Computers and Fluids 201 (2020) 104470

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Reducing data-driven dynamical subgrid scale models by physical constraints

Wouter Edeling[a,*], Daan Crommelin[a,b]

[a] Centrum Wiskunde & Informatica, Scientific Computing Group Science Park 123, 1098 XG Amsterdam, the Netherlands
[b] Korteweg-de Vries Institute for Mathematics, University of Amsterdam Science Park 105-107, 1098 XG Amsterdam, the Netherlands

ARTICLE INFO

Article history:
Received 11 October 2019
Revised 7 January 2020
Accepted 6 February 2020
Available online 7 February 2020

Keywords:
Turbulence
Data-driven
Subgrid scale model
Surrogate models

ABSTRACT

Recent years have seen a growing interest in using data-driven (machine-learning) techniques for the construction of cheap surrogate models of turbulent subgrid scale stresses. These stresses display complex spatio-temporal structures, and constitute a difficult surrogate target. In this paper we propose a data-preprocessing step, in which we derive alternative subgrid scale models which are virtually exact for a user-specified set of spatially integrated quantities of interest. The unclosed component of these new subgrid scale models is of the same size as this set of integrated quantities of interest. As a result, the corresponding training data is massively reduced in size, decreasing the complexity of the subsequent surrogate construction.

1. Introduction

Most numerical simulations of turbulent flow only capture a limited portion of all spatial and temporal scales present in the problem, which introduces the need for parameterizations. The effect of the unresolved flow scales enters the resolved-scale equations as an unclosed forcing term, often denoted as the subgrid scale term or the eddy forcing (in a oceanographic context), which is highly complex, dynamic, and shows intricate spatio-temporal correlations. Traditionally, the eddy forcing is approximated by deterministic parameterizations. In the context of geophysical flows, parameterizations are for instance based on the work of Gent-McWilliams [1], or through the inclusion of a tunable (hyper) viscosity term meant to damp the smallest resolved scales of the model [2].

Other sophisticated deterministic parameterizations have also been developed. For instance the authors of [3] use a maximum entropy approach to derive a probability density function for the unresolved scales of a two-dimensional turbulent flow problem, where the time average of the unresolved energy and enstrophy derivatives are constrained to zero. In [4], the transfer of energy and enstrophy in spectral space is analysed for a number of parameterizations, and compared to a high-fidelity reference solution of a two-dimensional turbulent flow case. A deterministic 'energy fixer' scheme is proposed, based on adding a weighted vorticity pattern to the computed vorticity field. In [5], the governing equations for two-dimensional turbulence are modified with a so-called 'thermostat', to drive the energy at the high wave numbers towards an observed statistical average.

Stochastic, data-driven methods for constructing a surrogate for the eddy forcing have risen in popularity over the years. Early contributions to this topic in the context of ocean modelling includes the work of [6], where the eddy-forcing is replaced by a space-time correlated random-forcing process. Furthermore, in [7] a conditional data resampling scheme is proposed to replace the unresolved scales of an ocean model. Other notable examples include the work of [8–10]. Probability density functions (pdfs) of the eddy forcing were constructed using a reference solution, conditioned on a suitable, resolved-scale variable which showed high-correlation with respect to the reference eddy forcing. More recently, approaches involving machine-learning have also found application to (two-dimensional) turbulence. In the work of [11], the eddy-forcing is parameterized via a feed-forward neural net, and in [12,13], neural networks are applied to predict the complete state of a general circulation model. Finally, the authors of [14] use a convolutional neural net to predict the subgrid scale stresses of a quasi-geostrophic ocean model.

Before any such method can be used, a sufficiently large set of reference data needs to be available. Common practice is to simply i) collect a database of snapshots of either the eddy forcing or the state of the reference model, and ii), proceed with the construction of some chosen surrogate method. The novelty of this paper is its position in between these two steps. We propose to precede

[*] Corresponding author.
E-mail address: wouter.edeling@cwi.nl (W. Edeling).

https://doi.org/10.1016/j.compfluid.2020.104470
0045-7930/© 2020 Elsevier Ltd. All rights reserved.

the surrogate construction by an additional procedure that replaces the eddy forcing with a new subgrid scale term which: i) is tailor-made to capture several spatially-integrated statistics of interest, ii) strikes a balance between physical insight and data-driven modelling, and iii) significantly reduces the amount of training data that is needed. Due to the latter point, we denote these new source terms as 'reduced' subgrid scale terms. Instead of creating a surrogate for an evolving field, we now only require a surrogate model for one scalar time series per statistical quantity-of-interest. Hence we no longer require a surrogate with the ability to capture the complex spatial correlation structure of the eddy forcing.

This paper is organised as follows. In chapter 2, we discuss the 2D turbulent flow problem we consider, and the discretisation method that is employed. The subsequent section outlines the derivation of the reduced subgrid scale terms. Section 4 shows the results of these new terms, as measured in their ability of capturing different sets of hand-selected quantities of interest. Finally, we conclude in Section 5.

2. Governing equations

We study the same model as in [3,15], i.e. the forced-dissipative vorticity equations for two-dimensional incompressible flow. The governing equations read

$$\frac{\partial\omega}{\partial t} + J(\Psi,\omega) = \nu\nabla^2\omega + \mu(F-\omega),$$
$$\nabla^2\Psi = \omega. \tag{1}$$

Here, ω is the vertical component of the vorticity, defined from the curl of the velocity field $\mathbf{V}$ as $\omega := \mathbf{e}_3 \cdot \nabla \times \mathbf{V}$, where $\mathbf{e}_3 := (0, 0, 1)^T$. The stream function Ψ relates to the horizontal velocity components by the well-known relations $u = -\partial\Psi/\partial y$ and $v = \partial\Psi/\partial x$. As in [3], the forcing term is chosen as the single Fourier mode $F = 2^{3/2}\cos(5x)\cos(5y)$. The system is fully periodic in x and y directions over a period of $2\pi L$, where L is a user-specified length scale, chosen as the earth's radius ($L = 6.371 \times 10^6[m]$). The inverse of the earth's angular velocity Ω^{-1} is chosen as a time scale, where $\Omega = 7.292 \times 10^{-5}[s^{-1}]$. This choice of Ω^{-1} allows us to express a simulation time period of a single 'day' as $24 \times 60^2 \times \Omega \approx 2\pi$ non-dimensional time units, which allows for an intuitive specification of the decay time of the diffusion and forcing term through the values of ν and μ. Since, once we are given these scaling factors, (1) is non-dimensionalized, and solved using values of ν and μ chosen such that a Fourier mode at the smallest retained spatial scale is exponentially damped with an e-folding time scale of 5 and 90 days respectively, see (4). For more details on the numerical setup we refer to [3]. Furthermore, our Python source code for (1) can be downloaded from [16].

Finally, the key term in (1) is the Jacobian, i.e. the nonlinear advection term defined as

$$J(\Psi,\omega) := \frac{\partial\Psi}{\partial x}\frac{\partial\omega}{\partial y} - \frac{\partial\Psi}{\partial y}\frac{\partial\omega}{\partial x}. \tag{2}$$

It is this term that leads to the need for a closure model when (1) is discretized on a relatively coarse grid which lacks the resolution to capture all turbulent eddies.

2.1. Discretization

We solve (1) by means of a spectral method, where we apply a truncated Fourier expansion:

$$\omega(x,y,t) \approx \tilde\omega(x,y,t) = \sum_{\mathbf{k}} \hat\omega_{\mathbf{k}}(t)e^{i(k_1x+k_2y)},$$
$$\Psi(x,y,t) \approx \tilde\Psi(x,y,t) = \sum_{\mathbf{k}} \hat\Psi_{\mathbf{k}}(t)e^{i(k_1x+k_2y)}. \tag{3}$$

The sum is taken over the components k_1 and k_2 of the wave number vector $\mathbf{k} := (k_1, k_2)^T$, and $-K' \le k_j \le K'$, $j = 1, 2$. These decompositions are inserted in (1), and solved for the Fourier coefficients $\hat\omega_{\mathbf{k}}$, $\hat\Psi_{\mathbf{k}}$ by means of the real Fast Fourier Transform. To avoid the aliasing problem in the nonlinear term (2), we use the well-known 2/3 rule, such that in practice the maximum resolved wave number is K, where $K \le 2K'/3$ [17].[1] Remember that ν and μ are chosen such that a Fourier mode at the finest retained scale is damped with an e-folding time scale of 5 and 90 days. This leads to the following expressions for ν and μ, where the former is a function of K:

$$\nu = \frac{1}{24\cdot 60^2}\frac{1}{\Omega}\frac{1}{K^2}\frac{1}{5} \quad \text{and} \quad \mu = \frac{1}{24\cdot 60^2}\frac{1}{\Omega}\frac{1}{90}, \tag{4}$$

[3]. Like the 2/3 rule, the use of a (hyper) viscosity term is a solution for removing high-wavenumber contributions from the solution [18]. Typically the viscosity is tuned in order to remove a build up of energy near the smallest scales, i.e. so-called spectral blocking. Another option is high-order Fourier smoothing, which keeps a portion of the Fourier modes beyond the cutoff from the 2/3 rule, see e.g. [19,20].

To advance the solution in time we use the second-order accurate AB/BDI2 scheme, which results in the following discrete system of equations [17]

$$\frac{3\hat\omega_{\mathbf{k}}^{n+1} - 4\hat\omega_{\mathbf{k}}^{n} + \hat\omega_{\mathbf{k}}^{n-1}}{2\Delta t} + 2\hat J_{\mathbf{k}}^{n} - \hat J_{\mathbf{k}}^{n-1} = -\nu k^2\hat\omega_{\mathbf{k}}^{n+1} + \mu\left(\hat F_{\mathbf{k}} - \hat\omega_{\mathbf{k}}^{n+1}\right),$$
$$-k^2\hat\Psi_{\mathbf{k}}^{n+1} - \hat\omega_{\mathbf{k}}^{n+1} = 0. \tag{5}$$

Here, Δt is the time step and $\hat J_{\mathbf{k}}^{n}$ is the Fourier coefficient of the Jacobian at time level n, computed with the pseudo spectral technique, and $k^2 := k_1^2 + k_2^2$.

