JM PAPERS

Grid-based calculation of the Lagrangian mean

Hossein A. Kafiabad†

School of Mathematics and Maxwell Institute for Mathematical Sciences, University of Edinburgh, Edinburgh EH9 3FD, UK

Lagrangian averaging has been shown to be more effective than the Eulerian mean in separating waves from slow dynamics in two time scale flows. It also appears in many reduced models that capture the wave feedback on the slow flow. Its calculation, however, requires tracking particles in time, which imposes several difficulties in grid-based numerical simulations or estimation from fixed-point measurements. To circumvent these difficulties, we propose a grid-based iterative method to calculate the Lagrangian mean without tracking particles in time, which also reduces computation, memory footprint and communication between processors in parallelised numerical models. To assess the accuracy of this method several examples are examined and discussed. We also explore an application of this method in the context of shallow-water equations by quantifying the validity of wave-averaged geostrophic balance – a modified form of geostrophic balance accounting for the effect of strong waves on slow dynamics.

Key words: waves in rotating fluids, computational methods, ocean circulation

1. Introduction

Large-scale geophysical flows are characterised by fast and slow time scales; the fast dynamics consists of inertia-gravity waves, and the slow dynamics is usually described by geostrophic and hydrostatic balance (Vanneste 2013; Vallis 2017). There is a particular interest in decomposing the flow into these components and analysing them separately. Focusing on the slow dynamics reduces the governing equations to a simpler set that is easier to interpret and solve (Charney 1971; Warn 1997; McIntyre & Norton 2000; Mohebalhojeh & Dritschel 2001). Likewise, mathematical tools specific to wave dynamics may be employed to analyse the fast component (Bühler 2009). In addition to bringing physical insights, such decompositions are used to parameterise the small-scale dynamics that are not resolved in ocean and weather models (Sutherland *et al.* 2019).

The most straightforward decomposition is averaging the variables over the fast time scale at fixed spatial points to derive the Eulerian mean. Although easy to implement,

† Email address for correspondence: h.kafiabad@ed.ac.uk

940 A21-1

this is not the most effective way of extracting wave signal from the mean flow. For instance, the Doppler shifting of the wave frequency by strong mean flows can eclipse the inherent time scale separation between them. To circumvent this issue, one can derive the Lagrangian mean (LM) instead by averaging the flow variables along particle trajectories and assigning the mean to a fixed particle label. Recent studies have shown a range of oceanic phenomena are better understood with Lagrangian averaging, including spontaneous wave generation (Nagai *et al.* 2015; Shakespeare & Hogg 2017, 2018), internal tides (Shakespeare & Hogg 2019) and their effect on the ecology of the coastal and reef zones (Bachman *et al.* 2020).

Another motivation for calculating the LM comes from its natural appearance in reduced models that are derived by wave averaging (Bretherton 1971; Grimshaw 1975; Andrews & McIntyre 1978; Bühler 2009; Gilbert & Vanneste 2018). A striking prediction, highlighting the fundamental role of the LM, is that the hydrostatic and geostrophic balance continues to hold in the presence of strong waves for LM quantities (Kafiabad, Vanneste & Young (2021), and references therein). Similarly, models of near-inertial waves interacting with the mean flow naturally involve LM variables (Xie & Vanneste 2015; Wagner & Young 2015, 2016; Asselin & Young 2020). In general, these studies show that the slow dynamics with the inclusion of wave feedback on the flow is more accurately represented by LM quantities. To examine the validity of these models and utilise them in operational and research models there is a growing need to compute the LM.

Computing the LM, however, requires more effort than computing the Eulerian mean. Most numerical models are discretised using spatial grid points and the majority of measurements are taken at fixed locations in space. To derive the LM, in these cases passive particles are seeded in the flow and advected (forward or backward in time) based on the interpolated velocities at particle positions (e.g. Nagai *et al.* 2015; Shakespeare & Hogg 2017, 2018, 2019). This requires saving the time series of several fields, which are not normally saved in numerical models, at high spatial resolution. The other issue with particle tracking comes up in parallelised solvers, where the computational domain is divided into many subdomains assigned to independent processors. Even uniformly seeded particles may leave their initial subdomain, causing additional communication between the processors and disturbing the computational load balance among them.

