Low-dimensional representation of intermittent geophysical turbulence with High-Order Statistics-informed Neural Networks (H-SiNN)

R. Foldes,[1, 2] E. Camporeale,[3, 4, 5] and R. Marino[1]

[1)]*Université de Lyon, CNRS, Ecole Centrale de Lyon, INSA de Lyon, Université Claude Bernard Lyon 1,*
Laboratoire de Mécanique des Fluides et d'Acoustique - UMR 5509, F-69134 Ecully,
France

[2)]*Dipartimento di Scienze Fisiche e Chimiche, Università dell'Aquila, 67100 Coppito (AQ),*
Italy

[3)]*Department of Physics and Astronomy, Queen Mary University of London, London E1 4NS,*
United Kingdom

[4)]*CIRES, University of Colorado, Boulder, CO, USA*

[5)]*NOAA Space Weather Prediction Center, Boulder, CO, USA*

(*Electronic mail: raffaello.foldes@ec-lyon.fr)

(Dated: January 17, 2024)

We present a novel machine learning approach to reduce the dimensionality of state variables in stratified turbulent flows governed by the Navier-Stokes equations in the Boussinesq approximation. The aim of the new method is to perform an accurate reconstruction of the temperature and the three-dimensional velocity of geophysical turbulent flows developing non-homogeneities, starting from a low-dimensional representation in latent space, yet conserving important information about non-Gaussian structures captured by high-order moments of distributions. To achieve this goal we modify the standard Convolutional Autoencoder (CAE) by implementing a customized loss function that enforces the accuracy of the reconstructed high-order statistical moments. We present results for compression coefficients up to 16 demonstrating how the proposed method is more efficient than a standard CAE in performing dimensionality reduction of simulations of stratified geophysical flows characterized by intermittent phenomena, as observed in the atmosphere and the oceans.

I. INTRODUCTION

Geophysical fluids are characterized by the interplay of non-linear vortices and waves, developing very complex turbulent dynamics due to the density stratification and the Earth's solid body rotation. Large-scale motions at planetary scales (of the order of 1000 km) emerge from the so-called quasi-geostrophic balance[1]. At mesoscale $O(100\,\text{km})$ and even more at the sub-mesoscale $O(10\,\text{km})$, when the turbulent eddy turnover time becomes comparable to the characteristic time scales associated to internal waves, motions feel less the constrain of the force balance; a regime generically identified as *stratified turbulence*[2]. This is characterized by the presence of shear layers leading to instabilities, the vertical extension of strata being controlled by the Brunt-Väisälä frequency N. In stratified geophysical flows, the probability density function (PDF) of the fields at the scales comparable to that of the mean flow can be characterized by fat tails with a departure from Gaussianity; in literature we refer to this phenomenology as *large-scale intermittency*[3–5], which differ from the classical (internal) small-scale intermittency detected as a departure from Gaussianity of the statistics of the field gradients. Large-scale non-Gaussian distributions of the dynamical fields are observed also in other fluid frameworks in nature, such as in the solar wind, where local shears associated to large values of large-scale increments of velocity and density are thought to trigger magnetohydrodynamic turbulent cascades in space plasmas[6,7]. Such peculiar behavior of stratified turbulent flows has been observed in the vertical velocity and (potential) temperature in both atmosphere[8–11] and oceans[12,13], and was extensively characterized in a variety

of numerical investigations[3–5,14–16] through the forth-order moment of the vertical velocity (i.e., through its kurtosis). Marino *et al.*[5] have shown how the extreme drafts, responsible for the non-Gaussian behavior of the vertical component of the velocity, do generate local turbulence and enhance the internal small-scale intermittency[4]. They provided as well evidence that these structures are associated with patches of enhanced kinetic and potential energy dissipation, making stably stratified flows in a certain parameter space (analyzed in Feraco *et al.*[3]) more efficient at dissipating energy and less homogeneous due to the irregular occurrence (in space and time) of these large-scale intermittent drafts. The community resorted to a variety of different techniques to model the wide range of scales over which dynamics develop in geophysical flows, from the implementation of codes based on Reynolds averaged Navier-Stokes (RANS) equations and Large Eddy Simulations schemes (LES)[17–19], to the simulation of rotating flows using reduced models[20]. On the other hand, progress made in the field of High-Performance Computing has made it possible to perform direct numerical simulations (DNS) of stratified turbulent flows in a parameter space of geophysical interest[21–24]. Regardless of the physical model implemented, a common trait of high-resolution three-dimensional simulations of geophysical flows is the large amount of data produced. For example, a DNS run with a resolution of 512^3 grid points in single precision requires more than 0.5GB of storage space per field and per time-step. Retaining the simulation output with high temporal cadence is therefore in many cases not doable, which requires to find a trade off though investigating the evolution of the fields in space and time is often needed to assess the complex dynamics de-

veloping in geophysical flows. The implementation of low-dimensional representations of the physical fields appears on the one hand to be a viable option for the creation of database allowing for the post-processing of massive data, relieving from the need to perform run-time analysis, ashearsnd on the other is useful for the usage of machine learning techniques able to assimilate features of the dynamical systems directly from the latent space[25–27]. The rationale behind the dimensionality reduction of the output of turbulent flow simulations serves therefore multiple purposes, storing data for post-processing at higher temporal frequency, enabling the efficient training of data-mining and machine learning techniques; implementing forecasting tools able to operate using coarse-grained descriptions of the system[28,29]. The study of low-dimensional manifolds has a long history in the context of dynamical systems and turbulence [30,31], laying at the foundation of so-called reduced order methods[32]. The latter have also recently benefited from progresses made in machine learning based approaches[33,34]. Dimensionality reduction techniques such as proper orthogonal decomposition (POD)[35–38] and dynamic mode decomposition[39,40] have been extensively used to address issues concerning a variety of flows[41–45]. Other examples of model order reduction techniques applied to fluid frameworks are Galerkin-projection based nonlinear methods[46,47] and system identification based auto-regressive models[48,49]. Techniques based on machine learning principles that have been employed in modeling turbulent flows, in very different contexts, are Gaussian process regression[50,51], symbolic regression[52,53], field inversion[54,55], artificial neural networks (ANNs)[56] and many others. In data-driven methods, ANNs are the most powerful tools in terms of their ability to generalize and to capture highly nonlinear phenomena, typical of turbulent processes. Other developments have focused on improving the reliability and accuracy of low fidelity models (i.e., RANS and LES), by using data from high fidelity simulations, e.g. DNS, either to learn proper closures[57–59] or to generate high-resolution synthetic turbulent states starting from a coarser description of the flow[60]. Convolutional neural networks (CNNs) have been successfully applied to the identification of flow structures[61] and to perform nonlinear modal decomposition in turbulent flows[62,63]; more recently, CNN-based architectures have been extended to fully 3D DNS of turbulent channel flow[64] and numerical simulations of flows characterized by the emergence of extreme events, in terms of boundary coherent structures[65]. These implementations emphasized the advantages of using CNNs over traditional methods based on principal component analysis (PCA), the former being able to capture the intrinsic nonlinear dynamics of turbulent flows. More generally, deep learning proves to be a powerful tool for the analysis and generation of reduced-order models of turbulent systems. In terms of machine learning applications to stratified flows, CNN-based deep learning has recently been applied by Salehipour and Peltier [66] to obtain a parametrization of diapycnal mixing using data from DNS. Convolutional AutoEncoders (CAE) are a family of autoencoders that proved successful in extracting information from two- and three-dimensional data[67] and their use has recently been proposed in the context of fluid-dynamics[68–70]. How-

ever, most of these approaches present severe limitations if implemented without proper tuning, and their use is often constrained to idealized and/or simplified cases due to the major constraints that come from capturing transient behavior[71]. Conversely, the presence of large-scale transient phenomena developing in geophysical flows, such as hurricanes, tornadoes, oceanic fronts, etc., can be viewed as an obstacle to the creation of reduced-order manifolds sufficiently informative to reliably recover properties and dynamical features of geophysical fluid systems. Multi-scale models, such as global circulation models (GCM), are powerful tools for investigating the Earth's atmosphere and the oceans which would greatly benefit from the possibility of simulating complex fluid dynamics on a dimensional phase space lower than what can be achieved numerically. The efficiency of machine learning approaches is reduced when applied to systems with strong transients, such as the non-stationary states resulting from the presence of intermittent phenomena in turbulent flows. The purpose of the study presented here is precisely to develop a machine learning tool based on CAE able to produce reliable low-order representations of stratified geophysical turbulent flows in a range of parameters characterized by the development of intermittent (in space and time) vertical velocity drafts, leading to the departure of the vertical velocity statistical moments from the Gaussian reference. We show that standard CAE architectures are well suited to learn a latent space containing enough information to reconstruct the general features of the original stratified turbulent flow. However, strong vertical drafts developing in a certain regime of the governing parameter, and in general high-order statistics of velocity and temperature in the presence of large- and small-scale intermittency, will typically not be well recovered. To overcome this issue, we introduce here a novel implementation of CAE: statistics-informed convolutional autoencoder (or SiCAE). The idea is based on a more general approach, applicable to any neural network, which we refer to as high-order statistics-informed neural network (H-SiNN). The concept of statistics-informed neural network (SiNN) was first introduced by Zhu, Tang, and Kim [72], where it has been proposed the addition of two terms to the loss function whose effect is to constrain both the PDF and the auto-correlation function of the reconstructed fields. On the other hand, in our implementation, the loss function explicitly enforces the preservation of high-order moments and this is done in order to ensure consistency between the statistics of the original and the reconstructed flow fields. Indeed, it is worth recalling that statistical moments of the velocity distribution function play a crucial role in the characterization of turbulent frameworks, being directly related to invariants such as energy and enstrophy, and also to other fundamental quantities, such as the dissipation occurring at the small scales. We will demonstrate here the enhanced capability of the H-SiNN, with respect to a standard CAE implementation, in recovering finer statistical features as well as flow inhomogeneities and intermittent dynamics.