2.2. Multiscale decomposition

As in [3], we apply a spectral filter in order to decompose the full reference solution into a resolved ($\mathcal{R}$) and an unresolved component ($\mathcal{U}$), i.e. we use

$$\hat\omega_{\mathbf{k}}^{\mathcal{R}} = \hat{\mathcal{T}}^{\mathcal{R}}\hat\omega_{\mathbf{k}}, \qquad \hat\omega_{\mathbf{k}}^{\mathcal{U}} = \hat{\mathcal{T}}^{\mathcal{U}}\hat\omega_{\mathbf{k}}, \tag{6}$$

where the projection operators $\hat{\mathcal{T}}^{\mathcal{R}}$ and $\hat{\mathcal{T}}^{\mathcal{U}}$ are depicted in Fig. 1. Note that the full projection operator $\hat{\mathcal{T}} := \hat{\mathcal{T}}^{\mathcal{R}} + \hat{\mathcal{T}}^{\mathcal{U}}$ also removes wave numbers due to the 2/3 rule.

Applying the resolved projection operator to the governing Eq. (1) results in the following resolved-scale transport equation

$$\frac{\partial\omega^{\mathcal{R}}}{\partial t} + \mathcal{T}^{\mathcal{R}}J(\Psi,\omega) = \nu\nabla^2\omega^{\mathcal{R}} + \mu\left(F^{\mathcal{R}} - \omega^{\mathcal{R}}\right). \tag{7}$$

Here, $\mathcal{T}^{\mathcal{R}}$ is the filter in physical space corresponding to $\hat{\mathcal{T}}^{\mathcal{R}}$. In practice we only use $\hat{\mathcal{T}}^{\mathcal{R}}$ since we solve the equations in spectral space, see (5). As mentioned, the key term is the Jacobian (2), since due to its non linearity, $\mathcal{T}^{\mathcal{R}}J(\Psi,\ \omega) \ne \mathcal{T}^{\mathcal{R}}J\left(\Psi^{\mathcal{R}},\ \omega^{\mathcal{R}}\right)$. We therefore write

$$\mathcal{T}^{\mathcal{R}}r := \mathcal{T}^{\mathcal{R}}\left[J(\Psi,\omega) - J\left(\Psi^{\mathcal{R}},\ \omega^{\mathcal{R}}\right)\right] \tag{8}$$

such that $\mathcal{T}^{\mathcal{R}}r$ is the exact subgrid-scale term, commonly referred to as the 'eddy forcing' [6]. We use the notation $\bar r := \mathcal{T}^{\mathcal{R}}r$ for the sake of brevity. The resolved-scale Eq. (7) can now be written as

$$\frac{\partial\omega^{\mathcal{R}}}{\partial t} + \mathcal{T}^{\mathcal{R}}J\left(\Psi^{\mathcal{R}},\omega^{\mathcal{R}}\right) = \nu\nabla^2\omega^{\mathcal{R}} + \mu\left(F^{\mathcal{R}} - \omega^{\mathcal{R}}\right) - \bar r. \tag{9}$$

A snapshot of the resolved vorticity $\omega^{\mathcal{R}}$ and corresponding resolved eddy forcing $\bar r$ is depicted in Fig. 2. Notice the abundence of small-scale structures in the eddy forcing compared to the vorticity field.

[1] We use $N \times N$ grids, with an even $N = 2^p$ (e.g. $p = 7$), such that $N = 2K'$ [17].

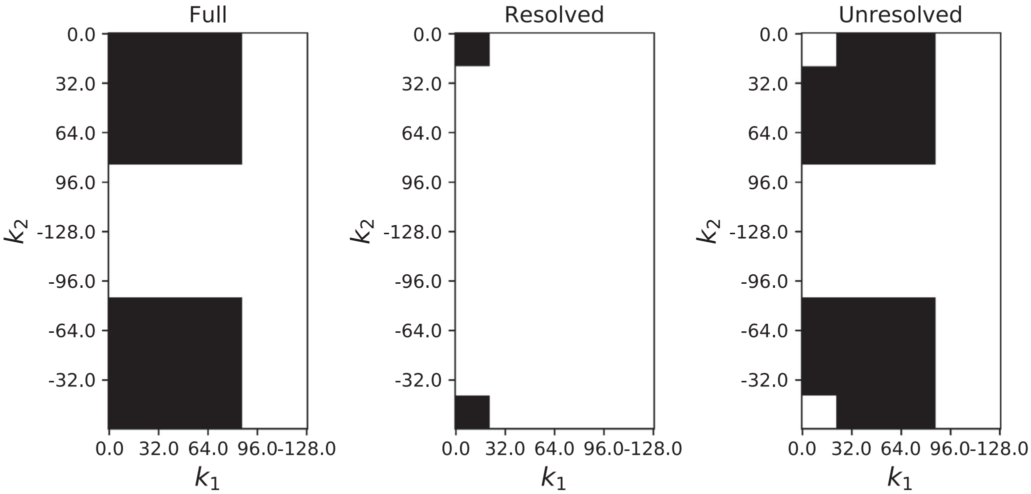

Fig. 1. Example of spectral filters (black=1, white=0) for the full, resolved and unresolved solutions. Due to the fact that we use the real FFT algorithm, only part of the spectrum is computed, as Fourier coefficients with opposite values of $\mathbf{k}$ are complex conjugates in order to enforce real ω and Ψ fields [17].

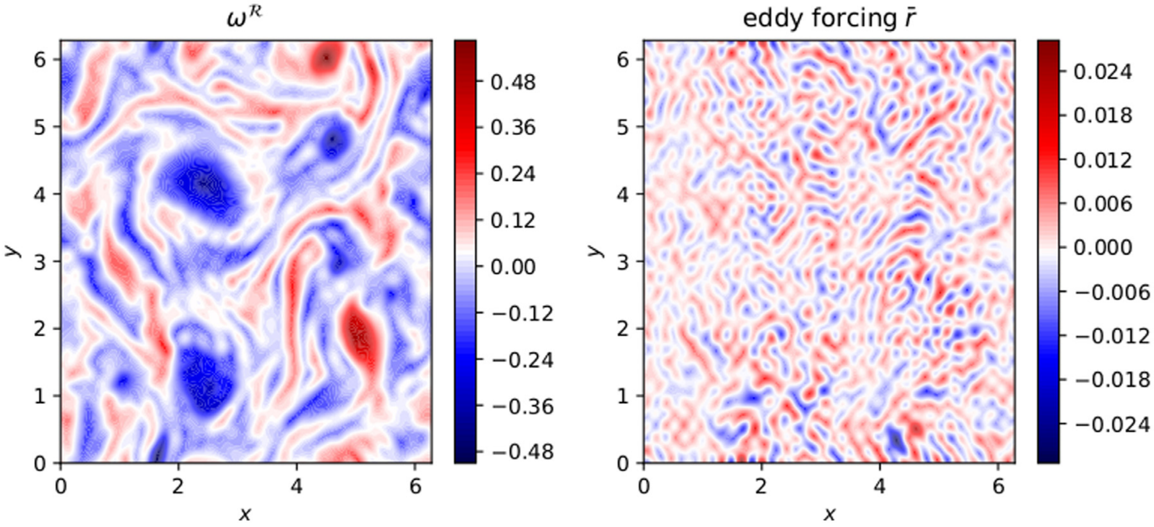

Fig. 2. A snapshot of the exact, reference vorticity field $\omega^{\mathcal{R}}$ and the corresponding eddy forcing.

2.3. Prediction of statistical QoIs

Our goal is to integrate (9) in time, such that we can compute the long-term climate statistics of a set of chosen Quantities of Interest (QoIs): $Q = \{Q_1(t), \ldots, Q_d(t)\}$. Consider a general time-dependent quantity of interest $Q_i(t)$, normalised by the area of the flow domain, as

$$Q_i(t) = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} q_i(\omega^{\mathcal{R}}, \psi^{\mathcal{R}}; x, y, t)\ \mathrm{d}x\mathrm{d}y, \quad i = 1, \ldots, d. \tag{10}$$

where q_i is some function of the primitive variables $\omega^{\mathcal{R}}$ and $\psi^{\mathcal{R}}$. Then, Q_i is governed by the following ordinary differential equation (ODE):

$$\frac{\mathrm{d}Q_i}{\mathrm{d}t} = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \frac{\partial q_i}{\partial \omega^{\mathcal{R}}}\frac{\partial \omega^{\mathcal{R}}}{\partial t} + \frac{\partial q_i}{\partial \psi^{\mathcal{R}}}\frac{\partial \psi^{\mathcal{R}}}{\partial t}\ \mathrm{d}x\mathrm{d}y$$
$$= \left(\frac{\partial q_i}{\partial \omega^{\mathcal{R}}}, \frac{\partial \omega^{\mathcal{R}}}{\partial t}\right) + \left(\frac{\partial q_i}{\partial \psi^{\mathcal{R}}}, \frac{\partial \psi^{\mathcal{R}}}{\partial t}\right), \quad i = 1, \ldots, d. \tag{11}$$

Here, we employ the short-hand

$$(\alpha, \beta) = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \alpha\beta\ \mathrm{d}x\mathrm{d}y. \tag{12}$$

From now on, we will assume that every full-field quantity (i.e. a quantity dependent upon x and y), is resolved, unless otherwise specified, and we drop the superscript $\mathcal{R}$. Several interesting physical quantities are governed by (11), depending on the choice of q_i. For instance, setting $q_i = \mathbf{V}\cdot\mathbf{V}/2$, where $\mathbf{V}$ is the vector containing the velocity components in x and y direction, yields the governing equation of the energy $Q_i = E$. In our particular case, this can be rewritten as $E := -(\psi, \omega)/2$ (see Appendix A). The ODE for E then follows from (11) as

$$\frac{\mathrm{d}E}{\mathrm{d}t} = -\frac{1}{2}\left(\psi, \frac{\partial\omega}{\partial t}\right) - \frac{1}{2}\left(\omega, \frac{\partial\psi}{\partial t}\right) = -\frac{1}{2}\left(\psi, \frac{\partial\omega}{\partial t}\right)$$
$$-\frac{1}{2}\left(\nabla^2\psi, \frac{\partial\psi}{\partial t}\right) = -\frac{1}{2}\left(\psi, \frac{\partial\omega}{\partial t}\right) - \frac{1}{2}\left(\psi, \nabla^2\frac{\partial\psi}{\partial t}\right)$$
$$= -\frac{1}{2}\left(\psi, \frac{\partial\omega}{\partial t}\right) - \frac{1}{2}\left(\psi, \frac{\partial}{\partial t}\nabla^2\psi\right) = -\left(\psi, \frac{\partial\omega}{\partial t}\right) \tag{13}$$

In the third equality, we made use of the self-adjoint nature of the Laplace operator, i.e. $\left(\nabla^2\alpha, \beta\right) = \left(\alpha, \nabla^2\beta\right)$.

