivities. The turbulent kinetic energy (TKE) (k) equation balances vertical turbulent diffusion, shear production, buoyancy production, and viscous dissipation with TKE Schmidt number $\sigma_k = 1$ (Launder & Spalding, 1974).

The dissipation (ϵ) equation uses $C_{\epsilon 1} = 1.44$ for shear production and $C_{\epsilon 2} = 1.92$ for dissipation (Launder & Spalding, 1974), and Schmidt number $\sigma_\epsilon = 1.2$ fixed by consistency with the law of the wall in a neutral logarithmic BL (Umlauf & Burchard, 2003). The buoyancy-production coefficient is implemented with the standard sign-of- N^2 split into stable and unstable branches $C_{\epsilon 3}^+$ and $C_{\epsilon 3}^-$. Although the standard GLS practice uses regime-dependent values, here we set $C_{\epsilon 3}^+ = C_{\epsilon 3}^- = -0.65$ uniformly. To preserve positivity of k on long integrations, dissipation is treated implicitly via $\omega_\epsilon = \epsilon/k$, and the negative part of the buoyancy production term is moved to the implicit side following a Patankar-style splitting (Burchard et al., 2003). Vertical diffusion of u , v , T , S , k , and ϵ is handled with a vertically implicit time discretization. Two limiters keep the closure well-posed in strongly stratified regions. First, the turbulent length scale is capped by a Galperin-type bound and by the domain depth, $\ell_{\max} = \min(L_z, 0.75\sqrt{k/N^2})$ (Galperin et al., 1988). Second, since the GLS closure relates dissipation to length scale via $\epsilon = (c_\mu^0)^3 k^{3/2}/\ell$, this length-scale cap induces a corresponding lower bound on dissipation, $\epsilon \geq (c_\mu^0)^3 k^{3/2}/\ell_{\max}$, which we enforce together with a hard floor of $1 \times 10^{-12} \text{ m}^2 \text{ s}^{-3}$. A TKE floor of $k_{\min} = 1 \times 10^{-6} \text{ m}^2 \text{ s}^{-2}$ and a 1-min Euler damping of any spurious negative TKE provide additional numerical robustness.

At the surface, we impose a zero-flux TKE boundary condition, omitting the wave-breaking TKE injection of Craig and Banner (1994). Surface forcing of turbulence enters only through the dissipation boundary condition, which is set using a Charnock roughness length $\ell_r = 0.11 u_*^2/g$ (Charnock, 1955; Umlauf & Burchard, 2003).

Appendix C: Training and Testing Suite

In this section, we provide the complete training and testing suite used to train, validate, and test NORi. Table C1 shows the training cases driven by shear instability (wind-driven mixing) and two testing cases used to evaluate the base closure, Table C2 shows the training free convection cases driven by buoyancy instability (no winds), Table C3 shows the training cases driven by both shear and convective instabilities, while Table C4 shows the remaining miscellaneous training cases that include cases with the same forcings and stratifications as some of those before but in different thermodynamic regimes (lower temperatures) as well as cases with no temperature or salinity gradient. Finally, Table C5 shows the validation cases used to select the final weights of the neural network after training. Note that for all tables we adopt this sign convention: positive J_T^{top} denotes upward heat flux (cooling); negative J_S^{top} denotes downward salt flux (evaporation); negative J_u^{top} denotes downward momentum flux (i.e., into the ocean). Also, $J_v^{\text{top}} = 0$ in all simulations since imposing nonzero J_v^{top} is dynamically equivalent to imposing nonzero J_u^{top} during training and inference since momentum is only implicitly provided to NORi via the Richardson number. Initial profiles ϕ are linear with vertical gradient $\partial_z \phi$ and surface value ϕ^{top} .

Table C1
Cases Used to Train and Evaluate the Base Closure

J_T^{top} ($^\circ\text{C m s}^{-1}$)	J_S^{top} (psu m s $^{-1}$)	J_u^{top} (m 2 s $^{-2}$)	$\frac{\partial T}{\partial z}$ ($^\circ\text{C m}^{-1}$)	$\frac{\partial S}{\partial z}$ (psu m $^{-1}$)	f (s $^{-1}$)	T^{top} ($^\circ\text{C}$)	S^{top} (psu)
0	0	-5×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	0	-2×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	0	-5×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	0	-2×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	0	-5×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
0	0	-2×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
-3×10^{-4}	3×10^{-5}	-5×10^{-4}	1.50×10^{-2}	1.50×10^{-3}	1×10^{-4}	20	37
0	0	-3.50×10^{-4}	1.50×10^{-2}	1.50×10^{-3}	1×10^{-4}	20	37

Note. Rows 1–6 are shear-driven training cases; rows 7–8 are held-out testing cases for the base closure and are not used to validate the neural networks.

Table C2

Free Convection Cases Used to Train NORi

J_T^{top} ($^{\circ}\text{C m s}^{-1}$)	J_S^{top} (psu m s^{-1})	J_u^{top} ($\text{m}^2 \text{s}^{-2}$)	$\frac{\partial T}{\partial z}$ ($^{\circ}\text{C m}^{-1}$)	$\frac{\partial S}{\partial z}$ (psu m^{-1})	f (s^{-1})	T^{top} ($^{\circ}\text{C}$)	S^{top} (psu)
5×10^{-4}	0	0	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
1×10^{-4}	0	0	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	-5×10^{-5}	0	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	-1×10^{-5}	0	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
3×10^{-4}	-3×10^{-5}	0	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
3×10^{-4}	-3×10^{-5}	0	1.40×10^{-2}	-2.10×10^{-3}	8×10^{-5}	18	36.6
1×10^{-4}	-1×10^{-5}	0	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	10	36.6
5×10^{-4}	0	0	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
1×10^{-4}	0	0	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	-5×10^{-5}	0	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	-1×10^{-5}	0	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
3×10^{-4}	-3×10^{-5}	0	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
3×10^{-4}	-3×10^{-5}	0	1.30×10^{-2}	-7.50×10^{-4}	0	14.5	35
1×10^{-4}	-1×10^{-5}	0	1.30×10^{-2}	7.50×10^{-4}	0	10	35
5×10^{-4}	0	0	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
1×10^{-4}	0	0	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
0	-5×10^{-5}	0	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
0	-1×10^{-5}	0	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
3×10^{-4}	-3×10^{-5}	0	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9

Table C3

Wind + Convection Cases Used to Train NORi

J_T^{top} ($^{\circ}\text{C m s}^{-1}$)	J_S^{top} (psu m s^{-1})	J_u^{top} ($\text{m}^2 \text{s}^{-2}$)	$\frac{\partial T}{\partial z}$ ($^{\circ}\text{C m}^{-1}$)	$\frac{\partial S}{\partial z}$ (psu m^{-1})	f (s^{-1})	T^{top} ($^{\circ}\text{C}$)	S^{top} (psu)
5×10^{-4}	0	-2×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
1×10^{-4}	0	-2×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
5×10^{-4}	0	-5×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
1×10^{-4}	0	-5×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	-5×10^{-5}	-2×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	-1×10^{-5}	-2×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	-5×10^{-5}	-5×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
0	-1×10^{-5}	-5×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
3×10^{-4}	-3×10^{-5}	-2×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
3×10^{-4}	-3×10^{-5}	-5×10^{-4}	1.40×10^{-2}	2.10×10^{-3}	8×10^{-5}	18	36.6
5×10^{-4}	0	-2×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
1×10^{-4}	0	-2×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
5×10^{-4}	0	-5×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
1×10^{-4}	0	-5×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	-5×10^{-5}	-2×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	-1×10^{-5}	-2×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	-5×10^{-5}	-5×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
0	-1×10^{-5}	-5×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
3×10^{-4}	-3×10^{-5}	-2×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35

Table C3

Continued

J_T^{top} ($^{\circ}\text{C m s}^{-1}$)	J_S^{top} (psu m s^{-1})	J_u^{top} ($\text{m}^2 \text{s}^{-2}$)	$\frac{\partial T}{\partial z}$ ($^{\circ}\text{C m}^{-1}$)	$\frac{\partial S}{\partial z}$ (psu m^{-1})	f (s^{-1})	T^{top} ($^{\circ}\text{C}$)	S^{top} (psu)
3×10^{-4}	-3×10^{-5}	-5×10^{-4}	1.30×10^{-2}	7.50×10^{-4}	0	14.5	35
5×10^{-4}	0	-2×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
1×10^{-4}	0	-2×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
5×10^{-4}	0	-5×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
1×10^{-4}	0	-5×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
0	-5×10^{-5}	-2×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
0	-1×10^{-5}	-2×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
0	-5×10^{-5}	-5×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
0	-1×10^{-5}	-5×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
5×10^{-4}	-5×10^{-5}	-2×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9
5×10^{-4}	-5×10^{-5}	-5×10^{-4}	-2.50×10^{-2}	-4.50×10^{-3}	-1.25×10^{-4}	0	33.9

Table C4

Miscellaneous Training Cases Used to Train NORi, Including Cases With No Temperature or Salinity Gradients and Cases With Very Strong Stratification at Warm Temperatures

J_T^{top} ($^{\circ}\text{C m s}^{-1}$)	J_S^{top} (psu m s^{-1})	J_u^{top} ($\text{m}^2 \text{s}^{-2}$)	$\frac{\partial T}{\partial z}$ ($^{\circ}\text{C m}^{-1}$)	$\frac{\partial S}{\partial z}$ (psu m^{-1})	f (s^{-1})	T^{top} ($^{\circ}\text{C}$)	S^{top} (psu)
3×10^{-4}	0	-1×10^{-4}	1×10^{-2}	0	1×10^{-4}	10	35
5×10^{-4}	0	-1×10^{-4}	1.40×10^{-2}	0	1×10^{-4}	10	37
0	-5×10^{-5}	-1×10^{-4}	0	-5×10^{-3}	1.50×10^{-4}	0	34
0	-5×10^{-5}	-1×10^{-4}	0	-1×10^{-3}	1×10^{-4}	10	35
5×10^{-4}	-5×10^{-5}	-1×10^{-4}	6×10^{-2}	8×10^{-3}	1×10^{-4}	30	37
3×10^{-4}	-3×10^{-5}	-1×10^{-4}	6×10^{-2}	8×10^{-3}	1×10^{-4}	30	37
1×10^{-4}	-1×10^{-5}	-1×10^{-4}	6×10^{-2}	8×10^{-3}	1×10^{-4}	30	37

Table C5

Validation Cases to Evaluate Performance and Select Final Trained Weights of NORi