The manuscript is organized as follows: in Sec. II the numerical simulations performed and analyzed are briefly introduced along with a description of the equations and pa-

rameters governing the dynamics of the stratified flows under study; Sec. III explains the classical CAE architecture, and in Sec. IV our implementation of statistical-informed CAE is thoroughly described; finally, Sec. V shows in detail the outcome of the application of the model reduction tool we developed to DNS of stratified turbulent flows developing large-scale intermittent events in the vertical component of the velocity and in the temperature field[3].

II. DNS SOLVER FOR STRATIFIED TURBULENCE

The CAEs are trained using the output of a direct numerical simulation of a stably stratified turbulent flow obtained by integrating the Navier-Stokes equations in the Boussinesq approximation in which the velocity field $\mathbf{u} = (\mathbf{u}_\perp, w)$ remains incompressible, $\nabla \cdot \mathbf{u} = 0$, while small density variations are taken into account only in the buoyancy term. Such a model can be written as:

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p - N\theta \hat{\mathbf{z}} + \nu \nabla^2 \mathbf{u} + \mathbf{F} \quad (1)$$

$$\partial_t \theta + (\mathbf{u} \cdot \nabla)\theta = Nw + \kappa \nabla^2 \theta . \quad (2)$$

with θ being the temperature fluctuations evaluated relative to a mean temperature profile θ_0, $N = [-g\partial_z \bar{\theta}/\theta_0]^{1/2}$ is the Brunt-Väisälä frequency. Such a simulation has been initialized with zero temperature fluctuations and random velocity modes applied at large-scale, in a Fourier shell centered at $k_0 = [2,3]$; the size of the computational box is associated with $k_{min} = 1$, with $k_{min} = 2\pi/L_0$ and the resolution is 512^3 grid-points. A random forcing F is imposed on the momentum equation at $k_F = 2\pi/L_f \in [2,3]$, continuously injecting kinetic energy into the system and allowing it to reach a turbulent stationary state. In the above equations, ν and κ are the kinematic viscosity and the thermal diffusivity, respectively and we take the Prandtl number $\nu/\kappa = 1$, with $\nu = 10^{-3}$; finally, p is the pressure. We adopt the following definitions for the dimensionless Reynolds and Froude numbers,

$$\mathrm{Re} = U_{rms}L_{int}/\nu, \ \mathrm{Fr} = U_{rms}/[L_{int}N] , \quad (3)$$

with U_{rms} and L_{int} the characteristic (root mean squared) velocity and the integral scale of the fluid, respectively. These parameters, and in particular the buoyancy Reynolds $R_B = \mathrm{ReFr}^2$, measure the relative strength of buoyancy to dissipation and are commonly used to distinguish between wave-dominated and turbulence-dominated regimes[73]. The equations (1–2) are integrated numerically using the Geophysical High-Order Suite for Turbulence (GHOST), a pseudo-spectral code that employs a hybrid parallelization combining MPI, OPENMP, and CUDA[74,75]. It allows for a variety of physical solvers and supports non-cubic geometry[76], and non-periodic boundary conditions[77]. In the following, we will make use of the statistical moments of a distribution function up to the fourth-order. The definition of the third- and forth-order moments, namely the skewness Sk_w and kurtosis K_w respectively, of the distribution of vertical velocity w is:

$$Sk_w = \frac{\langle (w - \bar{w})^3 \rangle}{\langle (w - \bar{w})^2 \rangle^{3/2}}, \quad (4)$$

Id	Compression factor	No. Layers	No. params.	Latent space
CF2	2	7	321,156	$64 \times 64 \times 128$
CF4	4	8	339,892	$32 \times 32 \times 256$
CF8	8	9	324,772	$32 \times 32 \times 128$
CF16	16	11	349,804	$16 \times 16 \times 256$

Table I. Description of the four CAEs architecture: the compression factor (CF), the number of layers referring to both the encoding and decoding part of the network. The number of parameters accounts for all the weights and biases that have to be optimized during the training phase, and the latent space is the reduced space after the encoding network is applied.

$$K_w = \frac{\langle (w - \bar{w})^4 \rangle}{\langle (w - \bar{w})^2 \rangle^2}, \quad (5)$$

where averages can be either computed over the entire simulation volume or on sub-domains, such as horizontal planes. It is worth recalling that $Sk = 0$ and $K = 3$ are the reference values for a Gaussian distribution, whereas values of K_w larger than 3 are indicative of PDFs with fat (non-Gaussian) tails[3]. As observed by Feraco et al.[3], the volume kurtosis K_w shows large non-Gaussian values in a narrow range of the Froude number, with a peak at $Fr \approx 0.076$, compatible with actual geophysical flows. The study presented here focuses on a run characterized by this value of the Froude number, in[3–5,78], exhibiting indeed high levels of large-scale intermittency (see Fig.2 of Feraco et al.[3]). This simulation is well resolved, meaning the ratio between the largest wave number $k_{max} \approx 512/3$ and the Kolmogorov scale $k_\eta = (\nu^3/\varepsilon_v)^{-1/4}$, ε_v being the kinetic dissipation rate, is $k_{max}/k_\eta \sim 1.8$.

III. CONVOLUTIONAL AUTOENCODER THEORY

An autoencoder is an unsupervised feed-forward artificial neural network (ANN) that aims to reconstruct a given data set through a process involving data compression and recovery[79]. Indeed, its main objective is to create a reliable and reduced representation of the input data set, which is particularly helpful for either creating reduced models or simpler but still informative representations of high-dimensional data. The network consists of an encoder part $P : \mathbb{R}^D \to \mathbb{R}^d$ (with $d < D$), which compresses the input data into a smaller space representation (or latent space), and a decoder part $Q : \mathbb{R}^d \to \mathbb{R}^D$, which reconstructs the encoded data back to the original input dimension. When the mapping kernel is linear, the autoencoder can be considered as a singular value decomposition analysis. An autoencoder that employs convolutional layers to perform the encoding and decoding operations on images, or more in general on 2D (or higher) data, is generally termed Convolutional AutoEncoder (CAE). During the convolution operation, the kernel swipes the data domain to extract features and learn spatial and/or temporal dependencies. The process is then carried out across multiple layers, obtaining representative features in a hierarchical sense[80]; CAEs with many hidden layers can be considered as deep neural networks (DNN). The initial layers learn a low-level representation or particular

details about the data, while subsequent layers focus on larger and larger-scale information. In our work we developed four CAEs with different compression factors (CFs), defined as the ratio between the input and the latent dimensions, from 2 to 16; this can be also seen as the ratio between the degrees of freedom of the initial system with respect to the reduced one, being generally proportional in numerical simulations to the number of grid points, therefore to the Reynolds number in the case of fluids. An example of a general CAE architecture used in our implementations is shown in Fig. 1. In Tab. I we report the main characteristics of the four CAEs implemented here; starting from CF2, we obtained higher CFs by adding more hidden layers to the initial architecture but trying to keep the number of parameters fixed (as possible) in order to make a more fair comparison of the four networks performance. Indeed, the number of weights and biases, as highlighted by the values in Tab. I, varies only by $\approx 10\%$. On the other hand, more hidden layers allow networks with higher CF to learn more features of the original fields during the compression step. The number of hidden layers is intended for the entire CAE architecture, comprising both the layers of the encoding n_e and decoding n_d part; they usually have a number of layers which follow the relation $n_d = n_e + 1$, while the CAE is symmetric if the total number of layers is even.

IV. HIGH-ORDER STATISTICS INFORMED NEURAL NETWORK: H-SINN

The aim of this work is to devise a method that attains a compression of the data volume of DNS for stratified turbulence flows, while preserving their statistics (up to the fourth-order moment, kurtosis). Mathematically, this goal can be cast as a multi-objective optimization (MOO) problem and one of the greatest advantage of using a neural network to solve MOO problems is that, in principle, it is sufficient to include any term that one wishes to minimize in the cost function and, as long as such terms are differentiable, the neural network will find a solution that approximately minimizes the objective (possibly a local minimum) through the back-propagation procedure. This is known as *scalarization method*, where the different terms are simply added together, each one weighted by its own scalar factor[81]. Determining those weights is crucial, especially when employing ANNs, to make sure that the cost function is not dominated by one or a few terms only and that the solution actually minimizes each term. Intuitively one can think of the weights as normalization factors when different terms in the cost function have different orders of magnitude. Our implementation of H-SiNN involves the use of a statistics-aware loss function in a CAE, thus creating what we call statistics-informed convolutional autoencoder (SiCAE). As mentioned above, we focus on up-to-fourth order moments of the vertical velocity distribution, since we already know that it presents the highest variability in our simulations. In particular, our aim is to preserve standard deviation, skewness, and kurtosis of the vertical component of the velocity field w, which we indicate with the symbols σ_w, Sk_w, and K_w, respectively. As a result, the cost function of the CAE com-

puted between the original data set y and its reconstruction $\tilde{y}$ is defined as:

$$\mathcal{L}(y,\tilde{y}) = \frac{1}{D}\left(W_1 \sum_{y}^{[u,v,w,\theta]} \sum_{i=1}^{M} \frac{(y_i - \tilde{y}_i)^2}{M} + W_2 \sum_{i=1}^{M} \frac{(\sigma_{w_i} - \sigma_{\tilde{w}_i})^2}{M}\right.$$
$$\left. + W_3 \sum_{i=1}^{M} \frac{(Sk_{w_i} - Sk_{\tilde{w}_i})^2}{M} + W_4 \sum_{i=1}^{M} \frac{(K_{w_i} - K_{\tilde{w}_i})^2}{M}\right)$$
$$= \frac{1}{D}\sum_{j=1}^{4} W_j R_j$$