Alternatively, when $q_i = \omega^n/n$, for $n = 1, 2, 3, \ldots$, the governing equations for Q_i follow directly from (11) as

$$\frac{\mathrm{d}Q_i}{\mathrm{d}t} = \left(\omega^{n-1}, \frac{\partial \omega}{\partial t}\right). \tag{14}$$

The most well-known QoI of this class is the enstrophy, defined as $Z := (\omega, \omega)/2$.

3. Deriving QoI-tailored subgrid scale models

Ultimately, our goal is to construct a surrogate for $\bar{r}$, using a database of reference snapshots in time. One possible approach is to then attempt to directly construct a surrogate for the full-field eddy forcing [11,21]. We will take a different approach. Instead of directly creating a (complex) full-field surrogate we will first perform a data-preprocessing step designed to simplify the surrogate construction. Note that our QoI Q_i are spatially integrated quantities. This implies that we can replace the exact eddy forcing with a simpler alternative that:

1. Yields vorticity fields which ‘track’ the reference values of the QoIs, i.e. where $Q_i^{ref}(t) - Q_i(t)$ is small for all t within the training period. Here, Q_i^{ref} is the reference value of Q_i, e.g. computed from (5) using the full projection operator displayed in Fig. 1.
2. Has an unclosed component which is of the same dimension as the QoI we attempt to approximate.

The second point implies that, instead of creating a surrogate for an evolving field, we now only need to train a surrogate model on one scalar time series per Q_i. For this reason we apply the adjective *reduced* to these new source terms, and to all quantities that derive from them, i.e. the reduced training data and reduced surrogates.

Initial results along these lines can be found in [15]. These results were specific to the energy E and enstrophy Z, and employed *ad-hoc* parameterizations in the reduced subgrid scale term. Here we develop a generalised, less *ad-hoc* method to compute reduced subgrid scale terms for any QoI set $Q = \{Q_1, \ldots, Q_d\}$ that follows (11). We begin with the following expansion for the reduced eddy forcing term

$$\underline{r} = \sum_{i=1}^{d} \tau_i(t) P_i(x, y, t). \tag{15}$$

Note that we denote the reduded eddy forcing by $\underline{r}$, whereas the exact eddy forcing is $\bar{r}$. The τ_i are the unclosed time series we wish to extract from the training data, and the $P_i^{\mathcal{R}}$ are resolved, full-field ‘patterns’. For reasons explained shortly, we propose a separate expansion for the P_i:

$$P_i = T_{i,1}(x, y; t) - \sum_{j=2}^{d} c_{i,j}(t) T_{i,j}(x, y; t), \quad i \in \{1, \ldots, d\} \tag{16}$$

where the T_{ij} are user-specified resolved quantities, e.g. ω or ψ. The time-dependent coefficients $c_{i,j}$ are also resolved, and computed by imposing certain orthogonality conditions.

For illustrative purposes, let us assume we have 3 QoI, i.e. $d = 3$. In this case there are 6 unknown $c_{i,j}$ in (16). Also note that each right-hand side of the 3 Q_i ODEs (11) contains an inner product involving $\partial\omega/\partial t$. We can insert the vorticity equations (9) here, such that (amongst others), a $(\partial q_i/\partial\omega, \bar{r})$ term appears in each ODE. Let us now introduce the short-hand notation $V_i := \partial q_i/\partial\omega$, such that $(V_i, \bar{r})$ is the subgrid scale term in the transport equations of our QoI. Thus, since we replace $\bar{r}$ by $\underline{r}$ (given by (15)), and that we aim to track Q_i^{ref} via $\underline{r}$, the only terms by which we can do so directly have the following form

$$(V_i, \underline{r}) = \tau_1(V_i, P_1) + \tau_2(V_i, P_2) + \tau_3(V_i, P_3), \quad i \in \{1, 2, 3\}. \tag{17}$$

This gives a total of 9 different $\tau_k(V_i, P_k)$ terms. As it stands now, all three τ_k influence each QoI Q_i through the above $(V_i, \underline{r})$ source term. However, for controlling Q_i, we would prefer to train τ_1 solely for tracking the reference value of Q_1, τ_2 solely for Q_2, and so on. By examining (17), we see that this can be achieved by imposing the following orthogonality conditions:

$$\left(V_i, P_j\right) = 0 \ \text{ if } \ i \neq j. \tag{18}$$

In the case of 3 QoI, (18) gives us 6 linear equations, closing the system for the 6 unknown $c_{i,j}$ of (16). When we group these equations by P_i, and insert (16), we get 3 linear systems:

$$\begin{bmatrix} \left(V_{j_1}, T_{i,2}\right) & \left(V_{j_1}, T_{i,3}\right) \\ \left(V_{j_2}, T_{i,2}\right) & \left(V_{j_2}, T_{i,3}\right) \end{bmatrix} \begin{bmatrix} c_{i,2} \\ c_{i,3} \end{bmatrix} = \begin{bmatrix} \left(V_{j_1}, T_{i,1}\right) \\ \left(V_{j_2}, T_{i,1}\right) \end{bmatrix}, \quad i \in \{1, 2, 3\}. \tag{19}$$

Here, the index set $\{j_1, j_2\}$ is defined as $\{j_1, j_2\} := \{1, \cdots, d\} \backslash \{i\}$, e.g. as $\{1, 3\}$ for $i = 2$ and $d = 3$. If we are tracking two QoI, (19) reduces to 2 uncoupled algebraic equations that can be solved for $c_{i,2}$ as

$$c_{i,2} = \frac{\left(V_j, T_{i,1}\right)}{\left(V_j, T_{i,2}\right)}, \quad i \in \{1, 2\}, \quad j = \{1, 2\} \backslash \{i\}. \tag{20}$$

Finally, the general case for tracking d QoI leads to d linear systems $A_i c_i = b_i$, where

$$A_i = \begin{bmatrix} \left(V_{j_1}, T_{i,2}\right) & \left(V_{j_1}, T_{i,3}\right) & \ldots & \left(V_{j_1}, T_{i,d}\right) \\ \vdots & \ddots & & \vdots \\ \left(V_{j_{d-1}}, T_{i,2}\right) & \left(V_{j_{d-1}}, T_{i,3}\right) & \ldots & \left(V_{j_{d-1}}, T_{i,d}\right) \end{bmatrix} \in \mathbb{R}^{d-1 \times d-1},$$

$$c_i = \begin{bmatrix} c_{i,2} \\ \vdots \\ c_{i,d-1} \end{bmatrix} \in \mathbb{R}^{d-1}, \quad \text{and} \quad b_i = \begin{bmatrix} \left(V_{j_1}, T_{i,1}\right) \\ \vdots \\ \left(V_{j_{d-1}}, T_{i,1}\right) \end{bmatrix} \in \mathbb{R}^{d-1}$$

$$i \in \{1, \ldots, d\}, \quad j \in \{1, \ldots, d\} \backslash \{i\}. \tag{21}$$

We reiterate that these linear systems involve only resolved quantities, and can therefore also be solved at any time in a predictive setting where we do not have access to training data. With (18) now satisfied, the Q_i subgrid scale term (17) now consist of just 1 term

$$(V_i, \underline{r}) = \tau_i(V_i, P_i), \quad i \in \{1, \ldots, d\}. \tag{22}$$

The physical insight we gained is that for Q_i, our eddy-forcing parameterization (15) results in a Q_i source term containing (V_i, P_i), which can act to either dissipate or produce Q_i at any given time. Therefore, in light of our goal to track the reference value Q_i^{ref} during the training phase, the task of τ_i must be to switch on dissipation or production conditional on the value of $\Delta Q_i := \mathcal{T}^{\mathcal{R}} Q_i^{ref} - Q_i$. Hence, the value of τ_i should depend on ΔQ_i.

In [15], we used *ad-hoc* parameterizations of the form $\tau_i \propto \tanh(\Delta Q_i/Q_i) \times \mathrm{sign}((V_i, P_i))$, which ensured that τ_i has the correct sign, given a value for ΔQ_i. Finding the proportionality constant requires tuning, although for $Q = \{E, Z\}$ the simple choice of 1 produces good results. However, this is unlikely to generalise to arbitrary sets Q, which we confirmed with numerical experiments which included $(\omega^2, \omega)/3$ in Q. Here we opt for a different parameterization which drops the *ad-hoc* functional tanh assumption. To find the value of τ_i directly, we equate (22) to a simple linear relaxation term,

$$\tau_i(V_i, P_i) = \Delta Q_i, \quad i \in \{1, \ldots, d\}. \tag{23}$$

Thus, if we set $\tau_i = \Delta Q_i/(V_i, P_i)$, we obtain a training data set for the τ_i which relaxes Q_i towards Q_i^{ref}, if $(V_i, P_i) \neq 0$. In the next section we will show for a number of cases that we obtain a τ_i training data set with a (near) perfect accuracy for our QoI, that is reduced in size by a factor N^2/d compared to the full-field eddy-forcing data from (8). Note that N is the spatial discretization of the resolved model in one dimension (typical values are 2^6, 2^7 or higher). Hence, we obtain a reduction in the number of degrees of freedom of several orders of magnitude, and it is therefore not a trivial matter that we retain accuracy of our QoIs.

4. Results

We now showcase our method on a number of examples. The main results can be reproduced by downloading the Python script from [22].