J_T^{top} ($^{\circ}\text{C m s}^{-1}$)	J_S^{top} (psu m s^{-1})	J_u^{top} ($\text{m}^2 \text{s}^{-2}$)	$\frac{\partial T}{\partial z}$ ($^{\circ}\text{C m}^{-1}$)	$\frac{\partial S}{\partial z}$ (psu m^{-1})	f (s^{-1})	T^{top} ($^{\circ}\text{C}$)	S^{top} (psu)
3.50×10^{-4}	0	0	1.50×10^{-2}	2×10^{-3}	1×10^{-4}	20	37
0	-3.50×10^{-5}	0	1.50×10^{-2}	2×10^{-3}	1×10^{-4}	20	37
2×10^{-4}	-2×10^{-5}	-1×10^{-4}	1.50×10^{-2}	2×10^{-3}	1×10^{-4}	20	37
2×10^{-4}	-1.50×10^{-5}	-3.50×10^{-4}	1×10^{-2}	1.50×10^{-3}	1×10^{-4}	17	36
1.50×10^{-4}	-2×10^{-5}	-2.50×10^{-4}	1.70×10^{-2}	1.80×10^{-3}	1×10^{-4}	12	37
5×10^{-5}	-5×10^{-6}	-4×10^{-4}	1.20×10^{-2}	1.20×10^{-3}	1×10^{-4}	10	37
3.50×10^{-4}	-3.50×10^{-5}	-3×10^{-4}	1×10^{-2}	-2×10^{-3}	1.50×10^{-4}	16	34
4.50×10^{-4}	-4×10^{-5}	-4.50×10^{-4}	1.30×10^{-2}	-1×10^{-3}	7×10^{-5}	13	35
3.50×10^{-4}	0	0	1×10^{-2}	5×10^{-4}	0	17	34.5
0	-3.50×10^{-5}	0	1×10^{-2}	5×10^{-4}	0	17	34.5
2×10^{-4}	-2×10^{-5}	-1×10^{-4}	1×10^{-2}	5×10^{-4}	0	17	34.5
2×10^{-4}	-1.50×10^{-5}	-3.50×10^{-4}	1.20×10^{-2}	7×10^{-4}	-1×10^{-5}	13	36
1.50×10^{-4}	-2×10^{-5}	-2.50×10^{-4}	1.60×10^{-2}	6×10^{-4}	-2×10^{-5}	16	34
5×10^{-5}	-5×10^{-6}	-4×10^{-4}	1.10×10^{-2}	3×10^{-4}	3×10^{-5}	10	36.5

Table C5
Continued

J_T^{top} ($^{\circ}\text{C m s}^{-1}$)	J_S^{top} (psu m s^{-1})	J_u^{top} ($\text{m}^2 \text{s}^{-2}$)	$\frac{\partial T}{\partial z}$ ($^{\circ}\text{C m}^{-1}$)	$\frac{\partial S}{\partial z}$ (psu m^{-1})	f (s^{-1})	T^{top} ($^{\circ}\text{C}$)	S^{top} (psu)
3.50×10^{-4}	-3.50×10^{-5}	-3×10^{-4}	1.50×10^{-2}	-5×10^{-4}	-5×10^{-5}	20	37
4.50×10^{-4}	-4×10^{-5}	-4.50×10^{-4}	1.70×10^{-2}	-8×10^{-4}	-1×10^{-4}	12	35
3.50×10^{-4}	0	0	-2×10^{-2}	-4.70×10^{-3}	-1.50×10^{-4}	0	34.5
0	-3.50×10^{-5}	0	-2×10^{-2}	-4.70×10^{-3}	-1.50×10^{-4}	0	34.5
2×10^{-4}	-2×10^{-5}	-1×10^{-4}	-2×10^{-2}	-4.70×10^{-3}	-1.50×10^{-4}	0	34.5
2×10^{-4}	-1.50×10^{-5}	-3.50×10^{-4}	-1.70×10^{-2}	-4×10^{-3}	1.50×10^{-4}	3	34
4×10^{-4}	-4×10^{-5}	-2.50×10^{-4}	-1.50×10^{-2}	-4×10^{-3}	1×10^{-4}	-1	36
5×10^{-5}	-5×10^{-6}	-4×10^{-4}	-1×10^{-2}	-3×10^{-3}	1.25×10^{-4}	0	33.5
3.50×10^{-4}	-3.50×10^{-5}	-3×10^{-4}	-2×10^{-2}	-4×10^{-3}	-1.40×10^{-4}	1	35.5
4.50×10^{-4}	-4×10^{-5}	-4.50×10^{-4}	-2×10^{-2}	-4.50×10^{-3}	-1×10^{-4}	2	34
4.50×10^{-4}	0	-2×10^{-4}	3×10^{-2}	0	1×10^{-4}	20	37
1.50×10^{-4}	0	-3×10^{-4}	1×10^{-2}	0	1×10^{-4}	15	35
0	-4×10^{-5}	-3×10^{-4}	0	-4.50×10^{-3}	-1×10^{-4}	17	36
0	-2×10^{-5}	-1.50×10^{-4}	0	-2.50×10^{-3}	-1×10^{-4}	16	34
4×10^{-4}	-4×10^{-5}	-4×10^{-4}	5×10^{-2}	-7×10^{-3}	1×10^{-4}	25	36
3×10^{-4}	-2×10^{-5}	-2×10^{-4}	3×10^{-2}	-7.50×10^{-3}	1×10^{-4}	27	35

Appendix D: Loss Scalings

As seen in Equations 31 and 32, during training of both the base closure and the neural networks, we use a weighted loss function to balance the contributions of each variable. The weights are set using normalization factors A_{ϕ}^a for the field ϕ in simulation a within the training suite. The difference between the loss functions for the base closure and neural network training is that velocity components u and v are not included in the neural network training loss as these fields are dominated by inertial oscillations which do not affect vertical mixing. The values of A_{ϕ}^a are recalculated at the beginning of each training stage to renormalize the training loss. A_{ϕ}^a is dependent on the relative contribution of temperature and salinity to the potential density in the initial condition of the neural ODEs, as well as the relative contribution of each loss component to the total loss. A_T^a and A_S^a are calculated as follows:

$$A_T^a = \frac{\alpha \|\Delta T_0^a\|_{\infty} + \beta \|\Delta S_0^a\|_{\infty}}{\alpha \|\Delta T_0^a\|_{\infty}}, \quad (\text{D1})$$

$$A_S^a = \frac{\alpha \|\Delta T_0^a\|_{\infty} + \beta \|\Delta S_0^a\|_{\infty}}{\beta \|\Delta S_0^a\|_{\infty}} \quad (\text{D2})$$

where T_0^a and S_0^a denote the initial temperature and salinity profiles in training case a , with $\|\Delta \phi_0^a\|_{\infty} = \max \bar{\phi}_0^a - \min \bar{\phi}_0^a$ giving their vertical range. Here, α and β are the thermal expansion coefficient and haline contraction coefficient of seawater, respectively. The remaining normalization factors are chosen such that

$$\frac{\mathcal{L}_{\sigma}^a}{\mathcal{L}_T^a + \mathcal{L}_S^a} = \frac{1}{9}, \quad (\text{D3})$$

$$\mathcal{L}_{\text{profile}}^a = \mathcal{L}_{\text{gradient}}^a, \quad (\text{D4})$$

$$\mathcal{L}_{\text{gradient}}^a = A_{\partial_z T}^a \delta(\partial_z \bar{T})^a + A_{\partial_z S}^a \delta(\partial_z \bar{S})^a + A_{\partial_z \sigma}^a \delta(\partial_z \bar{\sigma})^a \quad (\text{D5})$$

$$= A_{\text{gradient}}^a (A_T^a \delta(\partial_z \bar{T})^a + A_S^a \delta(\partial_z \bar{S})^a + A_{\sigma}^a \delta(\partial_z \bar{\sigma})^a), \quad (\text{D6})$$

$$\delta\bar{\phi} \equiv \frac{1}{N_t N_z} \sum_{i=1}^{N_t} \sum_{j=1}^{N_z} |\bar{\phi}^{i,j} - \bar{\phi}_{\text{LES}}^{i,j}|^2 \quad (\text{D7})$$

where $\bar{\phi}$ and $\bar{\phi}_{\text{LES}}$ are one of the vertical gradient profiles (temperature, salinity, potential density) as a function of vertical level i , time j , and scenario a —the overbars are a reminder that we are only parameterizing the area-averaged profiles.

Appendix E: Neural Network Architecture

In NORi, neural networks are used to predict temperature and salinity entrainment fluxes, thereby augmenting the base physical closure. This is reflected in the design of neural networks, where they are only active within the zone where entrainment occurs and are inactive everywhere else in the vertical column. For cell indices centered on the interior face $i = 2, 3, \dots, N$, where $i = 1$ indicates the location of the face at the ocean surface and $i = N + 1$ indicates the grid point at the bottom of the domain,

$$J_{\text{NN}_\phi}^i = \begin{cases} \text{NN}_\phi(\mathbf{h}^i; \boldsymbol{\theta}_\phi), & \text{for } i = i_{\min}, i_{\min} + 1, \dots, i_{\max}, \\ 0, & \text{otherwise,} \end{cases} \quad (\text{E1})$$

$$\mathbf{h}^i = \left[(\partial_z \bar{T})_{\text{zone}}^i, (\partial_z \bar{S})_{\text{zone}}^i, (\partial_z \bar{\sigma})_{\text{zone}}^i, \arctan(\mathbf{R}^i_{\text{zone}}), J_b^{\text{top}} \right]^\top, \quad (\text{E2})$$

$$\boldsymbol{\phi}_{\text{zone}}^i \equiv [\phi^{\min\{i+2, N\}}, \phi^{\min\{i+1, N\}}, \phi^i, \phi^{\max\{i-1, 2\}}, \phi^{\max\{i-2, 2\}}] \quad (\text{E3})$$

where NN_ϕ is a neural network to predict the turbulent flux of tracer ϕ , $\mathbf{h}^i$ is the input vector to the neural network at index i , $\boldsymbol{\theta}_\phi$ are the trainable weights for the neural network NN_ϕ , $i_{\min}$ and $i_{\max}$ are the first and last grid indices, between which (inclusive) the neural networks are active, $\boldsymbol{\phi}_{\text{zone}}^i$ denotes the five-element vector of variable ϕ around the local neighborhood of index i , ∂_z is the discrete gradient operator in z , and J_b^{top} is the surface buoyancy flux. For any face-centered grid point within the “entrainment zone,” the values of the temperature and salinity fluxes are predicted by NN_T and NN_S , respectively, with each network outputting a scalar that is the predicted value of temperature or salinity flux at the grid point.

The inputs of NN_T and NN_S are the temperature gradient, salinity gradient, potential density gradient, and gradient Richardson number of five grid points within the neighborhood of the grid point intended for inference, as well as the surface buoyancy flux. The inputs and outputs of the neural networks are normalized using a Z-score normalization against the entire training suite to ensure that for each variable, the distribution of their values within the training suite has zero mean and unit variance. More details on feature normalization can be found in Appendix G.

We adopt the principle of “simple, but not any simpler” in selecting the appropriate input variables to be included in the feature vector $\mathbf{h}^i$, and our rationale for each is as follows:

1. The rate of nonlocally forced entrainment, which is the process we want to model using neural networks, depends primarily on the strength of the buoyancy flux J_b^{top} at the ocean surface. This is because the surface buoyancy flux is the source of entrainment mixing through the generation of vertically coherent plumes. The buoyancy flux is a function of the temperature and salinity fluxes, but we provide only the buoyancy flux to the neural networks to eliminate the need for neural networks to learn the implicit correlation between temperature, salinity, and buoyancy fluxes. This is because the entrainment penetration depth is a function of buoyancy, not temperature or salinity separately.