(6)

with $D = 3(W_1 + W_2 + W_3 + W_4)$, being the total sum of the weights. The second, third, and fourth terms in (6) are the mean square errors (MSEs) between the moments of the original and the reconstructed vertical velocity W computed over horizontal planes (x,y), namely the standard deviation σ_w, the skewness Sk_w and kurtosis K_w respectively. We recall that for a normal distribution, the reference values are: $\sigma = 1$, $Sk = 0$ and $K = 3$. In addition, here, we propose a method for dynamically weighing the four terms in the cost function; each term is properly weighted with values W_i varying at each epoch during the training phase. In order to obtain a loss function with terms having the same order of magnitude during the whole training procedure at a given epoch m, the weight of the ith term W_i is proportional to the sum of the other three terms at the previous epoch $m-1$, e.g. $W_i^m = \sum_{j\neq i} W_j^{(m-1)} R_j^{(m-1)}$ (and normalized such that their sum is equal to 1). Even though the various terms of the statistics-informed loss function are well balanced, the training procedure with additional constraints either coming from the knowledge of the physical system, as it happens for physics-informed neural networks (PINNs), or from statistical information can be difficult, slower, and sometimes very noisy. In order to mitigate gradient's pathologies when dealing with complex loss functions having several terms, a different approach consists in defining W_i as proportional to the gradients of the i-th term with respect to the network parameters[82]. This technique was shown to be particularly efficient in balancing the loss function, especially in PINNs[82,83]. A claim was made in the referenced studies that using weights proportional to the loss itself leads to convergence issues, though it does not seem to be the case in our application, where the various loss terms are well balanced and the global loss trend decays properly (see Fig. 2). That being said, a comparison between the two approaches would be interesting as a future study as it may lead to further improve the novel CAE we propose here.

Finally, we have heuristically verified that the strategy of training an H-SiNN in two stages is computationally faster and more robust. In the first stage we treat the autoencoder as a classical CAE, with only the first term in Eq. 6. After we train for 50 epochs, we add the remaining terms in the cost function, i.e. starting from a network that achieves a low reconstruction error, in terms of mean error. During the second stage, we train the CAEs for 100 more epochs with the addition of statistical constraints on the first four moments of the vertical velocity PDF; this is shown in panels (b)-(f) of Fig. 2. A similar strategy has been successfully implemented for a va-

Input
512 x 512 x (4 variables)
(u, v, w, θ)

Output
512 x 512 x (4 variables)
(u, v, w, θ)

Latent Space

Conv2D
+
MaxPool2D

Conv2D
+
MaxPool2D

Conv2D
+
UpSampling2D

Conv2D
+
UpSampling2D

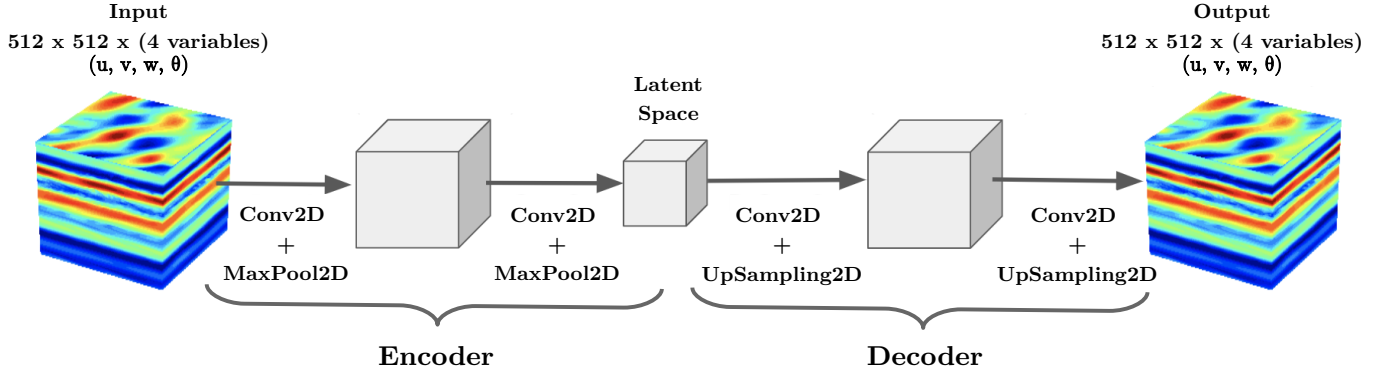**Encoder**   **Decoder**

Figure 1. Schematic representation of a general Convolutional Autoencoder (CAE). The dimension of the *latent space* and the number of encoding/decoding layers depend on the compression factor (CF). For each time step, the input is represented by horizontal slices (512^2) of the four physical variables produced by the simulations (u, v, w and θ) stacked together forming a volume of $512^2 \times 4$. The number of filters and layers composing the encoder (and decoder) part also depends on the CF, all the CAEs implemented in this work are symmetric.

riety applications, most of them in the context of PINNs[83,84]. The behavior of the different terms of the loss function during one hundred epochs of training is shown in Fig. 3. We can appreciate from the four panels how there are essentially three weights (W_{MSE}, W_{σ_w} and W_{Sk_w}) which are nearly equal for the entire training phase, whereas the coefficient weighting the kurtosis term is significantly smaller indicating a larger error on the forth-order moment, as expected.

A. Plane-by-plane approach

We mentioned that the main objective is to obtain a tool that is able to reproduce most of the features of the velocity **u** and temperature θ fields of a fluid presenting high anisotropy (stratification) and non-homogeneity, due to the presence of large-scale intermittent structures. For these last reasons, we further implemented the standard CAE algorithm to make it more suitable and better performing in our particular application. A key ingredient of stratified flows is anisotropy, introduced by gravity, which tends to suppress vertical motions. Therefore, even though we deal with fully three-dimensional simulations the original cubic volume is split in horizontal planes along the vertical direction (z) before being used as input for the CAEs. This means that, for each time and height value z, planes for every velocity component (u, v and w) and for θ are stacked together creating three-dimensional input data with dimensions $512 \times 512 \times 4$, as indicated in Fig. 1. Before being divided, the data are normalized $\hat{x} = (x - \mu)/\sigma$ using the average μ and standard deviation σ computed on each simulation cube. In this approach, we believe that passing the information from the three components of the velocity, together with the temperature, is essential for the CAE to better reconstruct the flow dynamics at a given time t and altitude z since their variation is strictly correlated within the primitive equations. This approach is limited in the possibility to add important information about the velocity field, like the compressibility condition $\nabla \cdot \mathbf{u} = 0$, since operating by-plane poses constraints on the vertical derivative of the veloc-
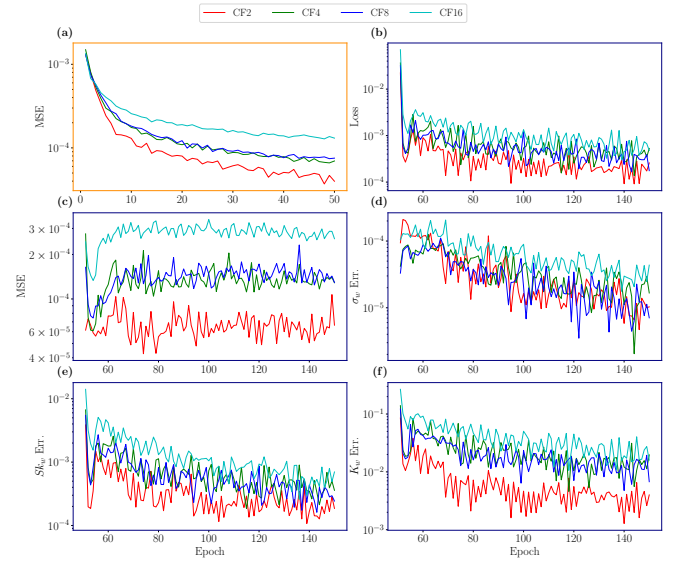

Figure 2. Panel (a) shows the mean squared error (MSE) during the first 50 epochs of the training stage where no other terms are added to the loss function. In panel (b) the total loss during 100 epochs of the second training stage, in which the loss function is composed by four terms which are represented in panels (c)–(f) for the various CAE with different compression factor (CF).

ity $\partial_z w$. This leads to values for the maximum and average divergence of **u** reported in Tab. II, where we can see that even if on average the condition is well satisfied in the reconstructed fields, locally this is not true while the condition is satisfied also point-wise for the original field.

V. RESULTS

We analyze in detail the output of a single simulation run for more than $45\tau_{NL}$ corresponding to 100 time steps. Since in our approach we divide each simulation cube in planes along
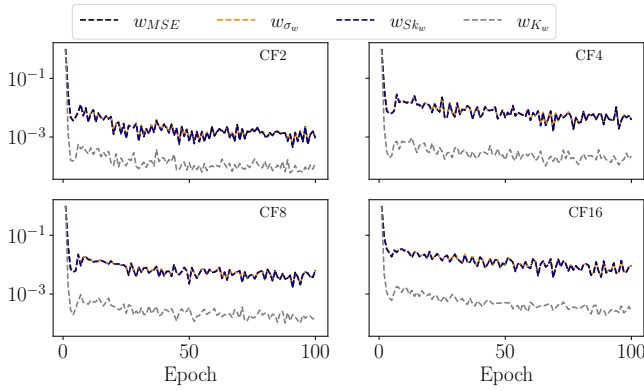
Figure 3. History of the weights multiplying the different terms of the loss functions: the mean squared error (MSE, black), the standard deviation (σ_w, orange), the skewness (Sk_w, blue) and the kurtosis (K_w, gray).