We propose a grid-based Lagrangian averaging (GBLA) method that does not require particle tracking. Instead, the LM is updated on grid points using the mean associated with the trajectories that pass through the grid points in the previous time step. As a result, there is no need for saving any additional time series. This method can be integrated with numerical models to efficiently calculate the LM 'on the fly'. It also maintains a load balance between the processors in parallel implementations and minimises the communication between them. We first describe this method and its mathematical justification in § 2. We then investigate its validity in five examples in § 3. An application of GBLA is considered in § 4, where we study the wave-averaged geostrophy for the rotating shallow-water equations. The paper concludes with a discussion in § 5.

2. GBLA

2.1. *Algorithm*

Let us consider the LM of the field $\phi(\boldsymbol{x}, t)$ from t_0 to $t_0 + T$. Before explaining the steps of GBLA, we clarify some notations and definitions. We denote the Eulerian mean of $\phi(\boldsymbol{x}, t)$

over the period of t_0 to t with

$$\bar{\phi}(x, t_0, t) = \frac{1}{t - t_0} \int_{t_0}^{t} \phi(x, \tau)\, d\tau. \tag{2.1}$$

The LM of ϕ is the time average of this quantity along the trajectory, that is, at fixed particle label instead of fixed spatial position. There are different ways of labelling particles; for instance, the particle position at the beginning or the end of the averaging interval may be selected as labels. Denoting this label by x, we can map it to the particle position at time t (denoted by X) using the particle displacement field $\xi(x, t)$,

$$X = \Xi(x, t) = x + \xi(x, t). \tag{2.2}$$

Using (2.1) and (2.2), the LM operator is mathematically described as

$$\overline{\phi}^{L}(x, t_0, t) = \frac{1}{t - t_0} \int_{t_0}^{t} \phi(x + \xi(x, \tau), \tau)\, d\tau. \tag{2.3}$$

Different choices of x lead to different displacement fields ξ. Andrews & McIntyre (1978) defined the 'generalised Lagrangian mean (GLM)' by requiring

$$\bar{\xi}(x, t) = 0. \tag{2.4}$$

In other words, x is now the mean position of the particle from t_0 to t. If the displacement field is small (compared with the changes in mean position), (2.4) gives the LM some Eulerian-like character as x is more than just a label and represents the spatial location of the trajectory for which the mean is calculated. As it will be shown in the examples of § 3.1, this choice of label is more effective at removing the fast time scale in the LM estimate provided that the mapping (2.2) does not have a fast time dependence. If other labels such as instantaneous position at the end or middle of the averaging interval are chosen, the filtered signal may still contain some fast oscillations or get distorted (see the numerical example of § 3.1). Moreover, the GLM definition holds properties that lead to significant analytical simplifications in modelling two time scale geophysical flows (Andrews & McIntyre 1978; Bühler & McIntyre 1998; Xie & Vanneste 2015; Gilbert & Vanneste 2018). Throughout this paper, we use (2.3) in conjunction with (2.4) to express the LM in terms of the mean position. For $t < (t_0 + T)$, (2.1) and (2.3) represent partial means, and for $t = t_0 + T$ the complete mean for the intended averaging interval. Considering that in this section we focus only on one averaging interval with a fixed starting point, we drop t_0 in the arguments of the mean operators to lighten the notation.

The core idea behind GBLA is to calculate partial LMs recursively in time. For instance, we assume that the partial LM from t_0 to t_k ($t_0 < t_k < t_0 + T$) for the particles that reach the grid points at time t_k is already computed. Through the steps laid out below, the LM is extended to t_{k+1} and calculated for a new set of particles that reach the grid points at time t_{k+1} (using the partial averages of the previous step). In this approach, at each time step a different set of trajectories is considered (the set that ends up at grid points at that time step), which is in contrast with particle tracking methods where one trajectory is followed all along over the entire averaging interval. The incremental extension of partial LMs continues to the end of the averaging interval $t_0 + T$. The completed LM field is then assigned to time $t_0 + T/2$. Note that in our demonstration we focus only on one averaging interval, but this process can be repeated for new intervals, and averaging procedure can be carried out simultaneously. For example, the next interval may start from t_m and end at t_{m+N} and so on. If the beginning of a later interval exceeds $t_0 + T$, the memory allocated

to the first interval (from t_0 to $t_0 + T$) can be freed in a systematic way and used for the new intervals.