4.1. Numerical setup

As a reference solution, we will solve (5) on a 256 × 256 computational grid. This grid corresponds to Fourier modes with $K' = 128$ (see (3)), which is reduced to a maximum resolved wave number K of 85 after applying the 2/3 rule [17]. In this case $\nu = 4.394 \cdot 10^{-6}$ and $\mu = 1.764 \cdot 10^{-3}$, see (4). For a resolved model, we use a projection operator $\hat{\mathcal{T}}^{\mathcal{R}}$ with a maximum resolved wave number of 21 (which corresponds to the accuracy of a 64 × 64 spatial discretization). We could keep ν as stated above for the resolved model, although for this value of ν, spectral power tends to accumulate at the cutoff wave number, which results in noisy vorticity fields. It is common practice to add an ‘eddy-viscosity’ model to introduce additional damping [3]. Note that in a geophysical context (to which this work belongs), an ‘eddy viscosity’ model amounts to an increased value of ν, and should not be confused with the more eloborate eddy-viscosity models as found in the Reynolds-Averaged Navier-Stokes simulation paradigm. In our case this model is introduced by setting $K = 21$ in (4), in order to obtain an increased value for the viscosity, i.e. $\nu = 7.198 \cdot 10^{-5}$. Unless otherwise stated, we will use this value for the viscosity.

As an initial condition we use

$$\omega(x, y, 0) = \sin(4x)\sin(4y) + 0.4\cos(3x)\cos(3y) + 0.3\cos(5x)\cos(5y) + 0.02\sin(x) + 0.02\cos(y), \tag{24}$$

[4], from which we first spin up the model for 250 days before performing the analysis detailed below.

4.2. Tracking 2 QoI

We will first demonstrate our approach by deriving a source term $\underline{r}$ which tracks the reference energy E and enstrophy Z during training. The choice of the Q_i dictates the choice of the V_i, due to $V_i := \partial q_i / \partial\omega$. Since the ODEs for E and Z are given by (13) and (14) (with $n = 2$), we must set $V_1 = -\Psi$ and $V_2 = \omega$. On the other hand, the basis functions $T_{i,j}$ of the orthogonal patterns (16) are a modelling choice. For simplicity, throughout the paper we will restrict the $T_{i,j}$ to the same set of terms that make up the V_i. Starting always with $T_{i,1} = V_i$, we get the following two-term expansion for the patterns P_i

$$P_1 = -\Psi - c_{1,2}\omega \quad \text{and} \quad P_2 = \omega + c_{2,2}\Psi. \tag{25}$$

Through (20), we find the values of the coefficients c_{ij}.

$$c_{1,2} = -\frac{(\omega, \Psi)}{(\omega, \omega)} = \frac{E}{Z} \quad \text{and} \quad c_{2,2} = -\frac{(\Psi, \omega)}{(\Psi, \Psi)} = \frac{E}{S}, \tag{26}$$

where we have defined S as the squared integrated stream function; $S := (\Psi,\ \Psi)/2$. Thus, the total source term in the vorticity equation is

$$\underline{r} = -\tau_1\left(\Psi + \frac{E}{Z}\cdot\omega\right) + \tau_2\left(\omega + \frac{E}{S}\cdot\Psi\right). \tag{27}$$

The expressions for τ_1 and τ_2 are found via (23), which amounts to examining the source term in the E and Z equations produced by (27). In this particular case we find

$$\tau_1(-\Psi, P_1) = 2\tau_1\left[S - \frac{E^2}{Z}\right] = \Delta E,$$
$$\tau_2(\omega, P_2) = 2\tau_2\left[Z - \frac{E^2}{S}\right] = \Delta Z, \tag{28}$$

for the source terms in the E and Z equation respectively, which are equated to ΔE and ΔZ. Here, $\Delta E := \mathcal{T}^{\mathcal{R}}E^{ref} - E$ and $\Delta Z := \mathcal{T}^{\mathcal{R}}Z^{ref} - Z$ are data extracted from the training database, where e.g. $T^{\mathcal{R}}Z^{ref}$ is the reference enstrophy computed with the projected reference vorticity $\mathcal{T}^{\mathcal{R}}\omega^{ref}$. At any time t_n during training, the values of τ_1 and τ_2 are found by

$$\tau_{1,n} = \frac{1}{2}\left[\frac{\Delta E_n}{S_n - E_n^2/Z_n}\right] \quad \text{and} \quad \tau_{2,n} = \frac{1}{2}\left[\frac{\Delta Z_n}{Z_n - E_n^2/S_n}\right]. \tag{29}$$

Finally, after gathering all terms we get the following expression for the (full-field) reduced eddy forcing at t_n

$$\underline{r}_n = -\frac{1}{2}\left[\frac{\Delta E_n}{S_n - E_n^2/Z_n}\right]\left(\Psi_n + \frac{E_n}{Z_n}\cdot\omega_n\right) + \frac{1}{2}\left[\frac{\Delta Z_n}{Z_n - E_n^2/S_n}\right]\left(\omega_n + \frac{E_n}{S_n}\cdot\Psi_n\right) \tag{30}$$

The analysis above is mainly for illustrative purposes. During simulation, we do not need to expand every term, and instead can directly compute terms like $(-\Psi, P_1)$ using numerical quadrature, see also Section 4.4.

Eq. (30) is a near-perfect substitute for the exact eddy forcing $\bar{r}$ (8) as far as the computation of E and Z is concerned, while the dimension of the unclosed component is reduced from 64^2 in $\bar{r}$, down to 2 in $\underline{r}$. To demonstrate this, we ran both the reference and the low-resolution model with (30) for 10 simulated years to gather E and Z data. The probability density functions (pdfs) for these data are shown in Fig. 3, which display a virtual overlap for the reference and low-resolution resolved model. The results of an eddy viscosity model, which is clearly too diffusive, are also shown for reference. Finally, the reduced training data, i.e. the ΔE and ΔZ time series are shown in Fig. 4.

These results demonstrate that, as far as our QoIs are concerned, nothing is lost compared to the system forced by the exact eddy forcing. Hence, we do not need to construct a (data-driven) surrogate for the full-field eddy forcing, which displays complex spatial correlations. Instead, the two scalar time series (29) provide statistically equivalent training data, i.e. they yield the same E and Z pdfs as the exact eddy forcing.

4.2.1. Energy and enstrophy spectra

Let us also examine the spectral breakdown of the energy and enstrophy, i.e. the spectra in wavenumber space. To compute these spectra, we map each wave number pair $(k_1,\ k_2)$ of the Fourier expansion (3) to a real number via the Euclidean distance, and add all energy and enstrophy contributions that fall within the bins given by

$$k - \frac{1}{2} \le \sqrt{k_1^2 + k_2^2} < k + \frac{1}{2}, \quad k = 0, 1, \ldots, \mathrm{ceil}\left(\sqrt{2}K\right). \tag{31}$$

[3]. Here, ceil(·) is the ceiling function which rounds a real number to the nearest integer above. Remember that K is the cutoff wave number, and the $\sqrt{2}$ term in the maximum value of k is a consequence of the square truncation in the spectral filter $\hat{\mathcal{T}}$, such that the maximum Euclidean distance is $\sqrt{K^2 + K^2} = \sqrt{2}K$. A further consequence of the square truncation is that between

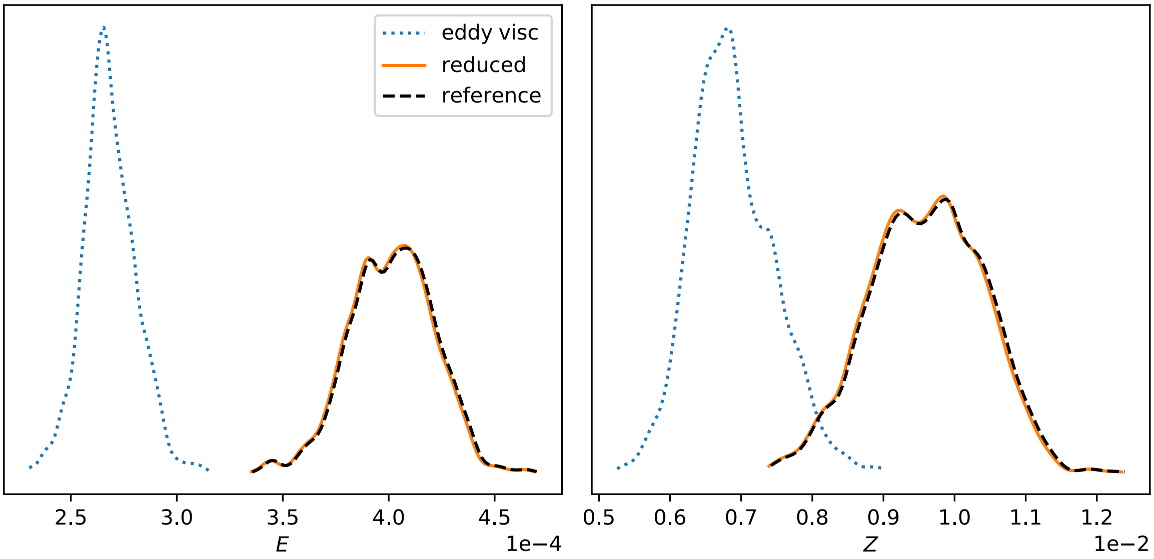

Fig. 3. The pdfs of the energy E (left), and the enstrophy Z (right) of both the reference (striped) and the low-resolution model (solid) with (30) for the eddy forcing. It is important to note that training data was used in the computation of the unclosed terms of (30), we are not creating a surrogate model here. The results of an eddy viscosity model are also included (dotted).