CAE	MSE $[\times 10^{-4}]$	σ_w err. $[\times 10^{-5}]$	Sk_w err. $[\times 10^{-4}]$	K_w err. $[\times 10^{-2}]$	Loss $[\times 10^{-4}]$
1st stage "standard loss"					
CF2	*0.39*	*0.13*	*10.9*	*18.4*	*0.39*
CF4	*0.55*	*0.45*	*13.2*	*22.4*	*0.55*
CF8	*0.54*	*0.82*	*19.1*	*19.0*	*0.54*
CF16	*1.1*	*5.3*	*44.9*	*41.7*	*1.1*
2nd stage "statistical-informed loss"					
CF2	0.67	1.03	1.9	0.04	0.44
CF4	1.0	1.0	3.2	1.4	4.0
CF8	1.3	0.074	3.6	1.7	2.3
CF16	2.3	0.32	12.0	5.9	10.0

Table III. Results of the four CAE on the test set for both the first (top) and second (bottom) training phase. Values in italic represent only check quantities monitored during the first training phase, though they are not included in the loss function at this stage.

the vertical direction which are then treated as independent during training, the data set we use for training and testing our CAEs comprises $N_s = 51,200$ samples, each with dimension $512^2 \times 4$ (512^2 grid-points for four variables u, v, w and θ). We divide the data set in training and test set on a temporal basis, taking the first 70% snapshots for the training phase and the remaining 30% for testing the performance.

The above strategy has been used to train four different statistics-preserving CAE, with compression factors (CF) of 2, 4, 8, and 16. Figure 2 shows the history of the different terms in the loss function, for an increasing number of epochs. Red, green, blue, and cyan lines are respectively for CF 2, 4, 8, and 16. The first stage is made of 50 epochs, and in the second stage, the networks are trained for an additional 100 epochs. The interesting feature, valid for all CFs, is that after the first stage (panel a) the MSE does not further decrease (panel c), while all other moments decrease by one order of magnitude or so (panels d-f). This shows that the second stage of training achieves the goal of minimizing the error of standard deviation, skewness, and kurtosis, without significantly degrading the point-wise accuracy of the reconstruction.

A. Global statistical properties of reconstructed fields

The trend of the statistical moments error appears to be inversely proportional to the moment order, meaning that the improvement on the reconstructed standard deviation (2nd-order moment) is more significant than what is achieved for the kurtosis (4th-order moment) within the same amount of

	Original	CF2	CF4	CF8	CF16
$|\langle \boldsymbol{\nabla} \cdot \mathbf{u} \rangle|$ $[\times 10^{-11}]$	0.42	1.83	3.26	2.37	2.06
$max\{|\boldsymbol{\nabla} \cdot \mathbf{u}|\}$	$5.9 \cdot 10^{-4}$	538	410	573	530

Table II. Average and maximum value of the divergence of the velocity field after the reconstruction with the different CAE.

epochs. This is also pointed out in Fig. 3 where, for the four different SiCAE, the trend of the various weights balancing the loss function terms is reported as a function of the epoch; let us recall that each weight W_i is inversely proportional to the sum of the three contributions given from the other moments, taken at the previous epoch $1/\sum_{(i \neq j)} W_j$. In addition, by looking at the behavior of the loss weights (Fig. 3), we observe that the error on the kurtosis is approximately between one and two orders of magnitude larger than the others. This significant difference may be due to the high variability of K_w in this simulation[5]. Instead, standard deviation and skewness are expected to vary less and to be closer to reference values of a Gaussian distribution. As it is reasonable, a trend with the compression factor is observed, and as expected the higher CF the larger the reconstruction error both on the mean field and the statistical moments; this is not related to the newly introduced loss function since already during the first training stage (panel (a)) this trend can be clearly observed. The details of the results obtained on the test set for the various terms involved in the loss function during the two training stages are summarized in Tab. III. We can see that including other terms into the loss function results in an average reconstruction error 2-3 times greater than what is obtained after the first training stage, even though it remains on the order of MSE$\sim 10^{-4}$, consistent with other results found in literature[85,86]. In addition, when constraints on the statistical moments are added to the model, we observe that the reconstruction of high-order moments improves up to 10 times for any compression factor.

We selected an interval of the numerical simulation where isolated extreme bursts occur, producing evident peaks of the kurtosis K_w (see Fig. 4, panel (a)). In Fig. 4 the vertical velocity kurtosis computed on the original data (black dashed curve) is shown as a function of the turnover time for the entire test set, and compared with the kurtosis obtained from the CAEs reconstructions (different colors for different CF values). Panels (a) and (b) show a comparison obtained respectively after the first and second training phase; the relative error between the original K_w^a and recovered K_w^r kurtosis is shown in panels (c, first phase) and (d, second phase). This figure clearly highlights how the second stage of the training (panels (b) and (d)) significantly improves the reconstruction
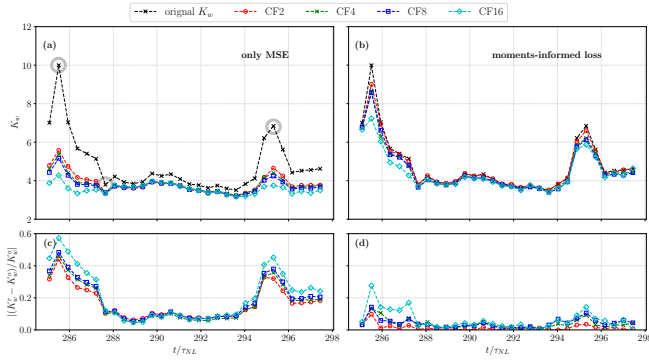
Figure 4. Behavior of the kurtosis of the vertical velocity K_w over time for the test set. Panels (a) and (c) show K_w obtained for the different compression factors (CFs) in different colors (top) and the mean absolute relative error on the reconstructed kurtosis (bottom) during the first training stage when the loss function is composed by the MSE only. The same results are shown in panels (b) and (d) for the second training stage where the modified loss is adopted. Gray circles in panel (a) represent three times analyzed in the following figures with very different values of kurtosis K_w of the vertical velocity.
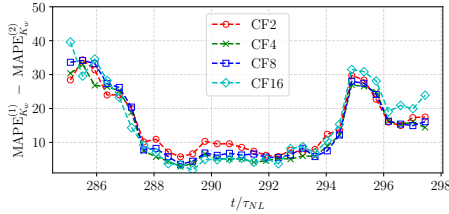
Figure 5. For each CAE, the behavior over time of the difference between the mean absolute percentage errors (MAPE) on the kurtosis K_w of the vertical velocity between the first $\mathrm{MAPE}_{K_w}^{(1)}$ and second training stage $\mathrm{MAPE}_{K_w}^{(2)}$. The MAPE is defined as $\mathrm{MAPE}_{K_w}^{(i)} = |(K_w^{r,(i)} - K_w^o)/K_w^o| \times 100$ with K_w^o and $K_w^{r,(i)}$ being the kurtosis of the original and reconstructed fields, respectively.
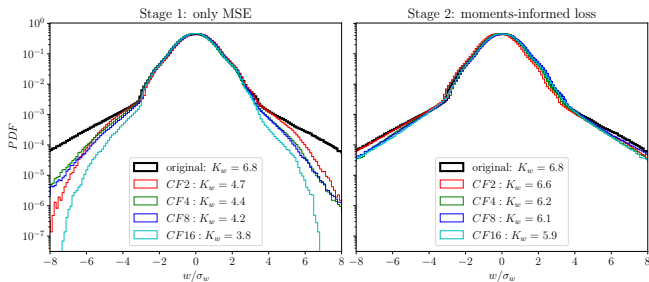

Figure 6. PDF of the vertical velocity $P(w)$ computed after the first training stage (left panel) and at the end of the entire training stage (right panel), taken at $t/\tau_{NL} \approx 295$. One can appreciate the PDF core being in excellent agreement during stage one, while the fat tails of the distribution are only recovered after introducing the custom terms to the loss function in the second training stage (right panel).