Equipped with the above definitions, we present the steps to compute $\overline{\phi}^L(\mathbf{\Xi}^{-1}(X_i, t_{k+1}), t_{k+1})$ from $\overline{\phi}^L(\mathbf{\Xi}^{-1}(\{X_\alpha\}, t_k), t_k)$, where X_i is an arbitrary grid point in space and $\{X_\alpha\}$ is a set consisting of X_i and its neighbours. In other words, our steps show how to calculate the partial LM (from t_0 to t_{k+1}) for the particle that passes through X_i at t_{k+1} by knowing the LM (from t_0 to t_k) for the particles that pass through X_i and its neighbouring grid points at t_k. To simplify the notation we use one spatial index i, which can be viewed as a multielement vector or replaced by several indices in higher than one dimension. Considering that $\overline{\phi}^L(\mathbf{\Xi}^{-1}(X_i, t_k), t_k)$ is the partial LM for the particle that reaches X_i at t_k and $\overline{\phi}^L(\mathbf{\Xi}^{-1}(X_i, t_{k+1}), t_{k+1})$ is the partial LM for the particle that reaches X_i at t_{k+1}, these two LMs belong to two separate trajectories (marked in red and blue in figure 1).

Step 1 Calculate $\phi(X_i, t_{k+1})$ and the velocity $u(X_i, t_{k+1})$ by integrating the governing equations from t_k to t_{k+1}. Note that a finer time step may be used for the time integration of the governing equations than that used for averaging; there might be several simulation time steps between t_k and t_{k+1}.

Step 2 Consider the particle that goes through X_i at time t_{k+1} (red trajectory in figure 1). Approximate the position of this particle at time t_k by advecting it backward in time,

$$Y_i = X_i - (t_{k+1} - t_k)u(X_i, t_{k+1}). \tag{2.5}$$

This point is marked in green in figure 1 and splits the red trajectory into two parts.

Step 3 Denote the partial LM of ϕ between t_0 and t_k for the red trajectory by $\overline{\phi}^L(\mathbf{\Xi}^{-1}(Y_i, t_k), t_k)$. Then, find its value by interpolation using the LM for the trajectories that go through the neighbouring grid points (marked by blue curves). For example, Y_i lies between $\{X_\alpha\}$ for $\alpha = i, j, l$ and p in figure 1. Therefore, the four values of $\overline{\phi}^L(\mathbf{\Xi}^{-1}(\{X_\alpha\}, t_k), t_k)$ may be used to interpolate $\overline{\phi}^L(\mathbf{\Xi}^{-1}(Y_i, t_k), t_k)$. Note that $\overline{\phi}^L(\mathbf{\Xi}^{-1}(Y_i, t_k), t_k)$ and $\overline{\phi}^L(\mathbf{\Xi}^{-1}(X_i, t_k), t_k)$ are both averages over $[t_0, t_k]$ but for different trajectories (highlighted in orange and light blue, respectively).

Step 4 Selecting the mean position of the red trajectory from t_0 to t_{k+1} as a label for the particle that passes through X_i at t_{k+1} and denoting it by x_i, use the definition (2.3) to compute

$$\begin{aligned} \overline{\phi}^L(\mathbf{\Xi}^{-1}(X_i, t_{k+1}), t_{k+1}) &= \frac{1}{t_{k+1} - t_0} \int_{t_0}^{t_{k+1}} \phi(x_i + \xi(x_i, \eta), \eta) \, d\eta \\ &= \frac{t_k - t_0}{t_{k+1} - t_0} \frac{1}{t_k - t_0} \int_{t_0}^{t_k} \phi(x_i + \xi(x_i, \eta), \eta) \, d\eta + \frac{1}{t_{k+1} - t_0} \int_{t_k}^{t_{k+1}} \phi(x_i + \xi(x_i, \eta), \eta) \, d\eta \\ &= \frac{t_k - t_0}{t_{k+1} - t_0} \overline{\phi}^L(\mathbf{\Xi}^{-1}(Y_i, t_k), t_k) + \frac{1}{t_{k+1} - t_0} \int_{t_k}^{t_{k+1}} \phi(x_i + \xi(x_i, \eta), \eta) \, d\eta. \end{aligned} \tag{2.6}$$