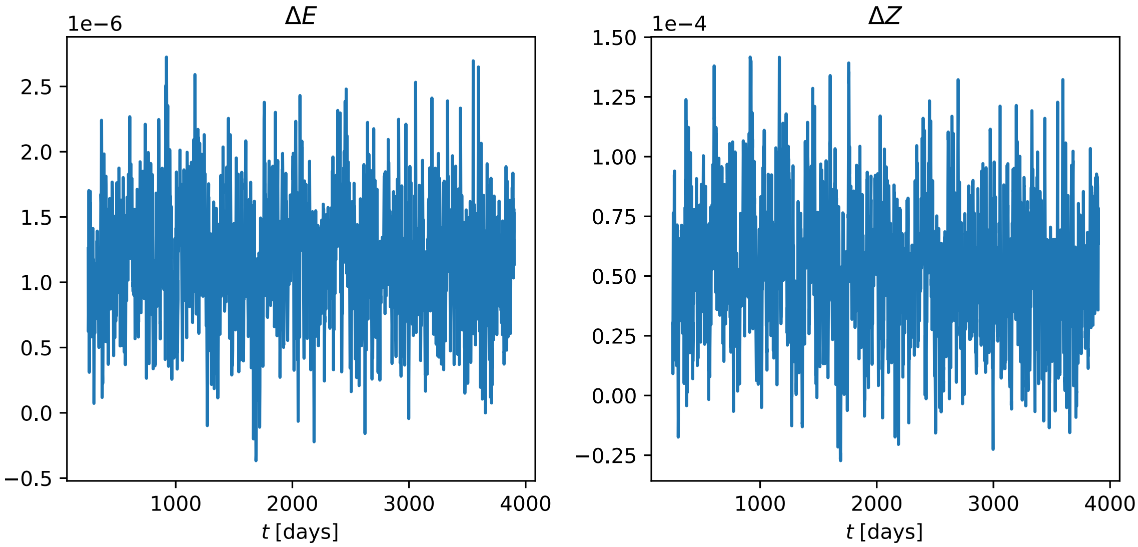

Fig. 4. The reduced training data.

$k = K$ and $k = \text{ceil}(\sqrt{2}K)$, all spectra of the resolved model are inaccurate (i.e. too steep), due to the fact that, in this range of k, $\hat{\mathcal{T}}^{\mathcal{R}}$ simply does not contain all (k_1, k_2) pairs that fall in the bins $[k-1/2, k+1/2]$, see Fig. 5. This inaccuracy also occurs if we use the exact eddy forcing (8), since it too is subject to the same truncation.

The spectra, averaged over the same 10 year simulation period from the preceding section, are shown in Fig. 6. The two vertical lines denote the range $k \in \left[K, \sqrt{2}K\right]$, where indeed the spectra are too steep compared to the reference. This is the case for both the eddy viscosity model and the reduced model, which virtually overlap in the high wave number region. From Fig. 6 it thus becomes clear that the reduced eddy forcing (30) modifies the response in the low wave numbers. Since we track QoI averaged over all (k_1, k_2), there is no explicitly imposed requirement that the spectrum of the reduced model will coincide with that of the reference for every value of k. That said, if a certain subrange of k is of particular interest, our approach can be easily modified in a scale-selective manner to target specific wave numbers only.

4.2.2. Concentrating on a specific wave number range

Instead of $V := \{-\Psi, \omega\}$, here we use $V := \{-\mathcal{T}^{\mathcal{K}}\Psi, \mathcal{T}^{\mathcal{K}}\omega\}$, where $\mathcal{T}^{\mathcal{K}}$ corresponds to the spectral filter $\hat{\mathcal{T}}^{\mathcal{K}} = \hat{\mathcal{T}}^{\mathcal{K}}\left(K, \text{ceil}(\sqrt{2}K)\right)$ shown in Fig. 7, which has the value of 1 only for $-K \le k \le \text{ceil}\left(\sqrt{2}K\right)$. When we further define our target QoIs as $\Delta Q_i := \mathcal{T}^{\mathcal{K}}(Q_i^{ref} - Q_i)$, the effect of the reduced eddy forcing is concentrated in the indicated subrange of k. If we repeat the 10 year simulation, for the reference and the (concentrated) reduced model, we obtain the results of Fig. 8. Note that in this case, the reduced model yields more accurate spectra in the range $K \le k \le \text{ceil}(\sqrt{2}K)$. It is in fact more accurate here than the result of the exact eddy forcing (not shown) would be. As previously mentioned, the exact eddy forcing also yields too steep spectra in $K \le k \le \text{ceil}(\sqrt{2}K)$, due to the square truncation. For smaller k, where our eddy forcing parameterization now does not act, we do observe some discrepancy with respect to the reference spectra. However, remember from Section 4.1, that we increased the viscosity with respect to the value of the reference setup, i.e. we included an eddy viscosity term in the resolved equations. The spectra, when using the same

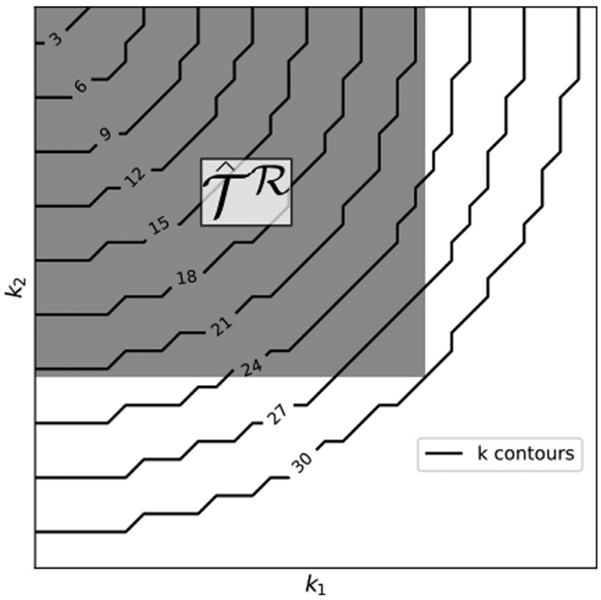

Fig. 5. The shaded area is the nonzero top left part of the spectral filter $\hat{\mathcal{T}}^{\mathcal{R}}$, see also Fig. 1. Superimposed on top are contours of k, which form concentric quarter circles. Note that $k = K = 21$ is the last quarter circle completely encompassed by the filter. For $k = \text{ceil}(\sqrt{2}K) = 30$, only a single wave number is captured by $\hat{\mathcal{T}}^{\mathcal{R}}$.

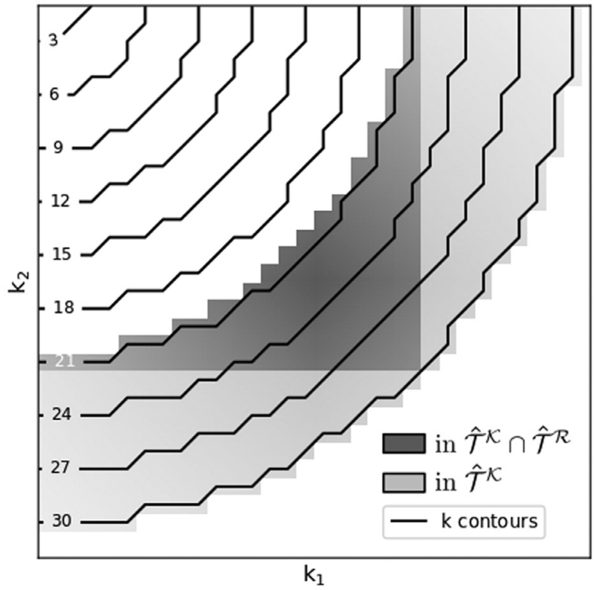
3
6
9
12
15
18
21
24
27
30
in $\hat{\mathcal{T}}^{\mathcal{K}} \cap \hat{\mathcal{T}}^{\mathcal{R}}$
in $\hat{\mathcal{T}}^{\mathcal{K}}$
k contours
k_2
k_1

Fig. 7. The light shaded area is the top left part of the spectral filter $\hat{\mathcal{T}}^{\mathcal{K}}(21, 30)$ with value 1. Superimposed on top are contours of k, which form concentric quarter circles. The dark shared area indiates the region shared by $\hat{\mathcal{T}}^{\mathcal{K}}$ and $\hat{\mathcal{T}}^{\mathcal{R}}$, which corresponds to the (k_1, k_2) vectors used by quantities such as $\mathcal{T}^{\mathcal{K}}\omega^{\mathcal{R}}$ or $\mathcal{T}^{\mathcal{K}}\psi^{\mathcal{R}}$. Quantities involving the reference, i.e. $\mathcal{T}^{\mathcal{K}}Q_i^{ref}$, correspond to (k_1, k_2) vectors that live in the light shaded area.

ν as the reference model (no eddy viscosity model included), are shown in Fig. 9. Here, we observe good overall agreement with the reference for both the energy and the enstrophy spectrum. Only near $k = K$ do we observe a small amount of spectral blocking, i.e. too much energy (and enstrophy) near the cutoff scale.

However, it should be noted that using such concentrated $\mathcal{T}^{\mathcal{K}}$ do not lead to a near-perfect pdf for the Q_i as before (not shown), due to the fact that we are only explicitly tracking $\mathcal{T}^{\mathcal{K}}Q_i^{ref}$, instead of the full Q_i^{ref}. The results are still improved over the eddy viscosity model though. If we wish to obtain both a pdf and a spectrum that matches the reference, we can employ two (partially overlapping) filters. For instance, we can use $V_1 = -\mathcal{T}^{\mathcal{K}}(0, K)\psi$, and $V_2 = -\mathcal{T}^{\mathcal{K}}(K-5, K)\psi$. The former ensures we track the overall energy, while the latter pushes the energy spectrum towards the reference just before the cutoff $k = K$. The pdf and spectrum of E obtained in the manner are shown in Fig. 10, both of which show a good match with the reference solution.