of the fields up to the fourth-order moment of the vertical velocity. All the CAEs show a significant reduction of the relative error, with marked improvements at the times at which the volume K_w exhibits large non-Gaussian values ($t \approx 285\tau_{NL}$ and $t \approx 295\tau_{NL}$). In addition, with the application of the modified loss we are able to obtain percentage errors on the kurtosis smaller than 10% on average. However, the improvement in the reconstruction of the fourth-order moment is obtained also for values close to the Gaussian reference ($K_w \approx 3$). Indeed, it is evident that our approach can be successfully extended to other situations where the PDFs of the dynamical fields are far from Gaussian-like shape, as is the case for the concentration of a passive scalar[87] or for the velocity in the boundary layer[8,88]. The statistical moments up to the fourth-order embedded into the loss function guarantee an optimal reconstruction of the statistical properties of the physical fields without any prior knowledge about the shape or variability of their distribution function. The improvement we obtain with the custom loss is quantified in Fig. 5 where we show the difference between the mean absolute percentage error (MAPE) computed on K_w after the first and second training stage, as a function of time (once again, note that the training is performed by treating each plane and each time independently, i.e. no correlation along the z direction or in time is taken into account). The improvement reaches up to $\approx 40\%$ at times with high values of K_w but in general, an increase of nearly 10% is observed at all times. In Fig. 6 we show the PDFs computed on the entire volume at time $t \approx 295\tau_{NL}$ (third gray circle in Fig. 4), when $K_w \simeq 6.8$, using both the original data (black) and data obtained from the four CAEs (colored curves). We notice how after the first training stage (left panel) the PDF core is already reliably recovered by the decoder, even though the tails significantly differ. In particular, the difference between original and reconstructed statistics seems to increase for higher $|w|$. The PDF in the right panel of Fig. 6, obtained after the second training stage, completely resembles the one computed from the original data, confirming that enforcing the statistical moments up to the forth-order is sufficient to enforce compatible PDFs. In Fig. 7 we show an example of large-scale intermittent structures observed at three different times, indicated by the gray circles in Fig. 4. The three-dimensional renderings represent the vertical velocity field w for low (top, $t \simeq 288\tau_{NL}$), medium (center, $t \simeq 295\tau_{NL}$) and high (bottom, $t \simeq 285\tau_{NL}$) values of the kurtosis K_w, where values $|w/\sigma_w| > 4$ are highlighted in solid color while smaller values are depicted as a transparent blue. Alongside each rendering the vertical profiles of the kurtosis K_w computed plane-by-plane on the original data (black), as well as on the reconstructed field (colored for the different CFs) are reported. Also for the top panel, when the kurtosis is small $K_w \simeq 3.8$ there are planes reaching values of $K_w \approx 6$, thus indicating a strong spatial variability even when the global kurtosis is close to the Gaussian reference value. These effects are enhanced when more vertical drafts develop within the flow (center and bottom) with kurtoses more than one order greater than the Gaussian reference values, as shown by the logarithmic scale in Fig. 7.
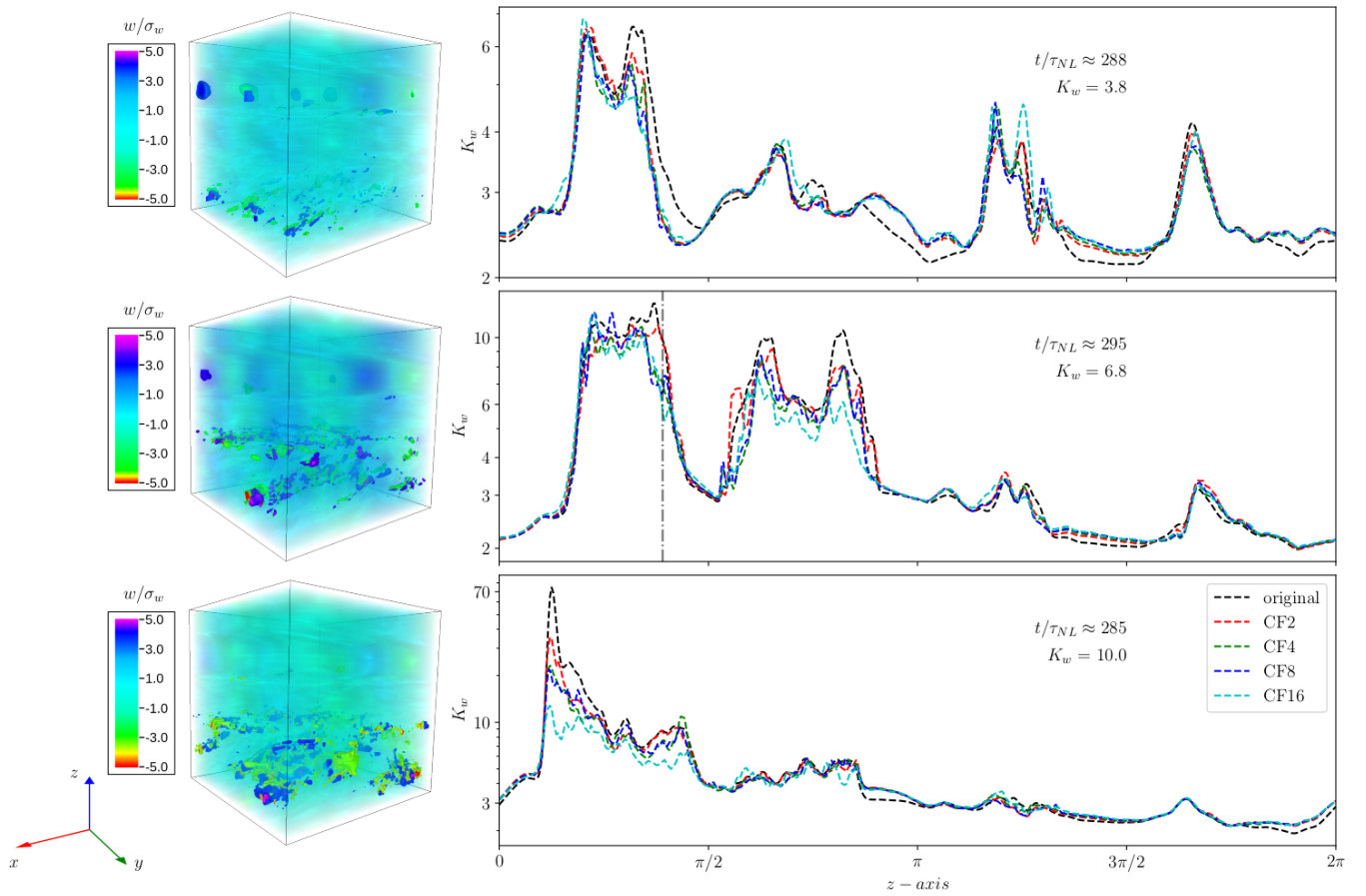
Figure 7. Left column: 3D renderings of the vertical velocity w/σ_w, where the most extreme structures have been highlighted in solid colors. Right column: Vertical profile of the vertical velocity kurtosis $K_w(z)$ of the original field (black dashed) and of the fields reconstructed with the SiCAE (colored, see legend). The vertical gray dash-dotted line (mid panel) represents the time and height taken for Fig. 9. Let us notice that the kurtosis is in logarithmic scale and the three panels show a different y-range for a clearer visualization.

B. CAE reconstruction of intermittent structures

We observed that the statistical properties and in particular the kurtosis of the vertical velocity of the flow under study are well recovered by the proposed implementation of statistics-informed CAEs. As already mentioned, the presence of large-scale extreme events produces intermittent patches of enhanced turbulence and regions populated by transient coherent structures, making the reconstruction of the fields and the associated non-Gaussian statistics challenging. In this section, we analyze how the introduction of global statistical terms to the loss function does improve the point-wise reconstruction of the full velocity ($\mathbf{u}$) and potential temperature (θ) field over the entire domain, also by means of visualizations. In Fig. 8 we report three-dimensional renderings of the vertical velocity w obtained with the four CAEs, as well as the original field for comparison. Data are taken at time $t \simeq 295\tau_{NL}$, characterized by relatively high kurtosis, $K_w \approx 6.8$. Extreme vertical drafts are localized in the regions of the flow characterized by very large values of the original normalized vertical velocity ($|w/\sigma_w|$), in Fig. 8 these correspond to the domain points with $|w/\sigma_w| > 4$. The same kind of structures emerge in the

renderings of the reconstructed field, up to the highest compression level (i.e., CF16); though for $CF = 16$ the shape and location of the regions characterized by vertical drafts appears blurred in the visualization compare to the original field, good spatial correlation can still be appreciated. The gray transparent shading represents velocity values below the threshold of $4\sigma_w$, and therefore the majority of the volume. The reconstruction made by the implemented CAEs involves also the other components of the velocity field (u and v), as well as the (potential) temperature fluctuations θ. In order to have a general overview of how the neural networks recover all the physical quantities interested in the analyzed DNSs we represent them with several panels in Fig. 9. This figure shows a horizontal cut (x,y) of the simulation domain taken at the same time as the previous figure (Fig. 8 at the height z^* indicated with a dash-dotted line in Fig. 7 (middle panel). The columns of Fig. 9 refer to the three components of the velocity field $\mathbf{u} = (u, v, w)$ and the temperature θ. All the images are represented by the same color bar which is not shown since the main objective of this figure is the comparison between the first row, the original data, and the others, being the reconstructed physical fields for increasing values of the com-
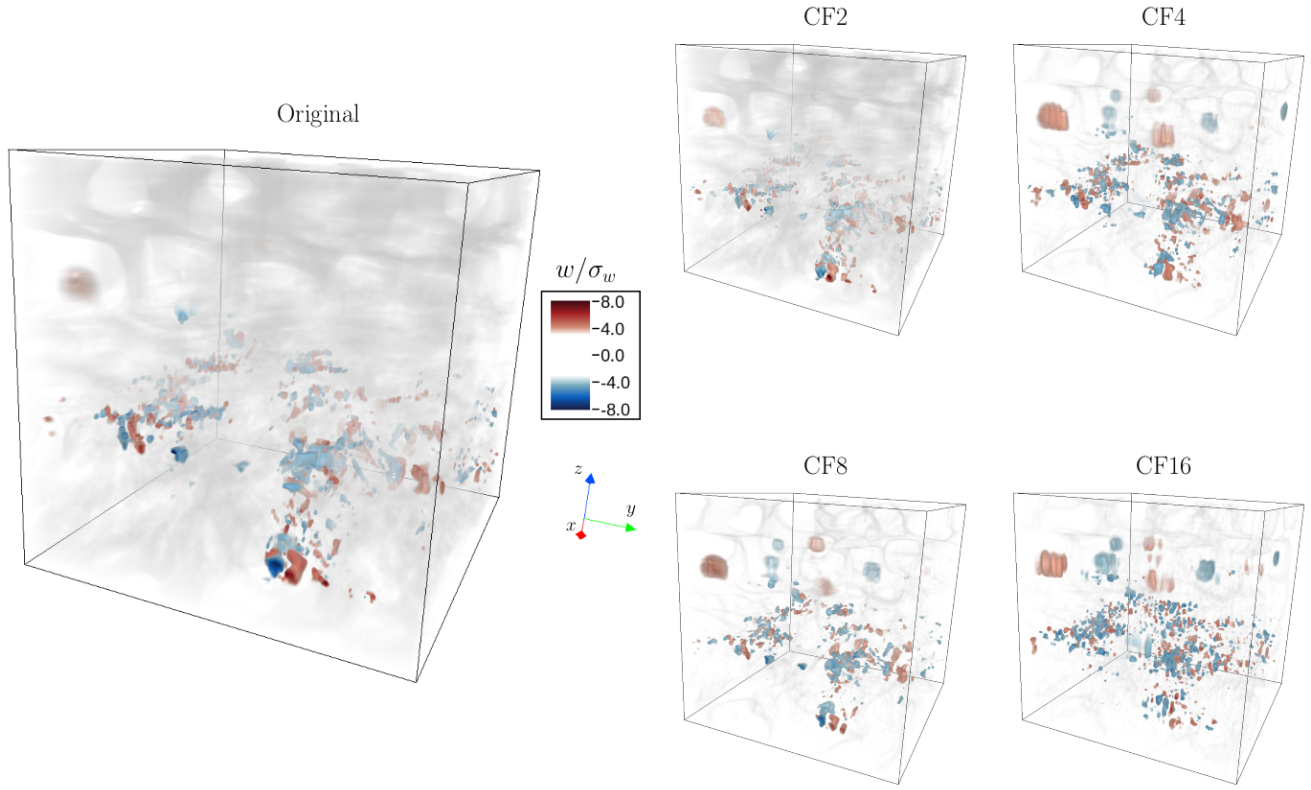
Figure 8. Three-dimensional visualizations, taken at $t/\tau_{NL} \approx 295$, of the extreme events in the vertical velocity w from the original data (big cube), alongside the field reconstructed by the four SiCAE (small cubes). The vertical drafts ($|w/\sigma_w| > 4$) are highlighted as solid colors (red/blue for positive/negative), whereas the other regions are represented as gray-shaded areas