The first term on the right-hand side is already computed in the previous step. The second term, which is the weighted mean of the red trajectory from t_k to t_{k+1}, can be approximated by

$$\int_{t_k}^{t_{k+1}} \phi(x_i + \xi(x_i, \eta), \eta) \, d\eta \approx \phi(X_i, t_{k+1}) \, (t_{k+1} - t_k). \tag{2.7}$$
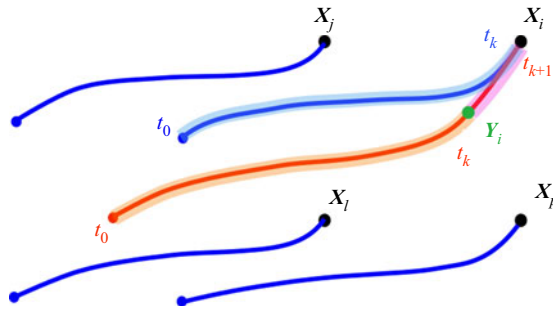
Figure 1. Schematic view of particle trajectories that pass through the grid points X_i, X_j, X_p and X_l that are marked by black. The trajectories that reach these grid points at t_k are marked in blue and the trajectory that reaches the grid point at t_{k+1} in red. For X_i, the times associated with the particle position of red and blue trajectories are shown as well.

A better approximation may be derived by the trapezoidal rule

$$\int_{t_k}^{t_{k+1}} \phi(x_i + \boldsymbol{\xi}(x_i, \eta), \eta) \, d\eta \approx \tfrac{1}{2} \left[\phi(Y_i, t_k) + \phi(X_i, t_{k+1}) \right] (t_{k+1} - t_k). \quad (2.8)$$

To calculate $\phi(Y_i, t_k)$, another interpolation is required. Hence, depending on the desired accuracy and computational cost, (2.8) may or may not be preferred to (2.7).

So far the calculation of the LM field in (2.6) is complete, but it is given for the end position of each particle X_i. If the definition of GLM with the condition (2.4) is desired, the LM should be assigned to the mean position of particles. In other words, we need to calculate $x_i = \boldsymbol{\Xi}^{-1}(X_i, t_{k+1})$, which maps the particle's end position X_i to its mean from t_0 to t_{k+1}. To calculate the mean position, we update it in a similar manner to (2.6) at each time step via step 5.

Step 5 Assuming $\boldsymbol{\Xi}^{-1}(X, t_k)$ is known for all grid points from the previous time step, compute the mean position of the red trajectory from t_0 to t_{k+1},

$$\boldsymbol{\Xi}^{-1}(X_i, t_{k+1}) = \frac{1}{t_{k+1} - t_0} \int_{t_0}^{t_{k+1}} x_i + \boldsymbol{\xi}(x_i, \eta) \, d\eta$$

$$= \frac{t_k - t_0}{t_{k+1} - t_0} \boldsymbol{\Xi}^{-1}(Y_i, t_k) + \frac{1}{t_{k+1} - t_0} \frac{1}{2} (Y_i + X_i)(t_{k+1} - t_k), \quad (2.9)$$

where $\boldsymbol{\Xi}^{-1}(Y_i, t_k)$ is interpolated using the neighbouring grid points. Note that (2.9) is simply a particular case of (2.6) in conjunction with (2.8) for $\phi(x, t) = x$.

The above steps are carried out for all the grid points and iteratively in time until $t_{k+1} = t_0 + T$. The completed LM is now given on scattered mean positions $\boldsymbol{\Xi}^{-1}(X_i, t_N)$ and can be interpolated back on the computational grid points.

2.2. *Parameter set-up, convergence and numerical error*

The GBLA method may be implemented on temporally and spatially decimated fields. To distinguish between the two, we denote the (iterative) averaging time step with $\Delta t = t_{k+1} - t_k$ and the simulation time step (used to solve the governing equations) with δt

throughout this paper. Similarly, Δx denotes the spatial resolution of GBLA, which could be coarser than the one used in simulations (i.e. weather or climate models). It is important to keep Δt and Δx smaller than the characteristic time and length scales of the fast phenomenon that is going to be filtered. To guarantee the stability of backward advection in (2.5) the Courant–Friedrichs–Lewy condition $u\Delta t < \Delta x$ should be satisfied.

Just like any numerical algorithm, GBLA is susceptible to numerical errors. Denoting the local error at each iteration (from t_k to t_{k+1}) by e_k, there are three main contributors to e_k.