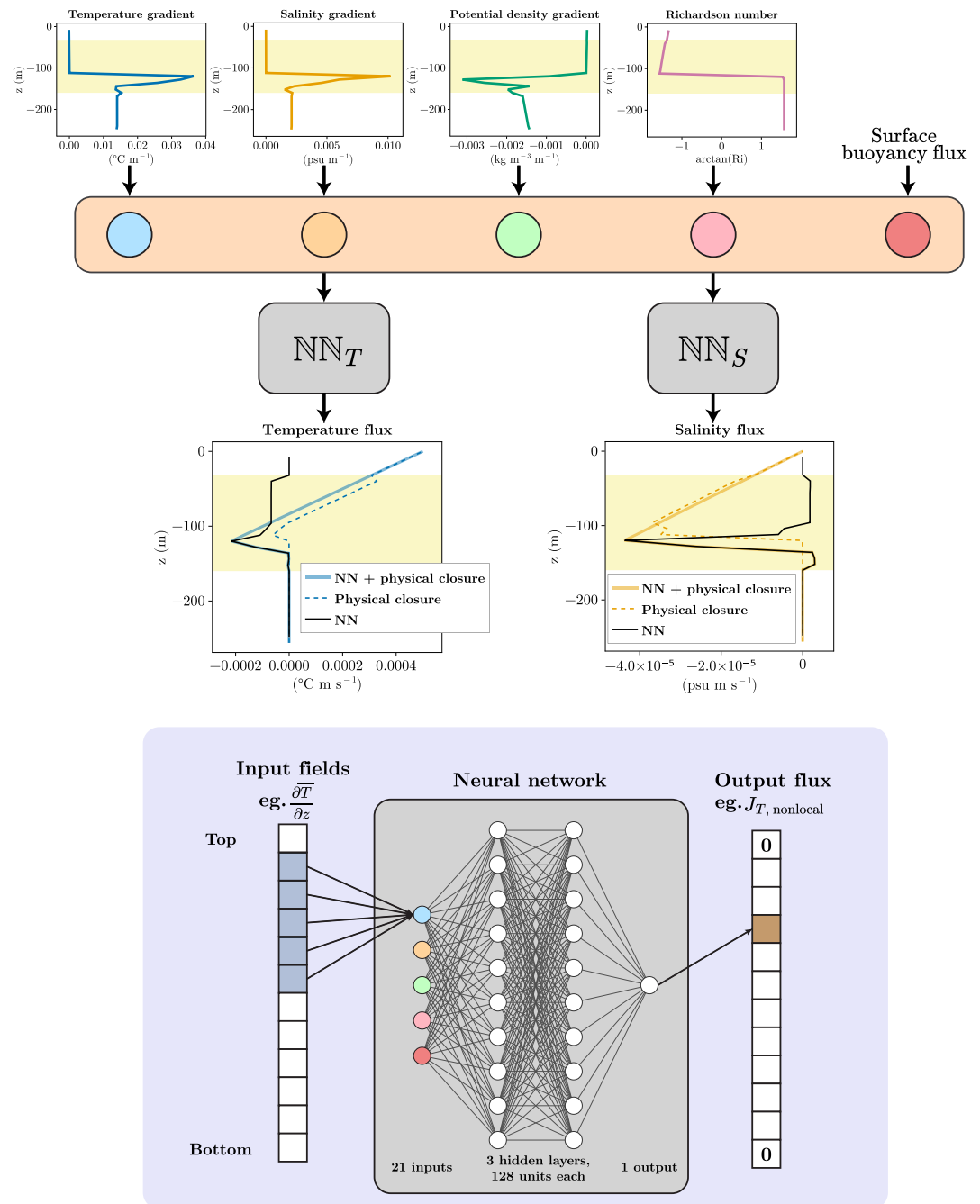


Figure E1. Schematic of NORi's neural network architecture, augmenting the base closure to capture missing entrainment fluxes. Top panel: the inputs of the neural networks $NN_T: \mathbb{R}^{21} \rightarrow \mathbb{R}$ and $NN_S: \mathbb{R}^{21} \rightarrow \mathbb{R}$, as well as their outputs, which are the entrainment temperature and salinity fluxes $J_{T,\text{nonlocal}}^i, J_{S,\text{nonlocal}}^i$. The light yellow zones indicate the depth range where the neural networks are active. Beyond these zones, neural network fluxes are zero. The temperature and salinity output fluxes show the neural networks augmenting the base closure to capture entrainment. Bottom panel: neural network architecture. The neural networks each produce a single output: the residual flux at grid point i given input $\mathbf{h}^i \in \mathbb{R}^{21}$, consisting of tracer gradients at five grid points in the local neighborhood of grid point i and the surface buoyancy flux (see Appendix E). The neural networks do not predict fluxes at the top and bottom grid points.

2. The strength of entrainment mixing depends strongly on the ratio between local shear and background stratification at the base of the mixed layer, which is encoded by $1/Ri$. A strong stratification in the thermocline would reduce the extent of penetrative convection, in which convective plumes overshoot into the

- interior due to inertia, while stronger local shear leads to stronger plume “rollup,” which also reduces the plume penetration. Information about local shear and stratification is encoded in the local Richardson number Ri .
3. The mixing rate due to entrainment depends on local tracer gradients regardless of their absolute values. To enforce this tracer invariance, the temperature, salinity, and potential density gradients $\frac{\partial \bar{T}}{\partial z}$, $\frac{\partial \bar{S}}{\partial z}$, and $\frac{\partial \bar{\sigma}}{\partial z}$, as well as the Richardson number Ri , are used as inputs to the neural networks instead of $\bar{u}$, $\bar{v}$, $\bar{T}$, $\bar{S}$, and $\bar{\sigma}$, since their actual values do not play a direct role in mixing dynamics.
 4. NORi is designed for the nonlinear equation of state TEOS-10 (Roquet et al., 2015). Therefore, the relationship between the temperature and salinity gradients on the one hand and the stratification on the other depends on the actual values of temperature and salinity. To encode information on nonlinearities in the equation of state explicitly to the neural networks, the potential density gradient $\frac{\partial \bar{\sigma}}{\partial z}$ is also provided.
 5. Entrainment can be thought of in terms of the interaction between nonlocal buoyancy fluxes and local tracer gradients. The mixing occurs only locally due to a nonlocal flux source. Therefore, tracer gradients are provided locally, while the surface buoyancy flux is provided nonlocally to the neural networks.

The weights of $\mathbb{NN}_T$ and $\mathbb{NN}_S$ are shared across all grid points. The local gradient Richardson number is provided as input to the neural networks in the form of $\arctan(Ri)$ so that it is bounded from above and below by $\pm \frac{\pi}{2}$, with appreciable gradient in the $|Ri| \lesssim 1$ regime, where turbulent dynamics are most sensitive to shear. The “entrainment zone” where the neural network is active is diagnosed from the diffusivity field κ by

$$i_\kappa = \min\{i \in \{1, 2, \dots, N+1\} : \kappa[i] = \kappa_0\}, \quad (E4)$$

$$i_{\min} = \max\{i_\kappa - 10, 2\}, \quad (E5)$$

$$i_{\max} = \min\{i_\kappa + 5, N\}, \quad (E6)$$

where i_κ is the grid point closest to the ocean surface where the base closure diffusivity κ is equal to the background diffusivity κ_0 . This is equivalent to the nearest point from the surface where $Ri \geq Ri^c$, with Ri^c defined by the base closure. The neural networks do not predict turbulent fluxes at the grid point at the ocean surface ($i = 1$) and at the bottom ($i = N + 1$). At these boundary points, the temperature and salinity fluxes are given by the boundary conditions. This restriction of the prediction zone of the neural networks reduces inference time, but implicitly assumes that convective plumes cannot penetrate deeper than 40 m below the BL base, which has been the case in all the LES generated. Figure E1 illustrates a schematic of the neural network architecture.

This design enforces some important physical principles:

1. The top and bottom fluxes are prescribed to satisfy the boundary conditions. This enforces tracer conservation as a hard constraint; NORi can only redistribute tracers in the interior.
2. In NORi, the local neighborhood visible to the neural networks as inputs spans 48 m, as we have found this to be a vertical extent that is sufficient for the neural networks to characterize the location of inference relative to the BL base while ensuring generalizability across different locations within the vertical column.
3. Entrainment has local effects: it enhances mixing in a small neighborhood near the base of the BL as it brings denser fluid from the ocean interior into the mixed layer. Therefore, the neural networks only need to know the information around their neighborhood to determine if entrainment occurs and produce a local flux to represent any mixing.

Appendix F: PDE Discretization of the 1D Column Model

To solve the partial differential Equations 11–14 numerically, we discretize the equations spatially and temporally. The spatial dimensions are discretized using the finite volume method, while we use a split implicit-explicit (IMEX) method for the temporal discretization. The diffusion terms are time stepped with the implicit backward Euler method, while the advective terms ($f\bar{u}$ and $f\bar{v}$) and neural network fluxes are time stepped with the explicit forward Euler method. The discretized equations are given by

$$\frac{\bar{u}^{n+1} - \bar{u}^n}{\Delta t} = \partial_z(\nu \partial_z \bar{u}^{n+1}) + f\bar{v}^n, \quad (F1)$$

$$\frac{\bar{v}^{n+1} - \bar{v}^n}{\Delta t} = \partial_z (\nu \partial_z \bar{v}^{n+1}) - f \bar{u}^n, \quad (\text{F2})$$

$$\frac{\bar{T}^{n+1} - \bar{T}^n}{\Delta t} = \partial_z (\kappa \partial_z \bar{T}^{n+1}) - \partial_z J_{\text{NN}_T}^n, \quad (\text{F3})$$

$$\frac{\bar{S}^{n+1} - \bar{S}^n}{\Delta t} = \partial_z (\kappa \partial_z \bar{S}^{n+1}) - \partial_z J_{\text{NN}_S}^n, \quad (\text{F4})$$

where $n = 0, 1, \dots$ indexes discrete time steps with $t^{n+1} - t^n = \Delta t$ and ∂_z is the discrete gradient operator in z .

Appendix G: Variable Normalization and Model Nondimensionalization

To promote better training behavior of neural networks and to allow the loss function to combine quantities with different units, we normalize the field and flux variables using a Z-score normalization such that the variables in the training suite have a zero mean and unit variance. This approach ensures that all features are treated equally during neural network training. The Z-score normalization is given by

$$\bar{u}^* = \frac{\bar{u} - \mu_{\bar{u}}}{\sigma_{\bar{u}}}, \quad (\text{G1})$$

$$\bar{v}^* = \frac{\bar{v} - \mu_{\bar{v}}}{\sigma_{\bar{v}}}, \quad (\text{G2})$$

$$\bar{T}^* = \frac{\bar{T} - \mu_{\bar{T}}}{\sigma_{\bar{T}}}, \quad (\text{G3})$$

$$\bar{S}^* = \frac{\bar{S} - \mu_{\bar{S}}}{\sigma_{\bar{S}}}, \quad (\text{G4})$$

$$\overline{u'w'}^* = \frac{\overline{u'w'} - \mu_{\overline{u'w'}}}{\sigma_{\overline{u'w'}}}, \quad (\text{G5})$$

$$\overline{v'w'}^* = \frac{\overline{v'w'} - \mu_{\overline{v'w'}}}{\sigma_{\overline{v'w'}}}, \quad (\text{G6})$$

$$\overline{w'T'}^* = \frac{\overline{w'T'} - \mu_{\overline{w'T'}}}{\sigma_{\overline{w'T'}}}, \quad (\text{G7})$$

$$\overline{w'S'}^* = \frac{\overline{w'S'} - \mu_{\overline{w'S'}}}{\sigma_{\overline{w'S'}}}, \quad (\text{G8})$$

where μ_{ϕ} and σ_{ϕ} are the mean and standard deviation of the field ϕ computed across all training simulations and in both space and time. Using the normalization above, starred quantities ϕ^* are nondimensional. We also nondimensionalize time and space using