pression factor. Since we are looking at the domain from the top (gravity is a vector entering the page), it is correct to have horizontal components strongly dominated by a nearly constant positive (red) for v and negative (blue) for u mean wind (see color bar); this is indeed the effect of Vertical Sheared Horizontal Winds (VSHWs) which are ubiquitous in stratified flows. Nevertheless, the horizontal components of the velocity show small-scale perturbations where extreme vertical drafts develop, as already seen throughout this manuscript. The extreme events developed in this snapshot are clearly visible in the vertical velocity w and partially from the temperature renderings (third and fourth columns in Fig. 9), and the same detail is captured also by the field reconstructions. Indeed, as already noticed for the three-dimensional visualization, the reconstruction is very reliable up to CF = 16 where a significant checkerboard effect starts developing everywhere in the domain. This is probably due to the combined effect of the high compression factor and of the statistical-informed loss function presenting large-scale statistical constraints. In fact, by looking at the reconstructed fields after the first stage of training (not shown here), one can observe the same artifact at high compression, though slightly reduced due to the absence of other terms in the loss function.

VI. CONCLUSION

The outcome of this work demonstrates how the capability of machine learning to reproduce dynamical fields in fluid mechanics can be enhanced by incorporating physical and/or statistical knowledge of the system under study. In particular, we proposed a novel implementation of convolutional autoencoder (CAE) that includes a modified loss function able to preserve the statistical moments of the vertical velocity field up to the fourth order. We focused on the vertical component of the velocity field produced by DNS of the Boussinesq equations with stable density stratification, since in a certain range of the parameters of geophysical interest - of the Froude number in particular - these stratified flows develop strong velocity drafts along the z direction, as observed in the atmosphere and oceans. The existence of this phenomenology resulting from the interplay of turbulent motions and internal gravity waves, represents a challenge when it comes to find an informative low-order representation of the physical fields. The comparison between a standard implementation of CAE and the SiCAE shows how the introduction of additional terms in the cost function to enforce high-order moments of the PDF, significantly improves the reconstruction of peculiar features of stratified geophysical flows, while maintaining an acceptable level of the error in the reconstruction of the mean fields. The diagnostics implemented here confirm that the novel al-
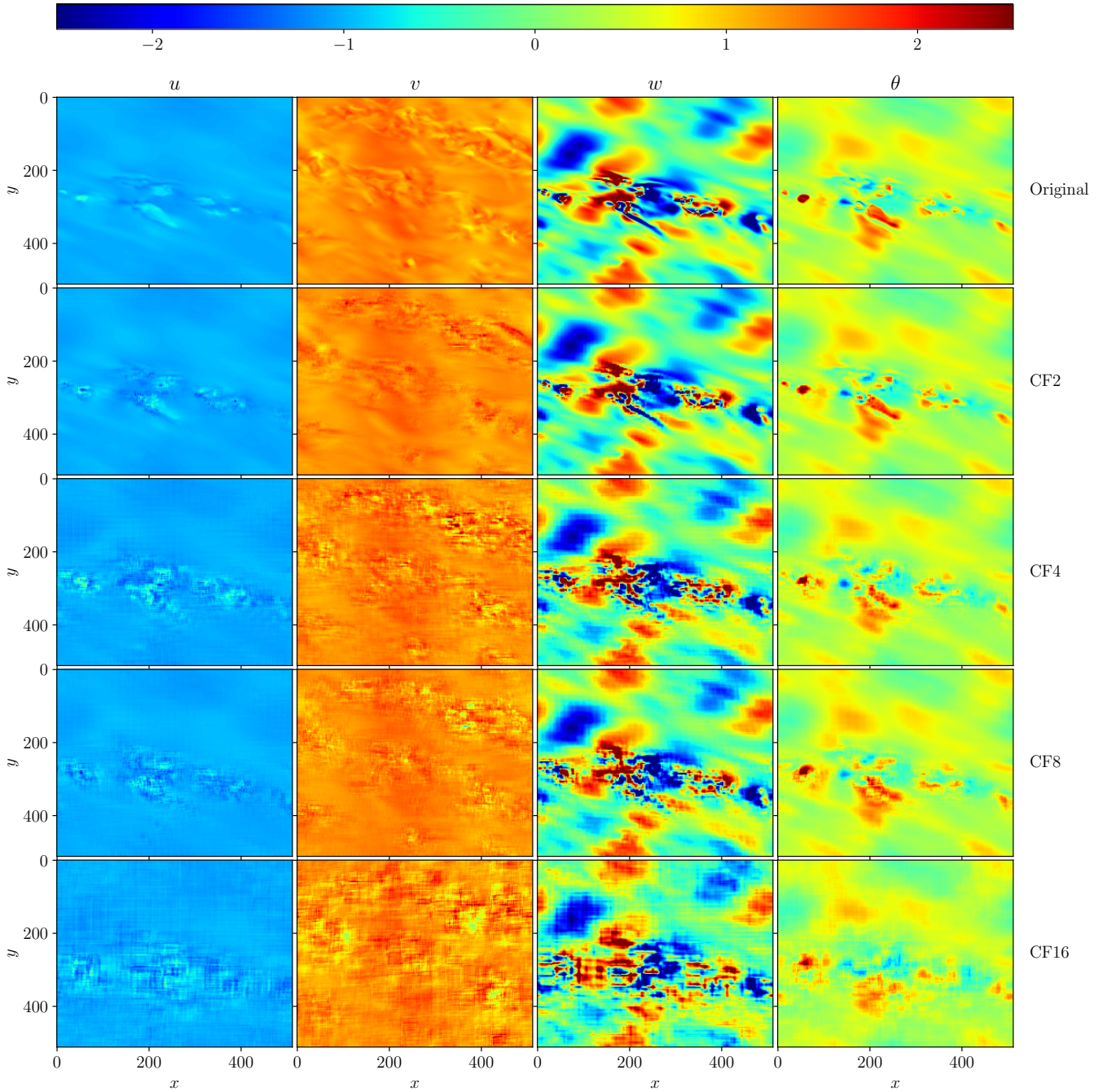
Figure 9. Horizontal slices of the four physical fields, from left to right: horizontal component of the velocity u and v (left and mid-left), the vertical component w (mid-right) and the potential temperature θ (right) taken at $t/\tau \approx 295$ with a kurtosis $K_w(z^*) = 10.0$. From the second line to the bottom the reconstructed field obtained with different CAE having increasing compression factor (CF).

gorithm we proposed is able to reconstruct with good accuracy physical fields characterized by highly variable PDFs. In the framework considered, the emergence of vertical drafts affects the local dynamics of stratified turbulent fluids, which end up being both non-stationary and non-homogeneous, making it more difficult to model their statistical description. The results we obtained can be potentially generalized to other physical fields and systems, and used to address for instance the re-

construction of intermittent passive scalar fields[89], in order to investigate their diffusion properties, or the velocity field obtained from observations of the planetary boundary layer[8,88] and the upper atmosphere[11,90]. Highly variable PDFs have also been obtainted from the analysis of dynamical fields measured in the oceans by means of GOMs[91], and using data from observations[12], further widening the application of the analysis proposed here, which can be very helpful in managing

large volumes of geophysical data from high-resolution numerical simulations and state-of-art observational campaigns. The addition of other terms to the loss function requires to have longer training phases and larger dataset, in order to converge to a well-trained model. Indeed, very often a compromise is made between having detailed reconstructions of the mean field dynamics and reproducing peculiar structures of the turbulent flows, which is precisely where the algorithm we presented improves the classical methodological approach based on CAE.