   (i) Finite differentiation used to advect particles backward in (2.5). Considering the Taylor expansion of particle position on a trajectory, the difference between the exact and approximated particle position at time t_k is $O(\Delta t^2)$.
  (ii) Interpolation of LM at time t_k. In the step 3 of the algorithm, we interpolate $\overline{\phi}^L(\boldsymbol{\Xi}^{-1}(\boldsymbol{Y}_i, t_k), t_k)$, leading to an error that depends on the selected interpolation scheme. Considering polynomial interpolation of degree n on a grid with spacing of Δx, one can show that the interpolation error is $O((\Delta x)^{n+1})$. For example, the linear interpolation error is $O((\Delta x)^2)$ and the cubic interpolation error $O((\Delta x)^3)$. Note that the order of error is independent of spatial dimension.
 (iii) Approximation of the last integral in (2.6) either by (2.7) or (2.8). The error of the former is $O(\Delta t)$ and the latter $O(\Delta t^3(\Delta x)^{n+1})$ considering the interpolation required to calculate $\phi(\boldsymbol{Y}_i, t_k)$.

Considering that there are linear leading terms in polynomial interpolation, the error in (i) directly translates into the total local error e_k. As a result, we can write

$$e_k = O(\Delta t^2) + O((\Delta x)^{n+1}) + O(\Delta t^s), \tag{2.10}$$

where $s = 1$ or 3 if (2.7) or (2.8) are used, respectively. Hence, depending on the value of s, the order of local error in Δt may be determined by (i) or (ii). The calculation of global error, which is the accumulation of local errors, is more complicated. It cannot be derived simply by multiplying e_k with the number of time steps $N = T/\Delta t$, because in the interpolation of the step 3 the neighbouring points contain the accumulated interpolation errors from the previous iterations. As a result, different sources of error become intertwined. Comparatively, the particle tracking approach does not have this type of error accumulation, because it uses the instantaneous values at neighbouring points for each time step. Hence, GBLA might be more affected by interpolation errors than the particle tracking methods. We leave the analytical expression of global error for a future study, but in § 3.1 and § 3.2.1 we calculate it numerically to investigate the convergence of GBLA.

3. Validation

3.1. *Illustrative one-dimensional flows*

We consider the following example with a velocity that consists of a fast and slow component:

$$u = u_f + u_s, \quad u_f = \epsilon \sin(20x - 20t), \quad u_s = e^{-a(x-\pi)^2}, \tag{3.1a–c}$$

where $a = 15$ for $x < \pi$ and $a = 1$ for $x \geq \pi$. Figure 2(a) displays the LM velocity of this flow regarded as a function of mean position using GBLA with cubic interpolation (shown in magenta) and particle tracking (shown in black). For both methods, $\Delta x = 4.2 \times 10^{-3}$,
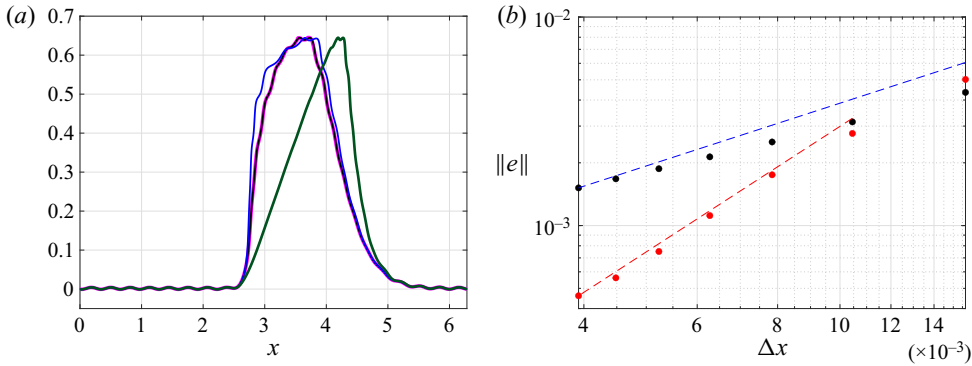
Figure 2. (*a*) The LM velocity of (3.1*a–c*) calculated for $\epsilon = 0.06$ from $t = 0$ to $t = 2\pi/3$ for $\Delta x = 4.2 \times 10^{-3}$ and $\Delta t = 2.6 \times 10^{-3}$ using: GBLA (with cubic interpolation) as a function of mean position (magenta); particle tracking as a function of mean position (black); particle tracking as a function of position at the middle of the averaging interval (blue); and particle tracking as a function of position at the end of the averaging interval (green). (*b*) The error norm $\|e\|$ of the GBLA scheme using linear interpolation (black dots) and cubic interpolation (red dots). The slope of blue guide line is 1 and that of red line is 2.