$$t^* = \frac{t}{\tau}, \quad (\text{G9})$$

$$z^* = \frac{z}{H}, \quad (\text{G10})$$

where τ is the training horizon for each simulation, while H is the total depth of the domain. Substituting the nondimensional variables above and decomposing the fluxes into local (base-closure) and nonlocal (neural-network) components, the equations solved during training of the NORi closure are

$$\frac{\partial \bar{u}^*}{\partial t^*} = \frac{\tau}{H^2} \frac{\partial}{\partial z^*} \left(\nu \frac{\partial \bar{u}^*}{\partial z^*} \right) + \frac{f\tau}{\sigma_{\bar{u}}} (\sigma_{\bar{v}} \bar{v}^* + \mu_{\bar{v}}), \quad (\text{G11})$$

$$\frac{\partial \bar{v}^*}{\partial t^*} = \frac{\tau}{H^2} \frac{\partial}{\partial z^*} \left(\nu \frac{\partial \bar{v}^*}{\partial z^*} \right) - \frac{f\tau}{\sigma_{\bar{v}}} (\sigma_{\bar{u}} \bar{u}^* + \mu_{\bar{u}}), \quad (\text{G12})$$

$$\frac{\partial \bar{T}^*}{\partial t^*} = \frac{\tau}{H^2} \frac{\partial}{\partial z^*} \left(\kappa \frac{\partial \bar{T}^*}{\partial z^*} \right) - \frac{\tau}{H} \frac{\sigma_{w'T'}}{\sigma_{\bar{T}}} \frac{\partial}{\partial z^*} J_{\text{NN}_T}^*, \quad (\text{G13})$$

$$\frac{\partial \bar{S}^*}{\partial t^*} = \frac{\tau}{H^2} \frac{\partial}{\partial z^*} \left(\kappa \frac{\partial \bar{S}^*}{\partial z^*} \right) - \frac{\tau}{H} \frac{\sigma_{w'S'}}{\sigma_{\bar{S}}} \frac{\partial}{\partial z^*} J_{\text{NN}_S}^*. \quad (\text{G14})$$

Here, $J_{\text{NN}_T}^*$ and $J_{\text{NN}_S}^*$ denote the normalized neural-network flux outputs.

Appendix H: Stratification Comparison With Ocean Weather Station Papa

In our comparison of the column model solutions forced with realistic conditions, we find that NORi, its unaugmented base closure, the $k - \epsilon$ closure of Umlauf and Burchard (2003), and the CATKE closure of Wagner, Hillier, et al. (2025) all produce a comparable stratification evolution when compared against OWS Papa observations. Figure H1 shows the evolution of temperature, salinity, and buoyancy in OWS Papa observations, the model-observation differences in these fields, as well as the vertically averaged temperature and salinity drifts between the models and observations. In general, all models mentioned above capture the evolution of the depth of the mixed layer against observations, as the temperature differences in the mixed layer are on the order of 0.25°C, while the salinity differences in the mixed layer are on the order of 0.05 psu. NORi and its base closure produce fields that are highly similar, indicating that nonlocal entrainment is weak at OWS Papa. NORi, its base closure, and CATKE generally underestimate the mixed layer depth slightly, while $k - \epsilon$ produces a deeper mixed layer. However, these differences are small and the overall skill of all models is high.

The constant flux adjustment imposed on the ocean surface in addition to the prescribed surface fluxes given by the OWS Papa observations minimizes the temperature and salinity drifts, which can be seen from the small values of the average drifts in the bottom row of Figure H1. The temperature has a maximum drift of around 0.1°C, while the salinity has a maximum drift of around 0.02 psu. Furthermore, most of the drift originates in the deep ocean, where lateral advection—which the 1D column model cannot represent—modifies the temperature and salinity budgets.

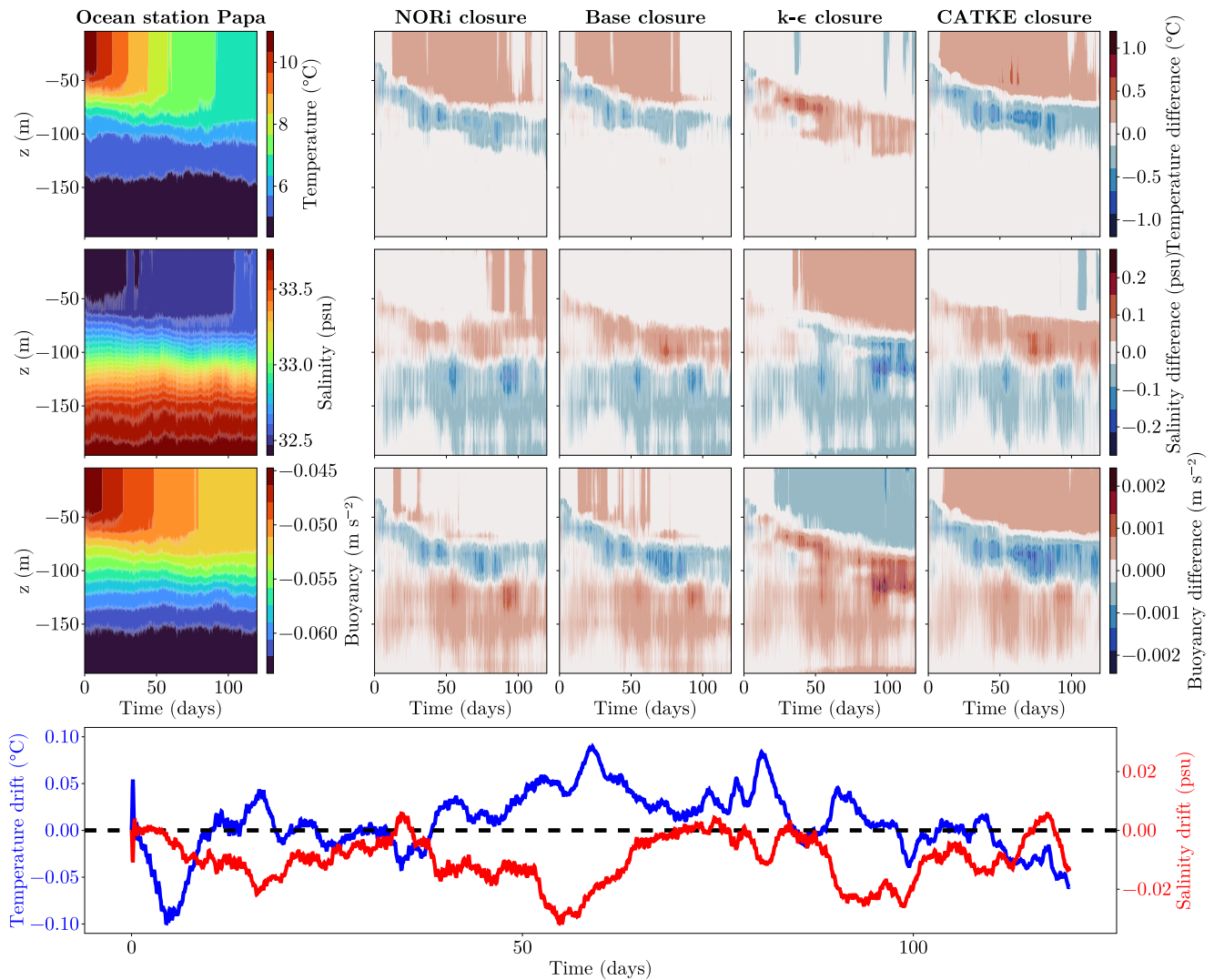


Figure H1. Left column, from top to bottom: Evolution of temperature, salinity, and buoyancy in OWS Papa observations. Right columns, from left to right: The temperature, salinity, and buoyancy differences between solutions of NORi, its unaugmented base closure, the $k - \epsilon$ closure of Umlauf and Burchard (2003), and the CATKE closure of Wagner, Hillier, et al. (2025) in the OWS Papa simulation. The differences are computed as model solutions minus observations. All contour plots are 8-year composites, taken from years with no large gaps in the OWS Papa data. Bottom row: The composite vertically averaged temperature and salinity drift between the models and observations. Since all models are forced with the same surface fluxes, their drifts are identical. The drifts shown are averaged over a running 24-hr window to smooth out high-frequency variability.

Appendix I: Double-Gyre Solution at 100 years

Here we show a snapshot of the zonal average of the double-gyre simulation at $t = 100$ years to complement Figure 13 in the main text. At $t = 100$ years, the double-gyre simulations begin to show significant differences between NORi, its unaugmented base closure, and $k - \epsilon$ closure. Figure 11 shows that, compared to $k - \epsilon$ and the base closure, NORi produces deeper mixed layers along the entire meridional extent of the basin, except in the northernmost regions where the deep mixed layers reach the bottom of the domain. However, we do not have a “ground truth” solution as a reference since this is an idealized setup. Therefore, further quantification of each model’s skill requires running realistic ocean simulations, where observational data are available for comparison. However, this is beyond the present scope, and we aim to provide a thorough and systematic evaluation of various closures in realistic ocean simulations in the future.

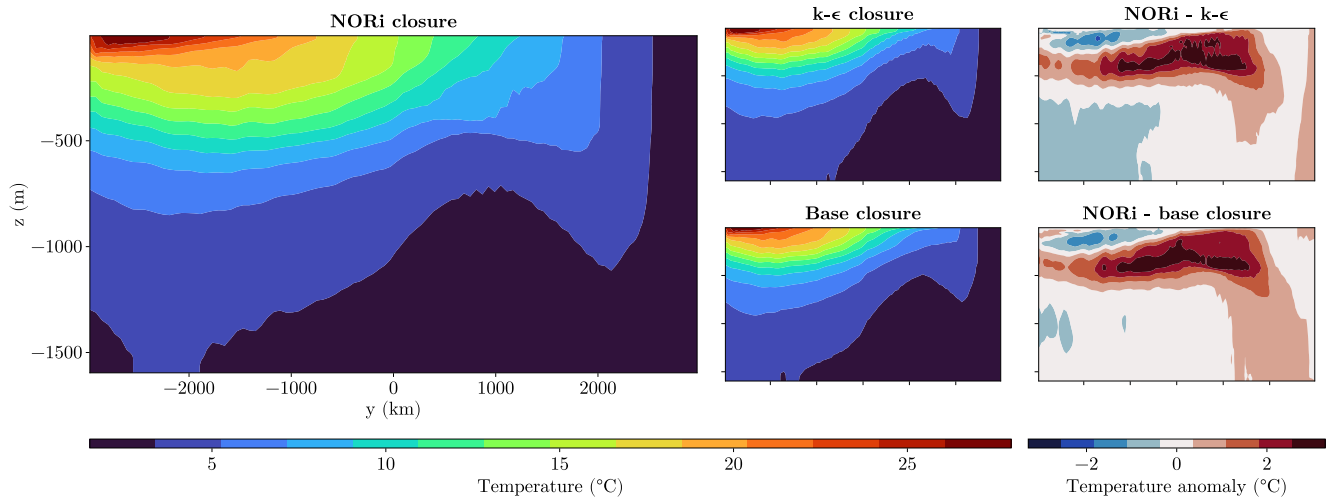


Figure 11. Zonally averaged temperature stratification comparison between NORi, the $k - \epsilon$ closure of Umlauf and Burchard (2003), and the unaugmented base closure in the double-gyre simulation. The snapshots are taken at $t = 100$ years. At this time, the simulation is not yet fully equilibrated. The rightmost column shows the temperature differences in stratification between NORi and the other two closures ($k - \epsilon$ and the unaugmented base closure).