ACKNOWLEDGMENTS

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] G. K. Vallis, *Atmospheric and Oceanic Fluid Dynamics: Fundamentals and Large-Scale Circulation*, 2nd ed. (Cambridge University Press, 2017).

[2] D. K. Lilly, "Stratified Turbulence and the Mesoscale Variability of the Atmosphere," Journal of the Atmospheric Sciences **40**, 749–761 (1983).

[3] F. Feraco, R. Marino, A. Pumir, L. Primavera, P. D. Mininni, A. Pouquet, and D. Rosenberg, "Vertical drafts and mixing in stratified turbulence: Sharp transition with froude number," EPL (Europhysics Letters) **123**, 44002 (2018).

[4] F. Feraco, R. Marino, L. Primavera, A. Pumir, P. D. Mininni, D. Rosenberg, A. Pouquet, R. Foldes, E. Lévêque, E. Camporeale, S. S. Cerri, H. Charuvil Asokan, J. L. Chau, J. P. Bertoglio, P. Salizzoni, and M. Marro, "Connecting large-scale velocity and temperature bursts with small-scale intermittency in stratified turbulence," EPL (Europhysics Letters) **135**, 14001 (2021).

[5] R. Marino, F. Feraco, L. Primavera, A. Pumir, A. Pouquet, D. Rosenberg, and P. D. Mininni, "Turbulence generation by large-scale extreme vertical drafts and the modulation of local energy dissipation in stably stratified geophysical flows," Phys. Rev. Fluids **7**, 033801 (2022).

[6] R. Marino, L. Sorriso-Valvo, R. D'Amicis, V. Carbone, R. Bruno, and P. Veltri, "On the Occurrence of the Third-order Scaling in High Latitude Solar Wind," Astroph. Journal **750**, 41 (2012).

[7] R. Marino and L. Sorriso-Valvo, "Scaling laws for the energy transfer in space plasma turbulence," Physics Reports **1006**, 1–144 (2023).

[8] L. Mahrt and N. Gamage, "Observations of turbulence in stratified flow," Journal of Atmospheric Sciences **44**, 1106 – 1121 (1987).

[9] L. Mahrt, "Intermittency of Atmospheric Turbulence," Journal of the Atmospheric Sciences **46**, 79–95 (1988).

[10] D. Lenschow, M. Lothon, S. Mayor, P. Sullivan, and G. Canut, "A comparison of higher-order vertical velocity moments in the convective boundary layer from lidar with in situ measurements and large-eddy simulation," Boundary-Layer Meteorology **143**, 107–123 (2011).

[11] J. Chau, R. Marino, F. Feraco, J. Urco, G. Baumgarten, F.-J. Luebken, W. Hocking, C. Schult, T. Renkwitz, and R. Latteck, "Radar observation of extreme vertical drafts in the polar summer mesosphere," Geophysical Research Letters **48** (2021), 10.1029/2021GL094918.

[12] E. A. D'Asaro, R.-C. Lien, and F. Henyey, "High-Frequency Internal Waves on the Oregon Continental Shelf," Journal of Physical Oceanography **37**, 1956 (2007).

[13] X. Capet, J. C. McWilliams, M. J. Molemaker, and A. F. Shchepetkin, "Mesoscale to Submesoscale Transition in the California Current System. Part I: Flow Structure, Eddy Flux, and Observational Tests," Journal of Physical Oceanography **38**, 29 (2008).

[14] C. Rorai, P. D. Mininni, and A. Pouquet, "Turbulence comes in bursts in stably stratified flows," Phys. Rev. E **89**, 043002 (2014).

[15] N. E. Sujovolsky, G. B. Mindlin, and P. D. Mininni, "Invariant manifolds in stratified turbulence," Phys. Rev. Fluids **4**, 052402 (2019).

[16] D. Rosenberg, A. Pouquet, and R. Marino, "Correlation between buoyancy flux, dissipation and potential vorticity in rotating stratified turbulence," Atmosphere **12**, 157 (2021).

[17] P. Spalart and S. Allmaras, "A one-equation turbulence model for aerodynamic flows," in *30th Aerospace Sciences Meeting and Exhibit* (1992).

[18] S. S. Girimaji, "Partially-Averaged Navier-Stokes Model for Turbulence: A Reynolds-Averaged Navier-Stokes to Direct Numerical Simulation Bridging Method," Journal of Applied Mechanics **73**, 413–421 (2005).

[19] P. R. Spalart, "Detached-eddy simulation," Annual Review of Fluid Mechanics **41**, 181–202 (2009).

[20] K. Julien, S. Legg, J. Mcwilliams, and J. Werne, "Rapidly rotating turbulent rayleigh-bénard convection," Journal of Fluid Mechanics **322**, 243–273 (1996).

[21] A. Pouquet and R. Marino, "Geophysical turbulence and the duality of the energy flow across scales," Phys. Rev. Lett. **111**, 234501 (2013).

[22] R. Marino, A. Pouquet, and D. Rosenberg, "Resolving the paradox of oceanic large-scale balance and small-scale mixing," Phys. Rev. Lett. **114**, 114504 (2015).

[23] D. Rosenberg, A. Pouquet, R. Marino, and P. D. Mininni, "Evidence for Bolgiano-Obukhov scaling in rotating stratified turbulence using high-resolution direct numerical simulations," Physics of Fluids **27**, 055105 (2015).

[24] D. Oks, P. D. Mininni, R. Marino, and A. Pouquet, "Inverse cascades and resonant triads in rotating and stratified turbulence," Physics of Fluids **29**, 111109 (2017).

[25] A. Parente, J. C. Sutherland, L. Tognotti, and P. J. Smith, "Identification of low-dimensional manifolds in turbulent flames," Proceedings of the Combustion Institute **32**, 1579–1586 (2009).

[26] Z. Wan, P. Vlachas, P. Koumoutsakos, and T. Sapsis, "Data-assisted reduced-order modeling of extreme events in complex dynamical systems," PLOS ONE **13** (2018), 10.1371/journal.pone.0197704.

[27] C. E. P. De Jesús and M. D. Graham, "Data-driven low-dimensional dynamic model of kolmogorov flow," Phys. Rev. Fluids **8**, 044402 (2023).

[28] K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton, "Data-driven discovery of coordinates and governing equations," Proceedings of the National Academy of Sciences **116**, 22445–22451 (2019).

[29] S. R. Bukka, R. Gupta, A. R. Magee, and R. K. Jaiman, "Assessment of unsteady flow predictions using hybrid deep learning based reduced-order models," Physics of Fluids **33**, 013601 (2021).

[30] L. Sirovich, "Turbulence and the dynamics of coherent structures. I. Coherent structures," Quart. Appl. Math. **45**, 561–571 (1987).

[31] K. Zdybał, E. Armstrong, J. C. Sutherland, and A. Parente, "Cost function for low-dimensional manifold topology assessment," Scientific Reports **12**, 14496 (2022).

[32] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza, "Model order reduction in fluid dynamics: Challenges and perspectives," in *Reduced Order Methods for Modeling and Computational Reduction* (Springer International Publishing, Cham, 2014) pp. 235–273.

[33] A. T. Mohan and D. V. Gaitonde, "A deep learning based approach to reduced order modeling for turbulent flow control using lstm neural networks," arXiv preprint arXiv:1804.09269 (2018).

[34] W. Chen, Q. Wang, J. S. Hesthaven, and C. Zhang, "Physics-informed machine learning for reduced-order modeling of nonlinear problems," Journal

of computational physics **446**, 110666 (2021).

[35] G. Berkooz, P. Holmes, and J. L. Lumley, "The proper orthogonal decomposition in the analysis of turbulent flows," Annual Review of Fluid Mechanics **25**, 539–575 (1993).

[36] P. Holmes, J. L. Lumley, and G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge Monographs on Mechanics (Cambridge University Press, 1996).

[37] H. Eivazi, S. Le Clainche, S. Hoyas, and R. Vinuesa, "Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows," Expert Systems with Applications **202**, 117038 (2022).

[38] B. Morgan, A. R. Murali, G. Preston, Y. A. Sima, L. A. Marcelo Chamorro, C. Bourantas, R. Torii, A. Mathur, A. Baumbach, M. C. Jacob, S. Karabasov, and R. Krams, "A physics-based machine learning technique rapidly reconstructs the wall-shear stress and pressure fields in coronary arteries," Frontiers in Cardiovascular Medicine **10** (2023), 10.3389/fcvm.2023.1221541.

[39] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," Journal of Fluid Mechanics **656**, 5–28 (2010).

[40] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic Mode Decomposition* (Society for Industrial and Applied Mathematics, Philadelphia, PA, 2016).

[41] G. Berkooz, P. J. Holmes, and J. Lumley, "The proper orthogonal decomposition in the analysis of turbulent flows," Annual Review of Fluid Mechanics **25**, 539–575 (2003).

[42] J. Garicano-Mena, B. Li, E. Ferrer, and E. Valero, "A composite dynamic mode decomposition analysis of turbulent channel flows," Physics of Fluids **31** (2019), 10.1063/1.5119342.

[43] M. O. Williams, P. J. Schmid, and J. N. Kutz, "Hybrid reduced-order integration with proper orthogonal decomposition and dynamic mode decomposition," Multiscale Modeling & Simulation **11**, 522–544 (2013).

[44] E. Marensi, G. Yalnız, B. Hof, and N. B. Budanur, "Symmetry-reduced dynamic mode decomposition of near-wall turbulence," Journal of Fluid Mechanics **954**, A10 (2023).

[45] Y. Cheng and Q. Chen, "Large eddy simulation and dynamic mode decomposition of turbulent mixing layers," Applied Sciences **11** (2021), 10.3390/app112412127.