4.3. Tracking three QoI

Let us define $Z_2 := (\omega^2, \omega)/3$ as our third QoI, such that $Q := \{E, Z, Z_2\}$. In this case, we must set $V := \{-\Psi, \omega, \omega^2\}$ respectively. Again, we choose the $T_{i,j}$ basis functions also from this set, beginning with $T_{i,1} = V_i$. The remaining $T_{i,j}$ (2 in this case), are chosen such that all A_i matrices (see (21)) are symmetric. Moreover, they will be positive-semidefinite, see Section 4.4. By examining (21), note that if we set the remaining $T_{i,k}$ as $T_{i,k} = V_{j_{k-1}}$, we always get a symmetric matrix. Once the $T_{i,j}$ are set, there is nothing left to parameterise, and all analysis follows automatically. The orthogo-

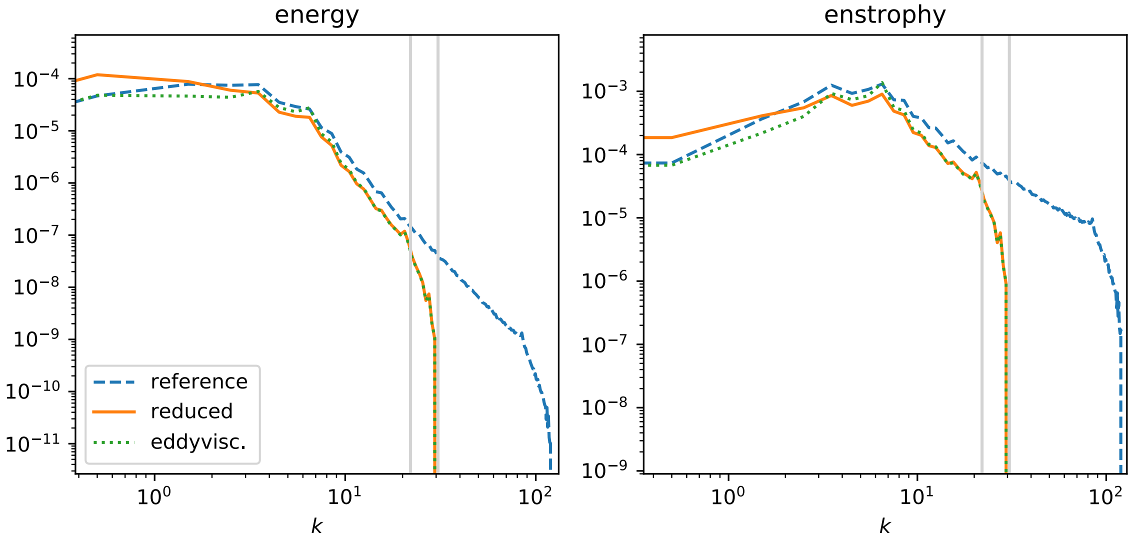

Fig. 6. Spectra of the energy and enstrophy for the reference (striped), model with eddy viscosity *and* reduced eddy forcing term (solid), and the baseline eddy viscosity model without reduced eddy forcing (dotted).

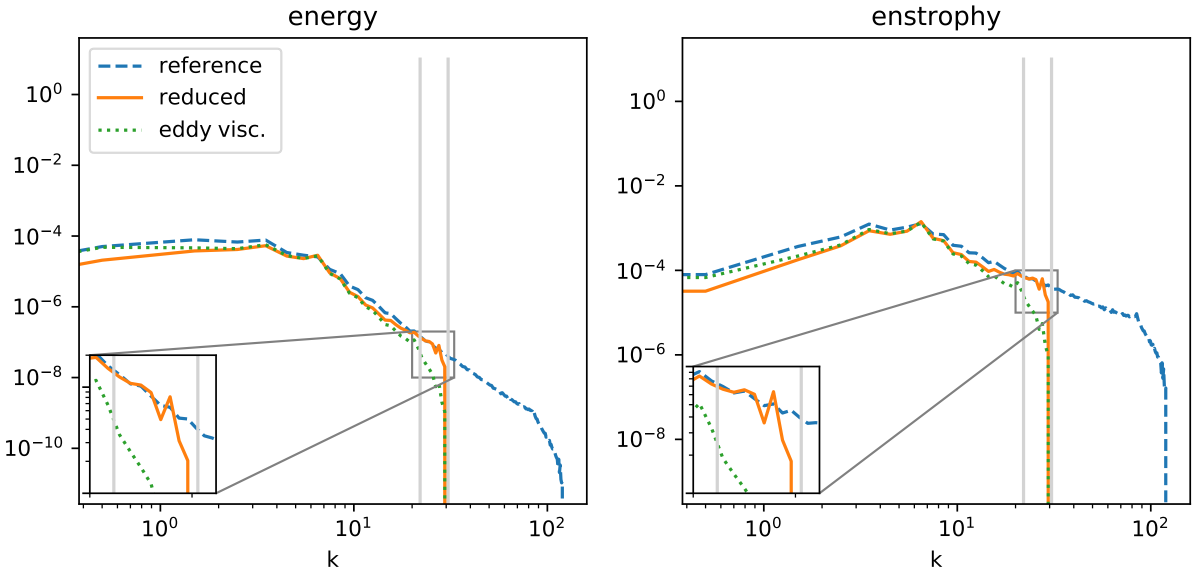

Fig. 8. Spectra of the energy and enstrophy for the reference (striped), model with eddy viscosity *and* reduced eddy forcing term with concentrated $\mathcal{T}^{\mathcal{K}}(21, 30)$ filter (solid), and the baseline eddy viscosity model without reduced eddy forcing (dotted). Note that the spectra of the reduced model are accurate in the range of $\mathcal{T}^{\mathcal{K}}$, i.e. $k \in [21, 30]$.

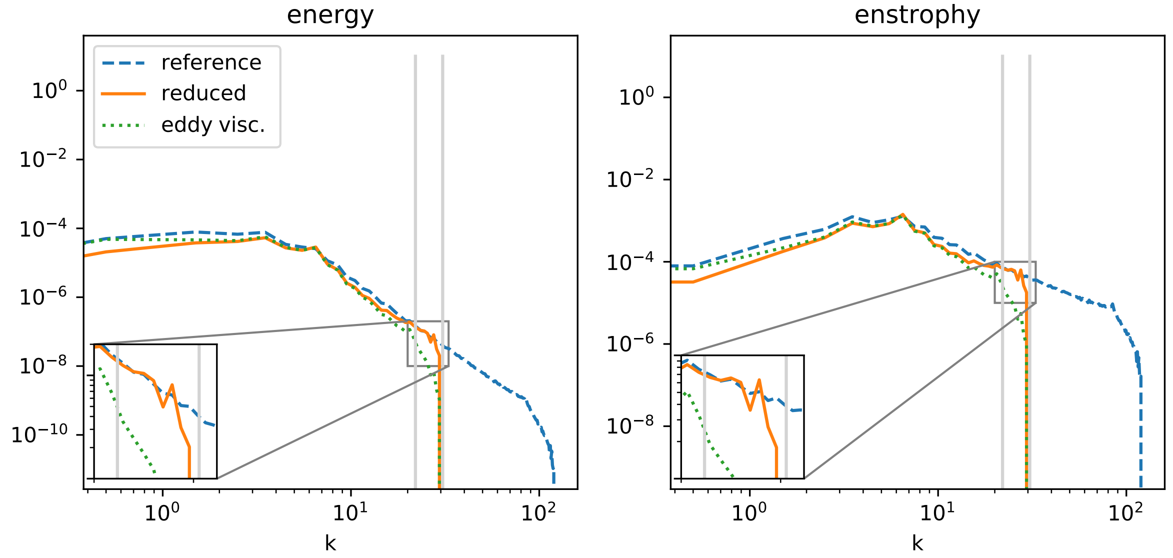

Fig. 9. Spectra of the energy and enstrophy for the reference (striped), model with reduced eddy forcing term with a concentrated $\mathcal{T}^{\mathcal{K}}(21, 30)$ filter (no eddy viscosity included) (solid), and the baseline eddy viscosity model (dotted). Note that the spectra of the reduced model are accurate in the range of $\mathcal{T}^{\mathcal{K}}$, i.e. $k \in [21, 30]$, but also for lower k, with the exception of some spectral blocking just before $k = K$.

nal patterns P_i (16) become

$$\begin{aligned} P_1 &= T_{1,1} - c_{1,2}T_{1,2} - c_{1,3}T_{1,3} = V_1 - c_{1,2}V_2 - c_{1,3}V_3 \\ &= -\Psi - c_{1,2}\omega - c_{1,3}\omega^2 \\ P_2 &= T_{2,1} - c_{2,2}T_{2,2} - c_{2,3}T_{2,3} = V_2 - c_{2,2}V_1 - c_{2,3}V_3 \\ &= \omega + c_{2,2}\Psi - c_{2,3}\omega^2 \\ P_3 &= T_{3,1} - c_{3,2}T_{3,2} - c_{3,3}T_{3,3} = V_3 - c_{3,2}V_1 - c_{3,3}V_2 \\ &= \omega^2 + c_{3,2}\Psi - c_{3,3}\omega. \end{aligned} \tag{32}$$

After which the orthogonality condition (18) leads to

$$\begin{bmatrix} (\omega, \omega) & \left(\omega, \omega^2\right) \\ \left(\omega, \omega^2\right) & \left(\omega^2, \omega^2\right) \end{bmatrix} \begin{bmatrix} c_{1,2} \\ c_{1,3} \end{bmatrix} = \begin{bmatrix} -(\omega, \Psi) \\ -\left(\omega^2, \Psi\right) \end{bmatrix},$$

$$\begin{bmatrix} (\Psi, \Psi) & -\left(\Psi, \omega^2\right) \\ -\left(\Psi, \omega^2\right) & \left(\omega^2, \omega^2\right) \end{bmatrix} \begin{bmatrix} c_{2,2} \\ c_{2,3} \end{bmatrix} = \begin{bmatrix} -(\Psi, \omega) \\ \left(\omega^2, \omega\right) \end{bmatrix},$$

$$\begin{bmatrix} (\Psi, \Psi) & -(\Psi, \omega) \\ -(\Psi, \omega) & (\omega, \omega) \end{bmatrix} \begin{bmatrix} c_{3,2} \\ c_{3,3} \end{bmatrix} = \begin{bmatrix} -\left(\Psi, \omega^2\right) \\ \left(\omega, \omega^2\right) \end{bmatrix}. \tag{33}$$

When we solve these three linear systems, we can compute the terms $-(\Psi, P_1)$, (ω, P_2) and (ω^2, P_3), and find the value of τ_i from the ΔQ_i data via (23). Written concisely, the reduced eddy forcing now reads

$$\underline{r} = \frac{\Delta E}{(-\Psi, P_1)}P_1 + \frac{\Delta Z}{(\omega, P_2)}P_2 + \frac{\Delta Z_2}{\left(\omega^2, P_3\right)}P_3, \tag{34}$$

where $P_i = P_i(x, y, t; c_{i,j})$.