Appendix J: NORi Solution Comparison With CATKE

In this section, we show results comparing NORi and CATKE (Wagner, Hillier, et al., 2025), a physics-based BL parameterization that uses a diagnostic mixing length with a convective adjustment component and a prognostic TKE. CATKE demonstrated superior performance compared to commonly used BL parameterizations such as KPP (Large et al., 1994) and the SMC-LT closure of Harcourt (2015) when benchmarked against idealized and realistic LES (Wagner, Hillier, et al., 2025).

Before making any comparisons, one important caveat is that CATKE is trained with an LES suite that takes into account the effects of additional mixing due to surface waves under shear-driven mixing scenarios, that is, Langmuir turbulence. In addition, CATKE and NORi are calibrated using different training suites with LES of different background stratification, surface forcings, equations of state, and rotation rates. Under the same surface wind stress and initial conditions, LES that take into account wave effects generally produce deeper mixed layers. We expect CATKE to produce deeper mixed layers than NORi in strongly wind-driven scenarios, since NORi is trained on simulations that do not take into account wave effects. This is because our goal in developing NORi is to illustrate a new design and training paradigm for data-driven ocean modeling, and we have chosen to demonstrate it using a reduced set of physics that capture the leading-order effects of vertical mixing due to winds and convection. Although the specific physical details of the training data are crucial, the NORi framework can be flexibly extended to incorporate other microturbulence physics through fine-tuning and/or transfer learning. Using this approach of augmenting a robust and simple physical closure with neural network components trained end-to-end, future versions of NORi can account for additional physical processes such as wave effects and rotation-dependent deep convection. They can also solve additional prognostic equations for turbulent quantities.

As expected, CATKE produces deeper mixed layers in single-column training and validation cases driven by strong winds. In Figure J1, the first and fourth columns are cases of strong wind, where CATKE produces slightly deeper mixed layers than NORi. As noted earlier, the LES solutions shown in Figure J1 do not take into account wave effects; therefore, it is not surprising that they are generally shallower than CATKE's predictions. Discrepancies between CATKE and NORi are expected given the differences in their training data.

When tested in realistic scenarios based on OWS Papa observations, NORi and CATKE produce very similar results as shown in Figure J2. CATKE produces slightly weaker mixing, as indicated by its shallower mixed layer depth. However, these differences are very small as both CATKE and NORi agree well with the observed profiles.

In the double-gyre simulation, the differences between CATKE and NORi are similar to those between $k - \epsilon$ and NORi (see Figures 11 and 13). At 2.5 years, as shown by the zonally averaged temperature in Figure J3, NORi produces shallower mixed layers than CATKE in the northern regions of the simulation. Interestingly, NORi also

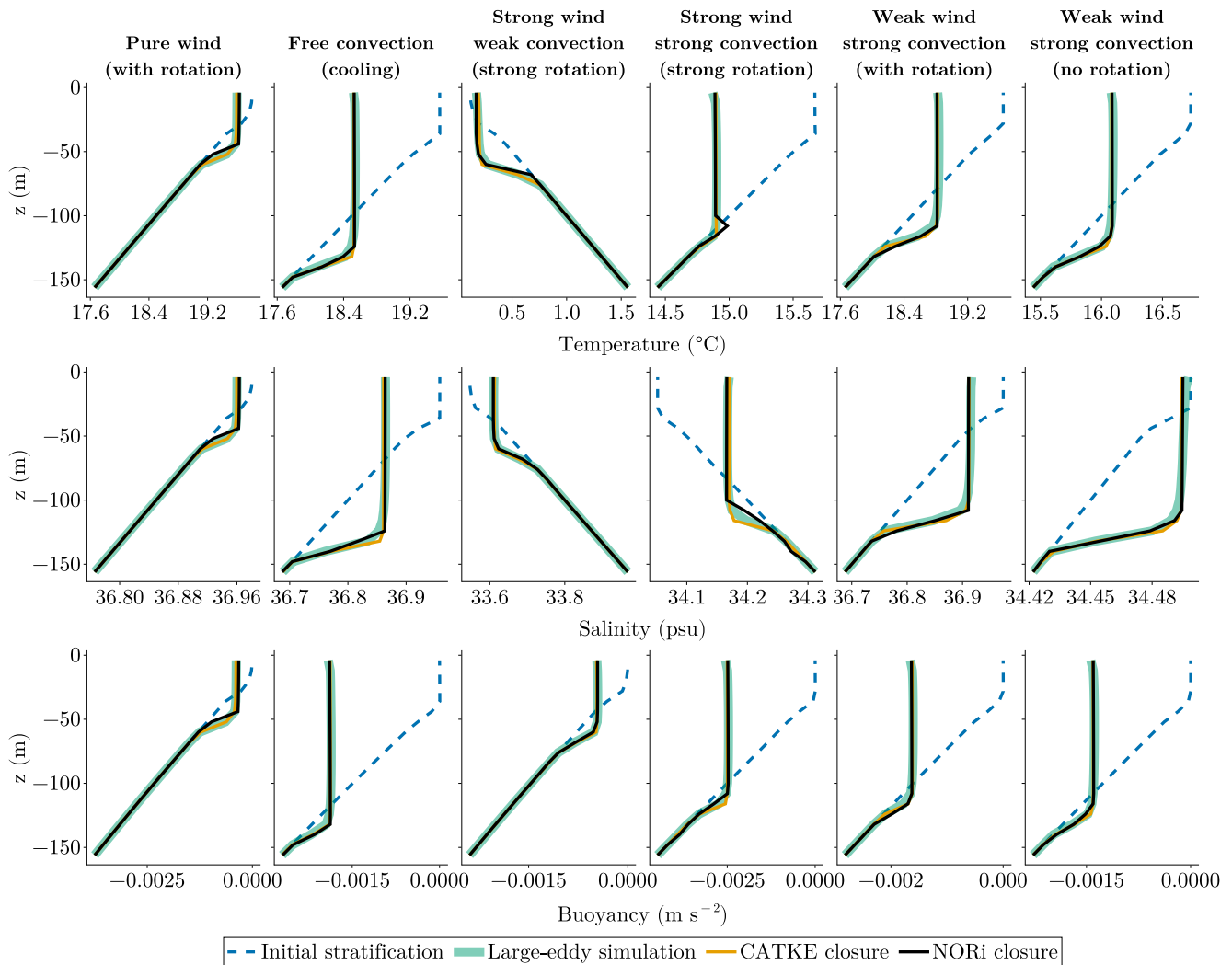


Figure J1. Comparison of NORi and CATKE performance in the column-model setting. CATKE (Wagner, Hillier, et al., 2025) is a physics-based boundary layer parameterization that uses a diagnostic mixing length with a convective adjustment component and a prognostic turbulent kinetic energy. We compare temperature, salinity, and buoyancy fields against large-eddy simulation solutions. All examples shown are validation cases which were not seen by NORi during training.

produces slightly deeper mixed layers in the southern part of the domain. At $t = 100$ years, we can see from Figure J4 that NORi generates mixed layers that are much deeper than CATKE in most regions, except in the northern part of the basin, where deep mixed layers reach the bottom, similar to the comparison with $k - \epsilon$. As we do not have a ground truth solution for this idealized setup, a thorough evaluation of NORi and CATKE would require running realistic regional/global ocean simulations where observational data are available for comparison. However, it is not surprising that CATKE and $k - \epsilon$ behave similarly, as CATKE is heavily inspired by two-equation closures such as $k - \epsilon$.

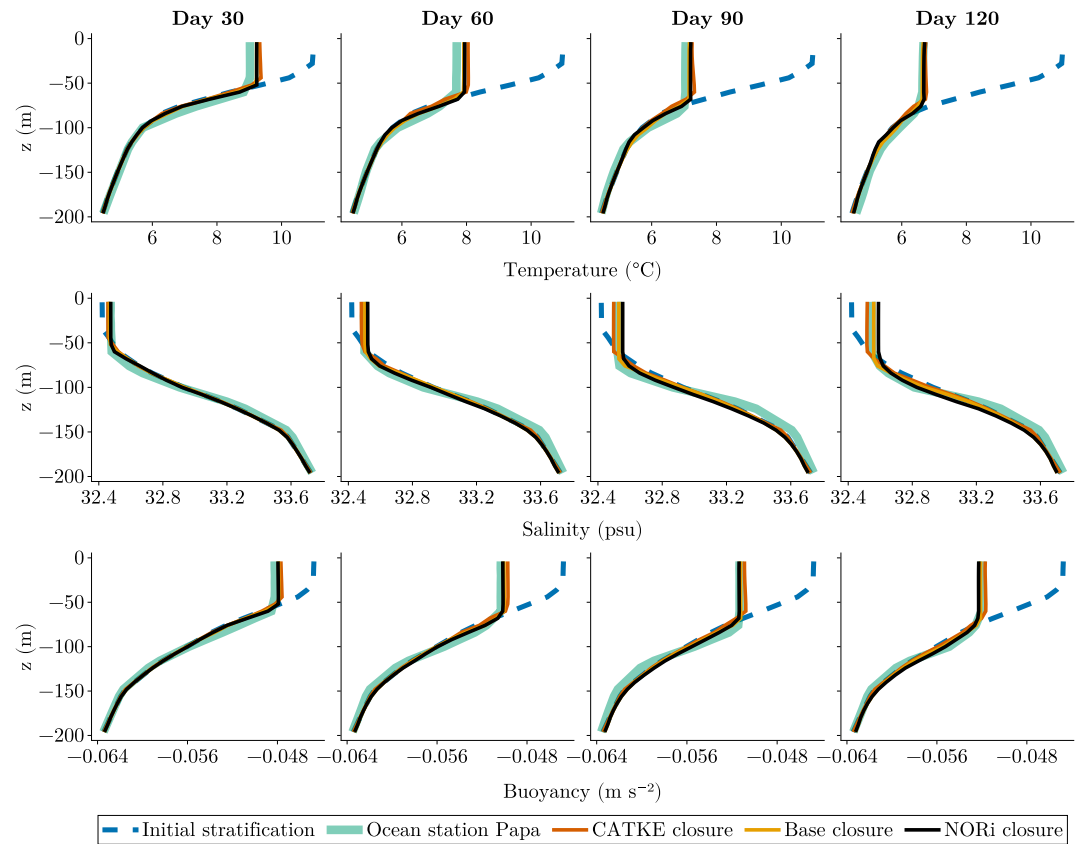


Figure J2. Monthly comparison of column-model simulations from NORi, its base closure, and CATKE against OWS Papa observations, run with prescribed surface fluxes given by OWS Papa observations. The simulations are initialized using the observed temperature and salinity profiles on 1 November and run for 120 days. The profiles shown are composites averaged over 8 different years between 2007 and 2023. Some years are excluded because there are large gaps (more than 72 hr) within the data. A constant flux correction is applied on top of the prescribed fluxes to mitigate the drifts in the model solution due to the discrepancies between the observed temperature and salinity budgets and those given by the prescribed fluxes. All models are forced with identical surface fluxes.