[46] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem, "The gnat method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows," Journal of Computational Physics **242**, 623 – 647 (2013).

[47] R. Maulik, A. Mohan, B. Lusch, S. Madireddy, P. Balaprakash, and D. Livescu, "Time-series learning of latent-space dynamics for reduced-order model closure," Physica D: Nonlinear Phenomena **405**, 132368 (2020).

[48] D. E. Raveh, "Identification of computational-fluid-dynamics based unsteady aerodynamic models for aeroelastic analysis," Journal of Aircraft **41**, 620–632 (2004).

[49] R. Sarma and R. P. Dwight, "Uncertainty reduction in aeroelastic systems with time-domain reduced-order models," AIAA Journal **55**, 2437–2449 (2017).

[50] M. Raissi and G. Karniadakis, "Deep multi-fidelity gaussian processes," (2016), arXiv:1604.07484 [cs.LG].

[51] Z. Zhang and K. Duraisamy, "Machine learning methods for data-driven turbulence modeling," in *22nd AIAA Computational Fluid Dynamics Conference* (2015).

[52] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," Proceedings of the National Academy of Sciences **113**, 3932–3937 (2016).

[53] M. Schmelzer, R. P. Dwight, and P. Cinnella, "Discovery of algebraic reynolds-stress models using sparse symbolic regression," Flow, Turbulence and Combustion **104**, 579–603 (2020).

[54] K. Duraisamy, Z. J. Zhang, and A. P. Singh, "New approaches in turbulence and transition modeling using data-driven techniques," in *53rd AIAA Aerospace Sciences Meeting* (2015).

[55] E. J. Parish and K. Duraisamy, "A paradigm for data-driven predictive modeling using field inversion and machine learning," Journal of Computational Physics **305**, 758 – 774 (2016).

[56] D. Kochkov, J. A. Smith, A. Alieva, Q. Wang, M. P. Brenner, and S. Hoyer, "Machine learning–accelerated computational fluid dynamics," Proceedings of the National Academy of Sciences **118**, e2101784118 (2021).

[57] A.-T. G. Charalampopoulos and T. P. Sapsis, "Machine-learning energy-preserving nonlocal closures for turbulent fluid flows and inertial tracers," Phys. Rev. Fluids **7**, 024305 (2022).

[58] R. McConkey, E. Yee, and F. S. Lien, "Deep structured neural networks for turbulence closure modeling," Physics of Fluids **34**, 035110 (2022).

[59] K. Srinivasan, M. D. Chekroun, and J. C. McWilliams, "Turbulence closure with small, local neural networks: Forced two-dimensional and β-plane flows," arXiv e-prints , arXiv:2304.05029 (2023).

[60] S. Angriman, P. Cobelli, P. D. Mininni, M. Obligado, and P. Clark di Leoni, "Assimilation of statistical data into turbulent flows using physics-informed neural networks," The European physical journal. E, Soft matter **46**, 13 (2023).

[61] M. Morimoto, K. Fukami, K. Zhang, N. Aditya G., and K. Fukagata, "Convolutional neural networks for fluid flow analysis: toward effective metamodeling and low dimensionalization," Theoretical and Computational Fluid Dynamics **35**, 633–658 (2021).

[62] K. Fukami, T. Nakamura, and K. Fukagata, "Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data," Physics of Fluids **32** (2020), 10.1063/5.0020721.

[63] T. Murata, K. Fukami, and K. Fukagata, "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics," Journal of Fluid Mechanics **882**, A13 (2020).

[64] T. Nakamura, K. Fukami, K. Hasegawa, Y. Nabae, and K. Fukagata, "Convolutional neural network and long short-term memory based reduced order surrogate for minimal turbulent channel flow," Physics of Fluids **33** (2021), 10.1063/5.0039845.

[65] N. A. Khoa Doan, A. Racca, and L. Magri, "Convolutional autoencoder for the spatiotemporal latent representation of turbulence," arXiv e-prints , arXiv:2301.13728 (2023).

[66] H. Salehipour and W. R. Peltier, "Deep learning of mixing by two 'atoms' of stratified turbulence," Journal of Fluid Mechanics **861**, R4 (2019).

[67] Y. Wang, Z. Xie, K. Xu, Y. Dou, and Y. Lei, "An efficient and effective convolutional auto-encoder extreme learning machine network for 3d feature learning," Neurocomputing **174**, 988 – 998 (2016).

[68] F. J. Gonzalez and M. Balajewicz, "Deep convolutional recurrent autoencoders for learning low-dimensional feature dynamics of fluid systems," (2018), arXiv:1808.01346 [math.DS].

[69] J. Xu and K. Duraisamy, "Multi-level Convolutional Autoencoder Networks for Parametric Prediction of Spatio-temporal Dynamics," (2019), arXiv:1912.11114 [physics.comp-ph].

[70] R. King, O. Hennigh, A. Mohan, and M. Chertkov, "From deep to physics-informed learning of turbulence: Diagnostics," (2018), arXiv:1810.07785 [physics.flu-dyn].

[71] J. N. Kutz, "Deep learning in fluid dynamics," Journal of Fluid Mechanics **814**, 1–4 (2017).

[72] Y. Zhu, Y.-H. Tang, and C. Kim, "Learning stochastic dynamics with statistics-informed neural network," Journal of Computational Physics **474**, 111819 (2023).

[73] G. N. Ivey, K. B. Winters, and J. R. Koseff, "Density stratification, turbulence, but how much mixing?" Annual Review of Fluid Mechanics **40**, 169–184 (2008).

[74] P. D. Mininni, D. Rosenberg, R. Reddy, and A. Pouquet, "A hybrid mpi–openmp scheme for scalable parallel pseudospectral computations for fluid turbulence," Parallel Computing **37**, 316–326 (2011).

[75] D. Rosenberg, P. D. Mininni, R. Reddy, and A. Pouquet, "Gpu parallelization of a hybrid pseudospectral geophysical turbulence framework using cuda," Atmosphere **11** (2020), 10.3390/atmos11020178.

[76] N. E. Sujovolsky, P. D. Mininni, and A. Pouquet, "Generation of turbulence through frontogenesis in sheared stratified flows," Physics of Fluids **30**, 086601 (2018).

[77] M. Fontana, O. P. Bruno, P. D. Mininni, and P. Dmitruk, "Fourier continuation method for incompressible fluids with boundaries," Computer Physics Communications **256**, 107482 (2020).

[78] A. Pouquet, D. Rosenberg, and R. Marino, "Linking dissipation, anisotropy, and intermittency in rotating stratified turbulence at the threshold of linear shear instabilities," Physics of Fluids **31**, 105116 (2019), https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/1.5114633/15960255/105116_1_online.pdf.

[79] D. DeMers and G. W. Cottrell, "Non-linear dimensionality reduction," in *Advances in Neural Information Processing Systems 5*, edited by S. J. Han-

son, J. D. Cowan, and C. L. Giles (Morgan-Kaufmann, 1993) pp. 580–587.

[80] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Artificial Neural Networks and Machine Learning – ICANN 2011*, edited by T. Honkela, W. Duch, M. Girolami, and S. Kaski (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011) pp. 52–59.

[81] N. Gunantara, "A review of multi-objective optimization: Methods and its applications," Cogent Engineering **5**, 1502242 (2018).

[82] S. Wang, Y. Teng, and P. Perdikaris, "Understanding and mitigating gradient flow pathologies in physics-informed neural networks," SIAM Journal on Scientific Computing **43**, A3055–A3081 (2021).

[83] P. Clark Di Leoni, K. Agarwal, T. Zaki, C. Meneveau, and J. Katz, "Reconstructing turbulent velocity and pressure fields from under-resolved noisy particle tracks using physics-informed neural networks," Experiments in Fluids **64** (2023), 10.1007/s00348-023-03629-4.

[84] E. Camporeale, G. J. Wilkie, A. Y. Drozdov, and J. Bortnik, "Data-driven discovery of fokker-planck equation for the earth's radiation belts electrons using physics-informed neural networks," Journal of Geophysical Research: Space Physics **127**, e2022JA030377 (2022), e2022JA030377 2022JA030377.

[85] A. Glaws, R. King, and M. Sprague, "Deep learning for in situ data compression of large turbulent flow simulations," Phys. Rev. Fluids **5**, 114602 (2020).

[86] M. Momenifar, E. Diao, V. Tarokh, and A. D. Bragg, "Dimension reduced turbulent flow data from deep vector quantisers," Journal of Turbulence **23**, 232–264 (2022).

[87] A. Pumir, B. I. Shraiman, and E. D. Siggia, "Exponential tails and random advection," Phys. Rev. Lett. **66**, 2984–2987 (1991).

[88] C. Brun, S. Blein, and J.-P. Chollet, "Large-eddy simulation of a katabatic jet along a convexly curved slope. part i: Statistical results," Journal of the Atmospheric Sciences **74** (2017), 10.1175/JAS-D-16-0152.1.

[89] M. Orsi, L. Soulhac, F. Feraco, M. Marro, D. Rosenberg, R. Marino, M. Boffadossi, and P. Salizzoni, "Scalar mixing in homogeneous isotropic turbulence: A numerical study," Phys. Rev. Fluids **6**, 034502 (2021).

[90] J. Vierinen, J. L. Chau, H. Charuvil, J. M. Urco, M. Clahsen, V. Avsarkisov, R. Marino, and R. Volz, "Observing mesospheric turbulence with specular meteor radars: A novel method for estimating second-order statistics of wind velocity," Earth and Space Science **6**, 1171–1195 (2019).

[91] B. Pearson and B. Fox-Kemper, "Log-normal turbulence dissipation in global ocean models," Phys. Rev. Lett. **120**, 094501 (2018).