We once more repeat the test case from Section 4.2, i.e. a 10 year simulation in which both the full reference model and the reduced model are run. The pdfs for all QoI are found in Fig. 11, which again show a near-perfect overlap. Note that in this case the eddy viscosity model does give an accurate prediction for the mean of Z_2, although the variance is underestimated.

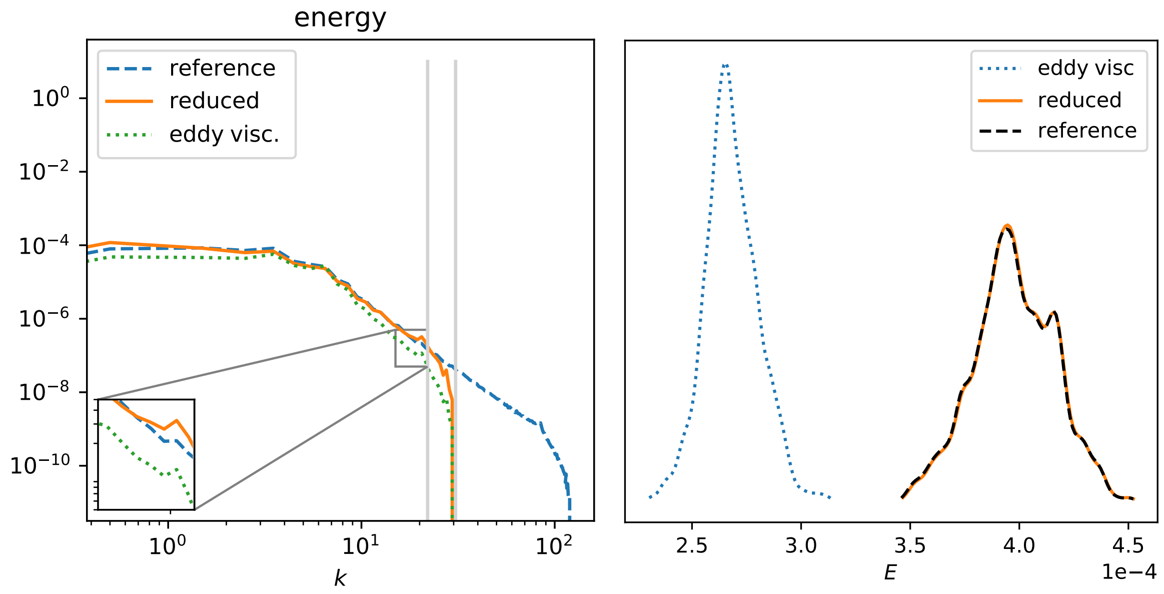

Fig. 10. (Left): Spectrum of the energy for the reference (striped), model with reduced eddy forcing term with a full $\mathcal{T}^{\mathcal{K}}(0,21)$ filter and a (partially overlapping) concentrated $\mathcal{T}^{\mathcal{K}}(16,21)$ filter (no eddy viscosity included) (solid), and the baseline eddy viscosity model (dotted). (Right): The corresponding pdfs of the energy.

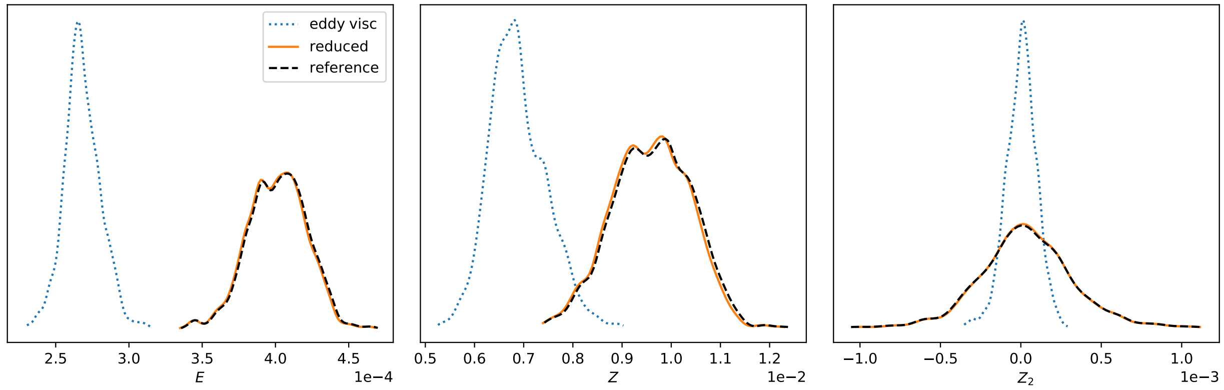

Fig. 11. The pdfs of the energy E (left), the enstrophy Z (middle), and Z_2 of both the reference (striped) and the low-resolution model (solid) with (30) as a model for the eddy forcing. The results of an eddy viscosity model are also included (dotted).

4.4. Quadrature and computational cost

The main cost of our proposed method is the computation of the various integrals that make up the linear system (21). As stated, we express the $T_{i,j}$ basis functions in terms of V_i, such that our integrals have the form (V_i, V_j) (see e.g. (33)). Then, given that: i) the order of the integrands is not important ($(\Psi, \omega) = (\omega, \Psi)$); and ii) repetition is allowed (e.g. (ω, ω)), we end up with combinations with repetition, such that the number of integrals n_{int} that needs to be calculated at every time step is given by

$$n_{int} = \binom{n+k-1}{k} = \binom{d+1}{2}. \qquad (35)$$

The last equality follows since we have $n = d$ basis functions to choose from and $k = 2$ places in every $(\ \cdot\ ,\ \ \cdot\ )$ operator. Hence for tracking 2 QoI we had to compute 3 integrals per time step, for $d = 3$ this was increased to 6, etc.

The results in this paper were generated by running the reference and the resolved model simultaneously (using the same time step), and manually calculating each integral using Simpson's rule of integration (see the software on [22]). Furthermore, for convenience, the same grid was used for both the resolved and the reference model, they differed only in the spectral space, see Fig. 1. While this serves as a proof-of-concept, it is not very efficient.

Let us therefore propose a more efficient numerical scheme, the source code of which can also be found at [22]. Besides executing the resolved model on its own, coarser, spatial grid, we also introduce a more efficient integration scheme. Using the Fourier expansions (see (3)) directly in the integral, we get

$$\begin{aligned}(\alpha(x,y,t), \beta(x,y,t)) &\approx \left(\tilde{\alpha}(x,y,t), \tilde{\beta}(x,y,t)\right) \\ &= \sum_{\mathbf{k}} \widehat{\alpha}_{\mathbf{k}}(t)\widehat{\beta}^*_{\mathbf{k}}(t) = \widehat{\alpha}^T\widehat{\beta}^* := \left\langle \widehat{\alpha}, \widehat{\beta}\right\rangle, \qquad (36)\end{aligned}$$

see Appendix B. Here, $\widehat{\alpha}$ is the vector of all Fourier coefficients $\widehat{\alpha}_{\mathbf{k}}$, and $(\ \cdot\ )^*$ denotes the complex conjugate. Now, let $\widehat{V} := \left[\widehat{V}_1, \ldots, \widehat{V}_d\right] \in \mathbb{R}^{N^2\times d}$ be the matrix of all Fourier coefficients[2] corresponding to $V = [V_1, \ldots, V_d]$. Then, the integrals of the d linear systems (21) (approximated using the dot products (36)), can all computed at once via a single dot product:

$$D := \widehat{V}^T\widehat{V} = \begin{bmatrix} \left\langle \widehat{V}_1, \widehat{V}_1\right\rangle & \ldots & \left\langle \widehat{V}_1, \widehat{V}_d\right\rangle \\ \vdots & \ddots & \vdots \\ \left\langle \widehat{V}_d, \widehat{V}_1\right\rangle & \ldots & \left\langle \widehat{V}_d, \widehat{V}_d\right\rangle \end{bmatrix} \qquad (37)$$

[2] We are using FFT here, not real FFT, thus the $\widehat{V}_i \in \mathbb{R}^{N^2}$.

The matrices A_i and their right-hand sides b_i from (21) are fully determined from the entries $D_{i,j}$ of matrix D, i.e.

$$A_i = \begin{bmatrix} D_{j_1,j_1} & \cdots & D_{j_1,j_{d-1}} \\ \vdots & \ddots & \vdots \\ D_{j_{d-1},j_1} & \ldots, & D_{j_{d-1},j_{d-1}} \end{bmatrix} \quad \text{and} \quad b_i = \begin{bmatrix} D_{i,j_1} \\ \vdots \\ D_{i,j_{d-1}} \end{bmatrix}$$
$$i \in \{1, \ldots, d\}, \quad j \in \{1, \ldots, d\}\backslash\{i\}. \tag{38}$$

Note that D and the A_i are Gramian matrices, and therefore are positive-semidefinite [23].

Let us close with a simple experiment to estimate to computational cost of our method. We first run both the resolved and the reference model in parallel for a simulated time of 50 days. This corresponds to a training run with a wall-clock time of T_{train}. Next, we run only the resolved model, using stored data from the training run to inform the τ_i. The wall clock time T_{surr} obtained in this manner would correspond to a run with a surrogate for the τ_i, assuming that the surrogate itself introduces a negligible overhead compared to the cost of running the resolved model. We obtained a ratio $T_{surr}/T_{train} \approx 0.19$, roughly a speed up of a factor of 5. Note that we kept the time step the same for both models, which we could have reduced for the resolved model in order to get a larger speed up. Furthermore, note that these results are also dependent upon the difference in the grid size of the reference and resolved model. If we increase the reference grid from 256 × 256 to 512 × 512 or higher, this obviously impacts the speed up factor.

5. Conclusion

We have proposed a data-preprocessing step for the closure problem in dynamic turbulent flows, in which the exact subgrid scale term is replaced by an alternative subgrid scale model which retains accuracy for a set of hand-selected, time-dependent quantities of interest. The size of the unclosed component of the new subgrid scale term is reduced from a full-field term, down to a component with the dimension equal to the number of quantities we aim to approximate in the first place. The idea is that this will facilitate easier, more transparent surrogate construction.