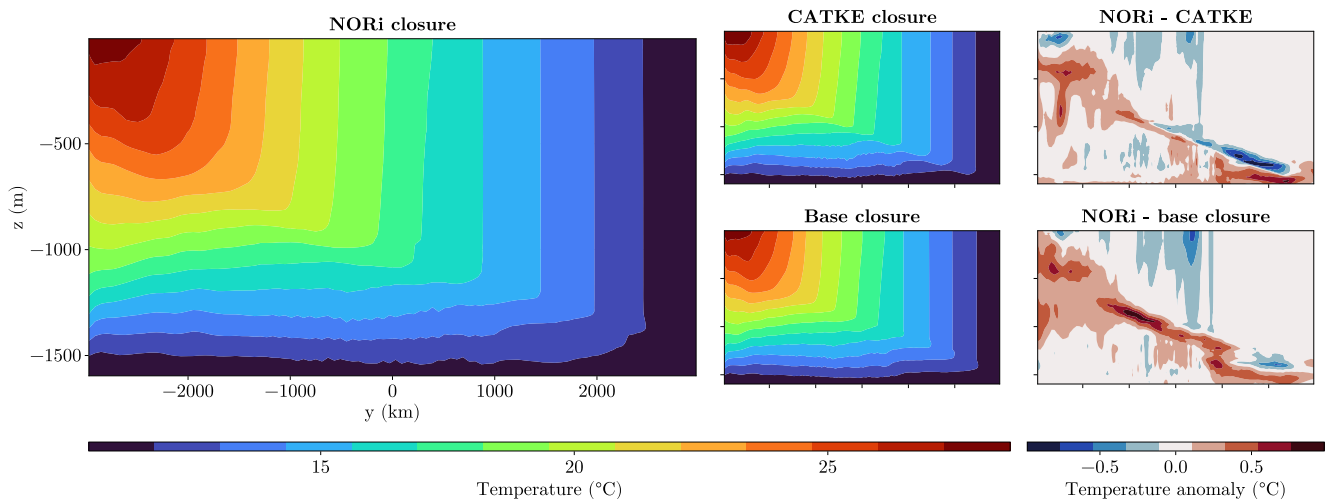


Figure J3. Zonally averaged temperature stratification comparison between NORi, CATKE (Wagner, Hillier, et al., 2025), and the unaugmented base closure in the double-gyre simulation. The snapshots are taken at $t = 2.5$ years, when the mixed layer extends to almost the entire water column in the north. The rightmost column shows the temperature differences in stratification between NORi and the other two closures (CATKE and the unaugmented base closure).

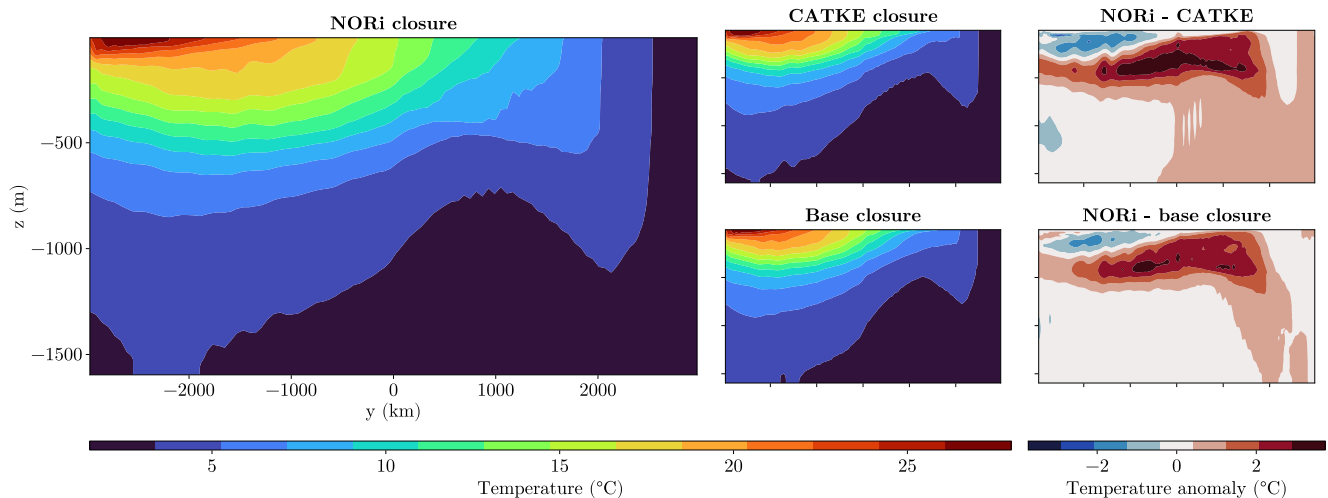


Figure J4. Same as Figure J3, but at $t = 100$ years.

AI Tools Disclosure

Artificial intelligence tools, including large language models and code-completion tools offered by OpenAI, Anthropic, and GitHub Copilot, were used to assist with code development, manuscript editing, and proofreading. All AI-generated suggestions were thoroughly reviewed and verified by the authors.

Conflict of Interest

The authors declare no conflicts of interest relevant to this study.

Availability Statement

The LES data used for the training and validation of NORi are generated from open source software Oceananigans.jl (Ramadhan et al., 2020; Wagner, Silvestri, et al., 2025). These LES data are packaged into a publicly available data set SOBLLES—A Salty Ocean Boundary Layer Large-Eddy Simulation Data set hosted on Zenodo (Lee, 2025a). The base closure in NORi is calibrated using open source software EnsembleKalmanProcesses.jl (Dunbar et al., 2022). The neural networks in NORi are implemented using open source software Lux.jl (Pal, 2023) and calibrated using Enzyme.jl (Moses & Churavy, 2020). Scripts for the training and validation of NORi, as well as its implementation on Oceananigans.jl, together with scripts to reproduce all figures in this document, are available in the GitHub repository NORiOceanParameterization.jl (Lee, 2025b). The data required to reproduce all figures in this paper are available through a data companion to NORiOceanParameterization.jl hosted on Zenodo (Lee, 2026).

References

- Balsara, D. S., & Shu, C.-W. (2000). Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy. *Journal of Computational Physics*, 160(2), 405–452. <https://doi.org/10.1006/jcph.2000.6443>
- Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1), 65–98. <https://doi.org/10.1137/141000671>
- Bi, K., Xie, L., Zhang, H., Chen, X., Gu, X., & Tian, Q. (2023). Accurate medium-range global weather forecasting with 3D neural networks. *Nature*, 619(7970), 533–538. <https://doi.org/10.1038/s41586-023-06185-3>
- Boccaletti, G., Ferrari, R., & Fox-Kemper, B. (2007). Mixed layer instabilities and restratification. *Journal of Physical Oceanography*, 37(9), 2228–2250. <https://doi.org/10.1175/JPO3101.1>
- Bolton, T., & Zanna, L. (2019). Applications of deep learning to ocean data inference and subgrid parameterization. *Journal of Advances in Modeling Earth Systems*, 11(1), 376–399. <https://doi.org/10.1029/2018MS001472>
- Burchard, H., & Bolding, K. (2001). Comparative analysis of four second-moment turbulence closure models for the oceanic mixed layer. *Journal of Physical Oceanography*, 31(8), 1943–1968. [https://doi.org/10.1175/1520-0485\(2001\)031<1943:CAOFMS>2.0.CO;2](https://doi.org/10.1175/1520-0485(2001)031<1943:CAOFMS>2.0.CO;2)
- Burchard, H., Deleersnijder, E., & Meister, A. (2003). A high-order conservative patankar-type discretisation for stiff systems of production–destruction equations. *Applied Numerical Mathematics*, 47(1), 1–30. [https://doi.org/10.1016/S0168-9274\(03\)00101-6](https://doi.org/10.1016/S0168-9274(03)00101-6)
- Canuto, V. M., Howard, A., Cheng, Y., & Dubovikov, M. S. (2001). Ocean turbulence. Part I: One-point closure model—Momentum and heat vertical diffusivities. *Journal of Physical Oceanography*, 31(6), 1413–1426. [https://doi.org/10.1175/1520-0485\(2001\)031<1413:otpio>2.0.co;2](https://doi.org/10.1175/1520-0485(2001)031<1413:otpio>2.0.co;2)

Acknowledgments

The authors thank Aakash Sane and the two other anonymous reviewers for their insightful comments and constructive suggestions that helped strengthen this work. This project is supported by Schmidt Sciences, LLC, and National Science Foundation award AGS-1835576. X. K. L. was also partly supported by the Norman C. Rasmussen Fellowship and the Callahan-Dee Fellowship. The work was initiated while X. K. L. was an undergraduate student in the Department of Physics at Imperial College London under the Imperial-MIT student exchange program, working with A. R. and J. M. Preliminary results are reported in A. R.'s PhD thesis (Ramadhan, 2023).