$\Delta t = 2.6 \times 10^{-3}$ and the averaging period $T = 2\pi/3$, which is longer than the period of the fast signal $2\pi/20$. The GBLA and particle tracking results (magenta and black curves) perfectly lie on top of each other, which confirms the validity of GBLA. To better quantify the accuracy and convergence of GBLA, in figure 2(*b*) we coarsen Δx and calculate the following error norm (representing the global error for the averaging period of T):

$$\|e\| = \sqrt{\frac{1}{N} \sum (\bar{u}_{GBLA}^L - \bar{u}_{exact}^L)^2}, \tag{3.2}$$

where the summation is carried out over all spatial grid points, and N is the number of spatial grid points. Here $\bar{u}_{GBLA}^L$ is the LM calculated using GBLA and $\bar{u}_{exact}^L$ is the exact value of LM which is approximated by particle tracking at much finer spatial and temporal resolutions of $\Delta x = 2 \times 10^{-3}$ and $\Delta t = 2.6 \times 10^{-3}$. Two different interpolation schemes are used in computing GBLA for figure 2(*b*): linear (black dots) and cubic (red dots). The overall message of this figure is that GBLA with different interpolation schemes converges to the exact solution if Δx is lowered. At smaller Δx, GBLA's error with cubic interpolation has a slope close to 2, and with linear interpolation a slope slightly shallower than 1. At larger Δx other sources of error contribute to $\|e\|$ leading to shallower slopes.

As mentioned in the previous section, we choose the definition of GLM theory by requiring (2.4) to hold and using mean position as a label for LM. We use the above example to clarify the importance of this choice. In figure 2(*a*), in addition to expressing the LM as a function of mean position, we show the LM as a function of particle position at the middle of the averaging interval (blue curve) and as a function of particle position at the end of the averaging interval (green curve). We have used particle tracking methods to calculate these results. The black and blue curves preserve the overall shape of the slow signal, but the green curve fails to do so, which shows that the LM in terms of end position may distort the mean-flow signal. The black and blue curves have differences as well, which are more visible at the steep velocity gradient before $x = \pi$. To understand the reason behind this discrepancy, consider a particle that travels through the relatively flat part of velocity profile in the first half of the averaging interval and through the steep velocity jump in the second half. As a result, the average speed of this particle is small in
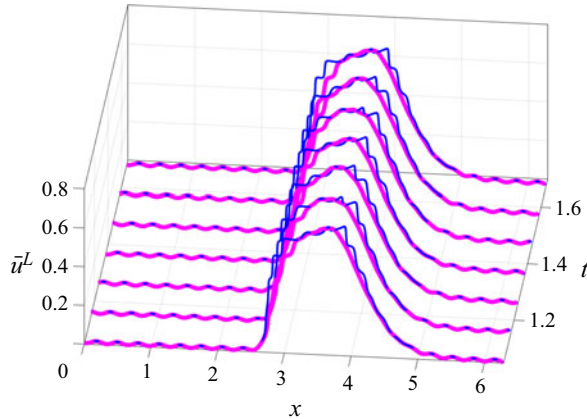
Figure 3. The LM velocity of (3.1*a–c*) over the period of $2\pi/3$, calculated for $\epsilon = 0.15$ at different times using: GBLA in terms of mean position (magenta) and particle tracking in terms of position at the middle of the averaging interval (blue).

the first half and large in the second half, leading to a relatively small change of position in the first half compared with the second half. Therefore, the mean position of this particle differs from its position at the middle of the averaging interval (which is closer to the position at the beginning of the interval).