We demonstrated our approach on the forced-dissipative, two-dimensional vorticity equations, for a number of different quantities of interest, to showcase the generality of our approach. Our alternative subgrid scale models can also be applied in a scale-selective manner, where for instance only the finer scales near the cutoff scale are modified. In ongoing work, we are constructing a stochastic surrogate model using various neural nets, trained on reduced data. Somewhat similar to our approach, the recent work of [24] employed a model with fixed spatial modes and time-varying coefficients applied to a turbulent shear flow problem. They compared neural nets with time-lagged features with long short-term memory (LSTM) networks, regarding their ability to predict the temporal evolution the coefficients. The LSTM networks displayed promising results for capturing the dynamical behavior of such chaotic systems.

Another potential research direction is extending the approach to three-dimensional problems. We do not expect large difficulties here. The main difference would be that the quantities of interest must be integrated over three spatial dimensions instead of two. We derived our framework for tracking spatially integrated QoIs with integrands that can be expressed as some function of the primitive variables (the stream function and vorticity in our case). In three dimensions, the same QoI type would be used. The accuracy depends upon the ability of our method to track these QoIs, which depends in turn upon the chosen model for the unclosed τ_i. In this paper we modelled them using linear relaxation towards the reference, which worked well for all QoIs considered. If in future applications accuracy problems arise, it is likely that the linear relaxation model is insufficient, or that QoI-specific relaxation time scales must be included.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Wouter Edeling: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Daan Crommelin:** Conceptualization, Writing - review & editing, Resources, Supervision, Project administration, Funding acquisition.

Acknowledgements

This research is funded by the Netherlands Organization for Scientific Research (NWO) through the Vidi project "Stochastic models for unresolved scales in geophysical flows", and from the European Union Horizon 2020 research and innovation programme under grant agreement #800925 (VECMA project).

Appendix A. Energy in terms of ψ and ω

The energy (density) is defined as

$$E^{\mathcal{R}} := \frac{1}{2}\left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \mathbf{V}^{\mathcal{R}} \cdot \mathbf{V}^{\mathcal{R}} \mathrm{d}x\mathrm{d}y, \tag{A.1}$$

where $\mathbf{V}^{\mathcal{R}}$ is the vector containing the velocity components in x and y direction. It can be rewritten as $E^{\mathcal{R}} = -\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$ via

$$\begin{aligned} \mathbf{V}^{\mathcal{R}} \cdot \mathbf{V}^{\mathcal{R}} &= \nabla\psi^{\mathcal{R}} \cdot \nabla\psi^{\mathcal{R}} = \nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right) - \psi^{\mathcal{R}}\nabla^2\psi^{\mathcal{R}} \\ &= \nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right) - \psi^{\mathcal{R}}\omega^{\mathcal{R}} \end{aligned} \tag{A.2}$$

The first equality follows from the definition $\mathbf{V}^{\mathcal{R}} := \left(-\partial\psi^{\mathcal{R}}/\partial y, \partial\psi^{\mathcal{R}}/\partial x\right)^T$, while the second stems from the product rule of a scalar $(\psi^{\mathcal{R}})$ and a vector $(\nabla\psi^{\mathcal{R}})$:

$$\nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right) = \nabla\psi^{\mathcal{R}} \cdot \nabla\psi^{\mathcal{R}} + \psi^{\mathcal{R}}\nabla^2\psi^{\mathcal{R}}. \tag{A.3}$$

Finally, the last equality of (A.2) simply follows from the governing Eq. (1). The term $\nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right)$ disappears when integrated over the spatial domain, after application of the divergence theorem in combination with the doubly periodic boundary conditions. This leaves $E^{\mathcal{R}} = -\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$ [3].

Appendix B. Quadrature using the Fourier expansion

Consider two real, spatially and time-dependent quantities $\alpha(x, y, t)$ and $\beta(x, y, t)$, for which we aim to approximate

$$(\alpha, \beta) = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \alpha\beta \,\mathrm{d}x\mathrm{d}y. \tag{B.1}$$

We then insert the Fourier expansions for α and β

$$\begin{aligned} (\alpha, \beta) \approx \left(\tilde{\alpha}, \tilde{\beta}\right) &= \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \sum_{\mathbf{k}} \hat{\alpha}_{\mathbf{k}} e^{i(k_1x+k_2y)} \\ &\quad \times \sum_{\mathbf{q}} \hat{\beta}_{\mathbf{q}} e^{i(q_1x+q_2y)} \,\mathrm{d}x\mathrm{d}y \\ &= \sum_{\mathbf{k}}\sum_{\mathbf{q}} \hat{\alpha}_{\mathbf{k}}\hat{\beta}_{\mathbf{q}} \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi} \\ &\quad \times \int_0^{2\pi} e^{i(k_1x+k_2y)} e^{i(q_1x+q_2y)} \,\mathrm{d}x\mathrm{d}y. \end{aligned} \tag{B.2}$$

Due to the orthogonality of the Fourier modes:

$$\int_0^{2\pi}\int_0^{2\pi} e^{i(k_1x+k_2y)}e^{i(q_1x+q_2y)}\,\mathrm{d}x\mathrm{d}y = \begin{cases}(2\pi)^2 & \mathbf{k}=-\mathbf{q}\\ 0 & \text{otherwise}\end{cases}. \tag{B.3}$$

Thus, (B.2) becomes

$$(\alpha,\beta)\approx\sum_{\mathbf{k}}\hat{\alpha}_{\mathbf{k}}\hat{\beta}_{-\mathbf{k}}. \tag{B.4}$$

Since α and β are real, $\hat{\beta}_{-\mathbf{k}}=\hat{\beta}^*_{\mathbf{k}}$ must hold, where $(\ \cdot\ )^*$ denotes the complex conjugate [17]. Hence we finally obtain,

$$(\alpha,\beta)\approx\sum_{\mathbf{k}}\hat{\alpha}_{\mathbf{k}}\hat{\beta}^*_{\mathbf{k}}. \tag{B.5}$$

Discrete Fourier transforms of α and β are scaled by a factor N^{-2} for two-dimensional quantities. Note that depending upon the specific implementation, these scaling factors may appear in either the FFT or the inverse FFT, or sometimes a scaling factor is applied to both. In our case, we used the Numpy FFT implementation to compute the Fourier coefficients, and we had to scale the right-hand side of (B.5) by N^{-4}.

References

[1] Gent P, Mcwilliams J. Isopycnal mixing in ocean circulation models. J Phys Oceanogr 1990;20(1):150–5.

[2] McWilliams J. The emergence of isolated coherent vortices in turbulent flow. J Fluid Mech 1984;146:21–43.

[3] Verkley W, Kalverla P, Severijns C. A maximum entropy approach to the parametrization of subgrid processes in two-dimensional flow. Q J R Meteorolog Soc 2016;142(699):2273–83.

[4] Thuburn J, Kent J, Wood N. Cascades, backscatter and conservation in numerical models of two-dimensional turbulence. Q J R Meteorolog Soc 2014;679(140):626–38.

[5] Frank J, Leimkuhler B, Myerscough K. Direct control of the small-scale energy balance in two-dimensional fluid dynamics. J Fluid Mech 2015;782:240–59.

[6] Berloff P. Random-forcing model of the mesoscale oceanic eddies. J Fluid Mech 2005;529:71–95.

[7] Verheul N, Viebahn J, Crommelin D. Covariate-based stochastic parameterization of baroclinic ocean eddies. Math Climate Weather Forecast 2017;3(1):90–117.

[8] Mana P, Zanna L. Toward a stochastic parameterization of ocean mesoscale eddies. Ocean Modell 2014;79:1–20.

[9] Zanna L, Mana P, Anstey J, David T, Bolton T. Scale-aware deterministic and stochastic parametrizations of eddy-mean flow interaction. Ocean Modell 2017;111:66–80.

[10] Grooms I, Zanna L. A note on 'toward a stochastic parameterization of ocean mesoscale eddies'. Ocean Modell 2017;113:30–3.

[11] Maulik R, San O, Rasheed A, Vedula P. Subgrid modelling for two-dimensional turbulence using neural networks. J Fluid Mech 2019;858:122–44.

[12] Scher S. Toward data-driven weather and climate forecasting: approximating a simple general circulation model with deep learning. Geophys Res Lett 2018;45(22):12–616.

[13] Scher S, Messori G. Weather and climate forecasting with neural networks: using general circulation models (gcms) with different complexity as a study ground. Geosci Model Dev 2019;12(7):2797–809.

[14] Bolton T, Zanna L. Applications of deep learning to ocean data inference and subgrid parameterization. J Adv Model Earth Syst 2019;11(1):376–99.

[15] Edeling W, Crommelin D. Reduced model-error source terms for fluid flow. In: Proceedings of the uncertainty quantification in computational sciences and engineering conference 2019. ECCOMAS; 2019.

[16] Edeling W.. vorticity-solver (github repository). https://github.com/wedeling/vorticity-solver; 2019a.

[17] Peyret R. Spectral methods for incompressible viscous flow, 148. Springer Science & Business Media; 2013.

[18] Canuto C, Hussaini M, Quarteroni A, Zang T. Spectral methods: evolution to complex geometries and applications to fluid dynamics. Springer Science & Business Media; 2007.

[19] Hou T, Li R. Computing nearly singular solutions using pseudo-spectral methods. J Comput Phys 2007;226(1):379–97.

[20] Margairaz F, Giometto M, Parlange M, Calaf M. Comparison of dealiasing schemes in large-eddy simulation of neutrally stratified atmospheric flows. Geosci Model Dev 2018;11(10):4069–84.

[21] Edeling W, Crommelin D. Towards data-driven dynamic surrogate models for ocean flow. In: Proceedings of the platform for advanced scientific computing conference 2019. ACM; 2019. p. 3.

[22] Edeling W.. Caf reduced sgs (github repository). https://github.com/wedeling/CAF_reduced_sgs; 2019b.

[23] Basilevsky A. Applied matrix algebra in the statistical sciences. Courier Corporation; 2013.

[24] Srinivasan P, Guastoni L, Azizpour H, Schlatter P, Vinuesa R. Predictions of turbulent shear flows using deep neural networks. Phys Rev Fluids 2019;4(5):54603.