- Carson, D. J. (1973). The development of a dry inversion-capped convectively unstable boundary layer. *Quarterly Journal of the Royal Meteorological Society*, 99(421), 450–467. <https://doi.org/10.1002/qj.49709942105>
- Charnock, H. (1955). Wind stress on a water surface. *Quarterly Journal of the Royal Meteorological Society*, 81(350), 639–640. <https://doi.org/10.1002/qj.49708135027>
- Chattopadhyay, A., Sun, Y. Q., & Hassanzadeh, P. (2023). Challenges of learning multi-scale dynamics with AI weather models: Implications for stability and one solution. *arXiv*. <https://doi.org/10.48550/ARXIV.2304.07029>
- Chen, R. T. Q., Rubanova, Y., Bettencourt, J., & Duvenaud, D. (2019). Neural ordinary differential equations. *arXiv*. <https://doi.org/10.48550/arXiv.1806.07366>
- Chen, Z., Wenegrat, J., Chor, T., & Marchesiello, P. (2025). Evaluating turbulence parameterizations at gray zone resolutions for the ocean surface boundary layer. *Journal of Advances in Modeling Earth Systems*, 17(10), e2025MS005104. <https://doi.org/10.1029/2025MS005104>
- Copernicus Climate Change Service. (2019). ERA5 monthly averaged data on single levels from 1940 to present. *Copernicus Climate Change Service (C3S) Climate Data Store (CDS)*. <https://doi.org/10.24381/CDS.F17050D7>
- Craig, P. D., & Banner, M. L. (1994). Modeling wave-enhanced turbulence in the ocean surface layer. *Journal of Physical Oceanography*, 24(12), 2546–2559. [https://doi.org/10.1175/1520-0485\(1994\)024<2546:MWETIT>2.0.CO;2](https://doi.org/10.1175/1520-0485(1994)024<2546:MWETIT>2.0.CO;2)
- Cronin, M., Anderson, N., Zhang, D., Berk, P., Wills, S., Serra, Y., et al. (2023). PMEL ocean climate stations as reference time series and research aggregate devices. *Oceanography*. <https://doi.org/10.5670/oceanog.2023.224>
- Cronin, M. F., Fairall, C. W., & McPhaden, M. J. (2006). An assessment of buoy-derived and numerical weather prediction surface heat fluxes in the tropical Pacific. *Journal of Geophysical Research*, 111(C6), 2005JC003324. <https://doi.org/10.1029/2005JC003324>
- Deardorff, J. W. (1972). Theoretical expression for the countergradient vertical heat flux. *Journal of Geophysical Research*, 77(30), 5900–5904. <https://doi.org/10.1029/JC077i030p05900>
- Deardorff, J. W., Willis, G. E., & Stockton, B. H. (1980). Laboratory studies of the entrainment zone of a convectively mixed layer. *Journal of Fluid Mechanics*, 100(1), 41–64. <https://doi.org/10.1017/S0022112080001000>
- Drazin, P. G., & Reid, W. H. (2004). *Hydrodynamic stability* (2nd ed.). Cambridge University Press. <https://doi.org/10.1017/CBO9780511616938>
- Dunbar, O. R. A., Lopez-Gomez, I., Garbuno-Inigo, A., Huang, D. Z., Bach, E., & Wu, J.-I. (2022). EnsembleKalmanProcesses.jl: Derivative-free ensemble-based model calibration. *Journal of Open Source Software*, 7(80), 4869. <https://doi.org/10.21105/joss.04869>
- DuVivier, A. K., Large, W. G., & Small, R. J. (2018). Argo observations of the deep mixing band in the Southern Ocean: A salinity modeling challenge. *Journal of Geophysical Research: Oceans*, 123(10), 7599–7617. <https://doi.org/10.1029/2018JC014275>
- Fairall, C. W., Bradley, E. F., Hare, J. E., Grachev, A. A., & Edson, J. B. (2003). Bulk parameterization of air–sea fluxes: Updates and verification for the COARE algorithm. *Journal of Climate*, 16(4), 571–591. [https://doi.org/10.1175/1520-0442\(2003\)016<0571:BPOASF>2.0.CO;2](https://doi.org/10.1175/1520-0442(2003)016<0571:BPOASF>2.0.CO;2)
- Fan, X., Fox-Kemper, B., Suzuki, N., Li, Q., Marchesiello, P., Sullivan, P. P., & Hall, P. S. (2024). Comparison of the Coastal and Regional Ocean Community model (CROCO) and NCAR-LES in non-hydrostatic simulations. *Geoscientific Model Development*, 17(10), 4095–4113. <https://doi.org/10.5194/gmd-17-4095-2024>
- Fox-Kemper, B., Ferrari, R., & Hallberg, R. (2008). Parameterization of mixed layer eddies. Part I: Theory and diagnosis. *Journal of Physical Oceanography*, 38(6), 1145–1165. <https://doi.org/10.1175/2007JPO3792.1>
- Frezat, H., Le Sommer, J., Fablet, R., Balarac, G., & Lguensat, R. (2022). A posteriori learning for quasi-geostrophic turbulence parametrization. *Journal of Advances in Modeling Earth Systems*, 14(11), e2022MS003124. <https://doi.org/10.1029/2022MS003124>
- Frostig, R., Johnson, M. J., & Leary, C. (2018). Compiling machine learning programs via high-level tracing. Retrieved from <https://mlsys.org/Conferences/doc/2018/146.pdf>
- Gaillard, F., Reynaud, T., Thierry, V., Kolodziejczyk, N., & Von Schuckmann, K. (2016). In situ–based reanalysis of the global ocean temperature and salinity with ISAS: Variability of the heat content and steric height. *Journal of Climate*, 29(4), 1305–1323. <https://doi.org/10.1175/JCLI-D-15-0028.1>
- Galperin, B., Kantha, L. H., Hassid, S., & Rosati, A. (1988). A quasi-equilibrium turbulent energy model for geophysical flows. *Journal of the Atmospheric Sciences*, 45(1), 55–62. [https://doi.org/10.1175/1520-0469\(1988\)045<0055:AQETEM>2.0.CO;2](https://doi.org/10.1175/1520-0469(1988)045<0055:AQETEM>2.0.CO;2)
- Gent, P. R., & McWilliams, J. C. (1990). Isopycnal mixing in ocean circulation models. *Journal of Physical Oceanography*, 20(1), 150–155. [https://doi.org/10.1175/1520-0485\(1990\)020<0150:imicm>2.0.co;2](https://doi.org/10.1175/1520-0485(1990)020<0150:imicm>2.0.co;2)
- Gjini, R., Morzfeld, M., Dunbar, O. R. A., & Schneider, T. (2025). The ensemble Kalman inversion race. *arXiv*. <https://doi.org/10.48550/ARXIV.2511.15853>
- Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 249–256). JMLR Workshop and Conference Proceedings.
- Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 315–323). JMLR Workshop and Conference Proceedings.
- Goyal, A., & Bengio, Y. (2022). Inductive biases for deep learning of higher-level cognition. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 478(2266), 20210068. <https://doi.org/10.1098/rspa.2021.0068>
- Grinstein, F. F., Margolin, L. G., & Rider, W. J. (Eds.). (2007). *Implicit large eddy simulation: Computing turbulent fluid dynamics* (1st ed.). Cambridge University Press. <https://doi.org/10.1017/CBO9780511618604>
- Harcourt, R. R. (2015). An improved second-moment closure model of Langmuir turbulence. *Journal of Physical Oceanography*, 45(1), 84–103. <https://doi.org/10.1175/JPO-D-14-0046.1>
- Hewitt, H. T., Roberts, M., Mathiot, P., Biastoch, A., Blockley, E., Chassignet, E. P., et al. (2020). Resolving and parameterising the ocean mesoscale in Earth system models. *Current Climate Change Reports*, 6(4), 137–152. <https://doi.org/10.1007/s40641-020-00164-w>
- Iglesias, M. A., Law, K. J. H., & Stuart, A. M. (2013). Ensemble Kalman methods for inverse problems. *Inverse Problems*, 29(4), 045001. <https://doi.org/10.1088/0266-5611/29/4/045001>
- Kang, S., & Constantinescu, E. M. (2023). Learning subgrid-scale models with neural ordinary differential equations. *Computers & Fluids*, 261, 105919. <https://doi.org/10.1016/j.compfluid.2023.105919>
- Kantha, L. H., & Clayson, C. A. (1994). An improved mixed layer model for geophysical applications. *Journal of Geophysical Research*, 99(C12), 25235–25266. <https://doi.org/10.1029/94JC02257>
- Kim, S., Ji, W., Deng, S., Ma, Y., & Rackauckas, C. (2021). Stiff neural ordinary differential equations. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(9), 093122. <https://doi.org/10.1063/5.0060697>
- Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization. *arXiv*. <https://doi.org/10.48550/arXiv.1412.6980>
- Kochkov, D., Yuval, J., Langmore, I., Norgaard, P., Smith, J., Mooers, G., et al. (2024). Neural general circulation models for weather and climate. *Nature*, 632(8027), 1060–1066. <https://doi.org/10.1038/s41586-024-07744-y>
- Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P., & Hoyer, S. (2021). Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21), e2101784118. <https://doi.org/10.1073/pnas.2101784118>

- Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirmsberger, P., Fortunato, M., Alet, F., et al. (2023). GraphCast: Learning skillful medium-range global weather forecasting. *arXiv*. <https://doi.org/10.48550/arXiv.2212.12794>
- Large, W. G., & Gent, P. R. (1999). Validation of vertical mixing in an equatorial ocean model using large eddy simulations and observations. *Journal of Physical Oceanography*, 29(3), 449–464. [https://doi.org/10.1175/1520-0485\(1999\)029<0449:vovmia>2.0.co;2](https://doi.org/10.1175/1520-0485(1999)029<0449:vovmia>2.0.co;2)
- Large, W. G., McWilliams, J. C., & Doney, S. C. (1994). Oceanic vertical mixing: A review and a model with a nonlocal boundary layer parameterization. *Reviews of Geophysics*, 32(4), 363–403. <https://doi.org/10.1029/94RG01872>
- Launder, B., & Spalding, D. (1974). The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 3(2), 269–289. [https://doi.org/10.1016/0045-7825\(74\)90029-2](https://doi.org/10.1016/0045-7825(74)90029-2)
- Lee, X. K. (2025a). SOBLLES: A salty ocean boundary layer large-eddy simulations dataset. *Zenodo*. <https://doi.org/10.5281/ZENODO.16278000>
- Lee, X. K. (2025b). xkykai/NORiOceanParameterization.jl: Initial release. *Zenodo*. <https://doi.org/10.5281/ZENODO.17773380>
- Lee, X. K. (2026). Data companion for NORiOceanParameterization.jl. *Zenodo*. <https://doi.org/10.5281/ZENODO.17605194>
- Legay, A., Deremble, B., & Burchard, H. (2025). Derivation and implementation of a non-local term to improve the Oceanic convection representation within the k-ε parameterization. *Journal of Advances in Modeling Earth Systems*, 17(1), e2024MS004243. <https://doi.org/10.1029/2024MS004243>
- Li, G., & Xie, S.-P. (2014). Tropical biases in CMIP5 multimodel ensemble: The excessive equatorial Pacific cold tongue and double ITCZ problems. *Journal of Climate*, 27(4), 1765–1780. <https://doi.org/10.1175/JCLI-D-13-00337.1>
- Liang, J.-H., Yuan, J., Wan, X., Liu, J., Liu, B., Jang, H., & Tyagi, M. (2022). Exploring the use of machine learning to parameterize vertical mixing in the ocean surface boundary layer. *Ocean Modelling*, 176, 102059. <https://doi.org/10.1016/j.ocemod.2022.102059>
- Marshall, J., & Schott, F. (1999). Open-ocean convection: Observations, theory, and models. *Reviews of Geophysics*, 37(1), 1–64. <https://doi.org/10.1029/98RG02739>
- McWilliams, J. C. (2006). *Fundamentals of geophysical fluid dynamics*. Cambridge University Press.
- Molod, A. (2009). Running GCM physics and dynamics on different grids: Algorithm and tests. *Tellus*, 61(3), 381–393. <https://doi.org/10.1111/j.1600-0870.2009.00394.x>
- Moses, W., Cheng, G., Churavy, V., Gelbrecht, M., Klöwer, M., Kump, J., et al. (2026). DJ4Earth: Differentiable and performance-portable Earth System Modeling via program transformations. *Journal of Advances in Modeling Earth Systems*, 18(5), e2025MS005615. <https://doi.org/10.1029/2025MS005615>
- Moses, W., & Churavy, V. (2020). Instead of rewriting foreign code for machine learning, automatically synthesize fast gradients. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (Vol. 33, pp. 12472–12485). Curran Associates, Inc. Retrieved from <https://proceedings.neurips.cc/paper/2020/file/9332c513ef44b682e9347822c2e457ac-Paper.pdf>
- Niiler, P. (1977). *One-dimensional models of the upper ocean, Modelling and Prediction of the Upper Layers of the Ocean* (pp. 143–172). Pergamon.
- Noh, Y., Cheon, W. G., Hong, S. Y., & Raasch, S. (2003). Improvement of the K-profile model for the planetary boundary layer based on large eddy simulation data. *Boundary-Layer Meteorology*, 107(2), 401–427. <https://doi.org/10.1023/A:1022146015946>
- Pacanowski, R., & Philander, S. (1981). Parameterization of vertical mixing in numerical models of tropical oceans. *Journal of Physical Oceanography*, 11(11), 1443–1451. [https://doi.org/10.1175/1520-0485\(1981\)011<1443:povmin>2.0.co;2](https://doi.org/10.1175/1520-0485(1981)011<1443:povmin>2.0.co;2)
- Pal, A. (2023). Lux: Explicit parameterization of deep neural networks in Julia [Object]. <https://doi.org/10.5281/ZENODO.7808904>
- Pathak, J., Subramanian, S., Harrington, P., Raja, S., Chattopadhyay, A., Mardani, M., et al. (2022). FourCastNet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv*. <https://doi.org/10.48550/arXiv.2202.11214>
- Pedlosky, J. (1996). *Ocean circulation theory*. Springer. <https://doi.org/10.1007/978-3-662-03204-6>
- Pressel, K. G., Mishra, S., Schneider, T., Kaul, C. M., & Tan, Z. (2017). Numerics and subgrid-scale modeling in large eddy simulations of stratocumulus clouds. *Journal of Advances in Modeling Earth Systems*, 9(2), 1342–1365. <https://doi.org/10.1002/2016MS000778>
- Price, J. F., Weller, R. A., & Pinkel, R. (1986). Diurnal cycling: Observations and models of the upper ocean response to diurnal heating, cooling, and wind mixing. *Journal of Geophysical Research*, 91(C7), 8411–8427. <https://doi.org/10.1029/JC091iC07p08411>
- Rackauckas, C., Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., et al. (2021). Universal differential equations for scientific machine learning. *arXiv*. <https://doi.org/10.48550/arXiv.2001.04385>
- Ramadhan, A. (2023). *Data-driven ocean modeling using neural differential equations (unpublished doctoral dissertation)*. Massachusetts Institute of Technology.
- Ramadhan, A., Wagner, G., Hill, C., Campin, J.-M., Churavy, V., Besard, T., et al. (2020). Oceananigans.jl: Fast and friendly geophysical fluid dynamics on GPUs. *Journal of Open Source Software*, 5(53), 2018. <https://doi.org/10.21105/joss.02018>
- Redi, M. H. (1982). Oceanic isopycnal mixing by coordinate rotation. *Journal of Physical Oceanography*, 12(10), 1154–1158. [https://doi.org/10.1175/1520-0485\(1982\)012<1154:oimbc>2.0.co;2](https://doi.org/10.1175/1520-0485(1982)012<1154:oimbc>2.0.co;2)
- Reffray, G., Bourdalle-Badie, R., & Calone, C. (2015). Modelling turbulent vertical mixing sensitivity using a 1-D version of NEMO. *Geoscientific Model Development*, 8(1), 69–86. <https://doi.org/10.5194/gmd-8-69-2015>
- Reichl, B. G., & Hallberg, R. (2018). A simplified energetics based planetary boundary layer (ePBL) approach for ocean climate simulations. *Ocean Modelling*, 132, 112–129. <https://doi.org/10.1016/j.ocemod.2018.10.004>
- Reuther, A., Kepner, J., Byun, C., Samsi, S., Arcand, W., Bestor, D., et al. (2018). Interactive supercomputing on 40,000 cores for machine learning and data analysis. In *2018 IEEE high performance extreme computing Conference (HPEC)* (pp. 1–6). IEEE. <https://doi.org/10.1109/HPEC.2018.8547629>
- Roquet, F., Madec, G., McDougall, T. J., & Barker, P. M. (2015). Accurate polynomial expressions for the density and specific volume of seawater using the TEOS-10 standard. *Ocean Modelling*, 90, 29–43. <https://doi.org/10.1016/j.ocemod.2015.04.002>
- Sallée, J., Shuckburgh, E., Bruneau, N., Meijers, A. J. S., Bracegirdle, T. J., & Wang, Z. (2013). Assessment of Southern Ocean mixed-layer depths in CMIP5 models: Historical bias and forcing response. *Journal of Geophysical Research: Oceans*, 118(4), 1845–1862. <https://doi.org/10.1002/jgrc.20157>
- Sane, A., Reichl, B. G., Adcroft, A., & Zanna, L. (2023). Parameterizing vertical mixing coefficients in the ocean surface boundary layer using neural networks. *Journal of Advances in Modeling Earth Systems*, 15(10), e2023MS003890. <https://doi.org/10.1029/2023MS003890>
- Sane, A., Reichl, B. G., Adcroft, A., & Zanna, L. (2026). Machine learned equations for vertical mixing coefficients in the ocean surface boundary layer. *Geophysical Research Letters*, 53(10), e2026GL122106. <https://doi.org/10.1029/2026GL122106>
- Schumann, U., & Sweet, R. A. (1988). Fast Fourier transforms for direct solution of Poisson's equation with staggered boundary conditions. *Journal of Computational Physics*, 75(1), 123–137. [https://doi.org/10.1016/0021-9991\(88\)90102-7](https://doi.org/10.1016/0021-9991(88)90102-7)

- Silvestri, S., Wagner, G. L., Campin, J., Constantinou, N. C., Hill, C. N., Souza, A., & Ferrari, R. (2024). A new WENO-based momentum advection scheme for simulations of ocean mesoscale turbulence. *Journal of Advances in Modeling Earth Systems*, 16(7), e2023MS004130. <https://doi.org/10.1029/2023MS004130>
- Silvestri, S., Wagner, G. L., Constantinou, N. C., Hill, C. N., Campin, J., Souza, A. N., et al. (2025). A GPU-based ocean dynamical core for routine mesoscale-resolving climate simulations. *Journal of Advances in Modeling Earth Systems*, 17(4), e2024MS004465. <https://doi.org/10.1029/2024MS004465>
- Sirignano, J., MacArt, J. F., & Freund, J. B. (2020). DPM: A deep learning PDE augmentation method with application to large-eddy simulation. *Journal of Computational Physics*, 423, 109811. <https://doi.org/10.1016/j.jcp.2020.109811>
- Smolarkiewicz, P. K., Margolin, L. G., & Wyszogrodzki, A. A. (2007). Implicit large-eddy simulation in meteorology: From boundary layers to climate. *Journal of Fluids Engineering*, 129(12), 1533–1539. <https://doi.org/10.1115/1.2801678>
- Souza, A. N., Wagner, G. L., Ramadhan, A., Allen, B., Churavy, V., Schloss, J., et al. (2020). Uncertainty quantification of ocean parameterizations: Application to the K-Profile-Parameterization for penetrative convection. *Journal of Advances in Modeling Earth Systems*, 12(12), e2020MS002108. <https://doi.org/10.1029/2020MS002108>
- Stachenfeld, K., Fielding, D. B., Kochkov, D., Cranmer, M., Pfaff, T., Godwin, J., et al. (2021). Learned coarse models for efficient turbulence simulation. *arXiv*. <https://doi.org/10.48550/ARXIV.2112.15275>
- Subel, A., Chattopadhyay, A., Guan, Y., & Hassanzadeh, P. (2021). Data-driven subgrid-scale modeling of forced burgers turbulence using deep learning with generalization to higher Reynolds numbers via transfer learning. *Physics of Fluids*, 33(3), 031702. <https://doi.org/10.1063/5.0040286>
- Taylor, J. R., & Ferrari, R. (2010). Buoyancy and wind-driven convection at mixed layer density fronts. *Journal of Physical Oceanography*, 40(6), 1222–1242. <https://doi.org/10.1175/2010JPO4365.1>
- Taylor, J. R., & Thompson, A. F. (2023). Submesoscale dynamics in the upper ocean. *Annual Review of Fluid Mechanics*, 55(1), 103–127. <https://doi.org/10.1146/annurev-fluid-031422-095147>
- The Climode Group, Marshall, J., Ferrari, R., Forget, G., Maze, G., Andersson, A., et al. (2009). The CLIMODE field campaign: Observing the cycle of convection and restratification over the Gulf Stream. *Bulletin of the American Meteorological Society*, 90(9), 1337–1350. <https://doi.org/10.1175/2009BAMS2706.1>
- Thomas, L., & Ferrari, R. (2008). Friction, frontogenesis, and the stratification of the surface mixed layer. *Journal of Physical Oceanography*, 38(11), 2501–2518. <https://doi.org/10.1175/2008JPO3797.1>
- Treguier, A. M., De Boyer Montégut, C., Bozec, A., Chassignet, E. P., Fox-Kemper, B., McC. Hogg, A., et al. (2023). The mixed-layer depth in the Ocean Model Intercomparison Project (OMIP): Impact of resolving mesoscale eddies. *Geoscientific Model Development*, 16(13), 3849–3872. <https://doi.org/10.5194/gmd-16-3849-2023>
- Umlauf, L., & Burchard, H. (2003). A generic length-scale equation for geophysical turbulence models. *Journal of Marine Research*, 61(2), 235–265. <https://doi.org/10.1357/002224003322005087>
- Umlauf, L., & Burchard, H. (2005). Second-order turbulence closure models for geophysical boundary layers. A review of recent work. *Continental Shelf Research*, 25(7–8), 795–827. <https://doi.org/10.1016/j.csr.2004.08.004>
- Vallis, G. K. (2017). *Atmospheric and oceanic fluid dynamics: Fundamentals and large-scale circulation* (2nd ed.). Cambridge University Press. <https://doi.org/10.1017/9781107588417>
- Van Roekel, L., Adcroft, A. J., Danabasoglu, G., Griffies, S. M., Kauffman, B., Large, W., et al. (2018). The KPP boundary layer scheme for the ocean: Revisiting its formulation and benchmarking one-dimensional simulations relative to LES. *Journal of Advances in Modeling Earth Systems*, 10(11), 2647–2685. <https://doi.org/10.1029/2018MS001336>
- van Zanten, M. C., Duynkerke, P. G., & Cuijpers, J. W. M. (1999). Entrainment parameterization in convective boundary layers. *Journal of the Atmospheric Sciences*, 56(6), 813–828. [https://doi.org/10.1175/1520-0469\(1999\)056<0813:EPICBL>2.0.CO;2](https://doi.org/10.1175/1520-0469(1999)056<0813:EPICBL>2.0.CO;2)
- Vreugdenhil, C. A., & Gayen, B. (2021). Ocean convection. *Fluids*, 6(10), 360. <https://doi.org/10.3390/fluids6100360>
- Wagner, G. L., Hillier, A., Constantinou, N. C., Silvestri, S., Souza, A., Burns, K. J., et al. (2025). Formulation and calibration of CATKE, a one-equation parameterization for microscale ocean mixing. *Journal of Advances in Modeling Earth Systems*, 17(4), e2024MS004522. <https://doi.org/10.1029/2024MS004522>
- Wagner, G. L., Silvestri, S., Constantinou, N. C., Ramadhan, A., Campin, J.-M., Hill, C., et al. (2025). High-level, high-resolution ocean modeling at all scales with Oceananigans. *arXiv*. <https://doi.org/10.48550/ARXIV.2502.14148>
- Wei, Z., Li, Q., & Chen, B. (2026). A direct assessment of Langmuir turbulence parameterizations in idealized coastal merging boundary layers. *Journal of Advances in Modeling Earth Systems*, 18(1), e2025MS004993. <https://doi.org/10.1029/2025MS004993>
- Young, W. R. (1994). The subinertial mixed layer approximation. *Journal of Physical Oceanography*, 24(8), 1812–1826. [https://doi.org/10.1175/1520-0485\(1994\)024<1812:tsmla>2.0.co;2](https://doi.org/10.1175/1520-0485(1994)024<1812:tsmla>2.0.co;2)
- Yu, L. (2007). Global variations in oceanic evaporation (1958–2005): The role of the changing wind speed. *Journal of Climate*, 20(21), 5376–5390. <https://doi.org/10.1175/2007JCLI1714.1>
- Yuan, J., Liang, J., Chassignet, E. P., Zavala-Romero, O., Wan, X., & Cronin, M. F. (2024). The K-profile parameterization augmented by deep neural networks (KPP_dnn) in the General Ocean Turbulence Model (GOTM). *Journal of Advances in Modeling Earth Systems*, 16(9), e2024MS004405. <https://doi.org/10.1029/2024MS004405>
- Zanna, L., Gregory, W., Perezogin, P., Sane, A., Zhang, C., Adcroft, A., et al. (2025). A framework for hybrid physics-AI coupled ocean models. *arXiv*. <https://doi.org/10.48550/ARXIV.2510.22676>
- Zaron, E. D., & Moum, J. N. (2009). A new look at Richardson number mixing schemes for equatorial ocean modeling. *Journal of Physical Oceanography*, 39(10), 2652–2664. <https://doi.org/10.1175/2009JPO4133.1>