In figure 3 we increased the wave amplitude to $\epsilon = 0.15$ to illustrate how the GLM definition of LM can be more effective in removing the fast time scale (all other parameters are the same as those used for figure 2*a*). The LM at different times is shown in terms of the mean position calculated by GBLA (magenta curve) and in terms of the position at the middle of the averaging interval calculated by particle tracking (blue curve). There is still a noticeable trace of fast signal in the blue curves, which propagates rightward in time. This is because the particle position at the middle of the averaging period oscillates with the fast time scale, whereas the mean position is less affected by the these oscillations. This example is deliberately chosen to amplify the difference for various choices of spatial labels. For smaller-amplitude waves and milder background velocity gradients the magenta and blue curves converge to each other. Applying Lagrangian filtering to oceanic flows, many authors have used the final position as a label for LM (Shakespeare & Hogg 2017, 2019) or the position at the middle of the averaging interval (Shakespeare *et al.* 2021). If a more accurate removal of fast time scale is desired, expressing LM in terms of mean position should be considered instead.

3.2. *Using a materially conserved quantity to investigate the validity of GBLA*

In this section, we consider the materially conserved quantities – which are constant on each trajectory – to study the accuracy of GBLA. For these quantities, the LM expressed in terms of final position should be equal to their instantaneous values at the end of the averaging interval. We employ this fact to see how reliable GBLA is. We remind the reader that this is merely for studying the validity of GBLA; as discussed before using the GLM definition and expressing the LM in terms of mean position instead of final position is preferred when it comes to the applications of Lagrangian averaging. As a result, step 5 of § 2 is skipped in the calculations of this part.
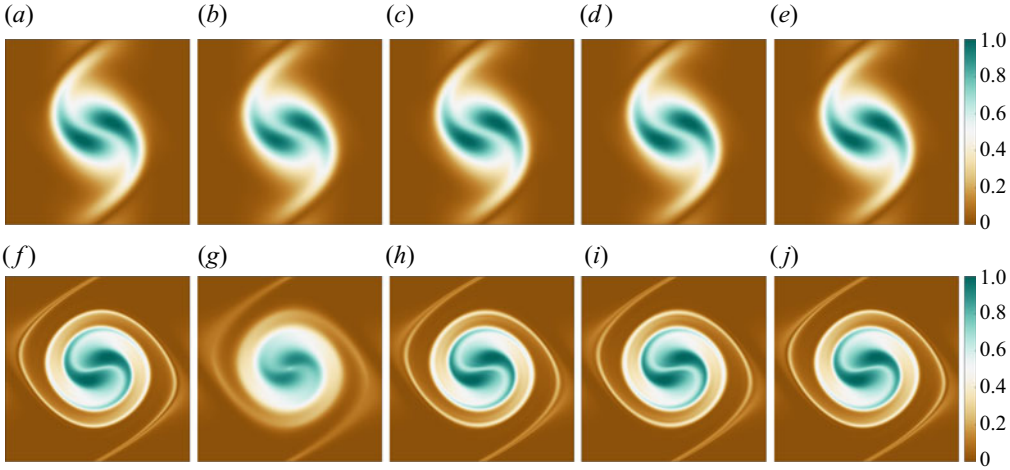
Figure 4. Instantaneous and LM vorticity field for two like-signed vortices. Panels $(a\text{--}e)$ correspond to fields at $t = 10$ and $(f\text{--}j)$ to $t = 30$. (a,f) Instantaneous vorticity; (b,g) GBLA using linear interpolation and (2.7); (c,h) GBLA using cubic interpolation and (2.7); (d,i) GBLA using cubic interpolation and (2.8); (c,h) LM by particle tracking and using cubic interpolation.

3.2.1. *Conservation of vorticity in two-dimensional incompressible flow*

For two-dimensional incompressible inviscid flows, vorticity is conserved on particle trajectories,

$$\frac{\partial \zeta}{\partial t} + u\frac{\partial \zeta}{\partial x} + v\frac{\partial v}{\partial y} = 0, \quad \zeta = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}. \tag{3.3a,b}$$

As a result, $\zeta(X, t) = \overline{\zeta}^L(\Xi^{-1}(X, t), t)$, where $X = (x, y)$. Therefore, to examine GBLA we derive the LM vorticity (expressed in terms of the final particle position) and compare it with the instantaneous vorticity field at the end of the averaging interval. We initialise the numerical simulation with two like-signed vortices,

$$\zeta(x, y, t = 0) = \exp(-(x - \pi + 0.1)^2 - (y - \pi + \pi/3)^2)$$
$$+ \exp(-(x - \pi - 0.1)^2 - (y - \pi - \pi/3)^2). \tag{3.4}$$

We employ a standard pseudo-spectral code to solve (3.3a,b) on a $2\pi \times 2\pi$ domain with periodic boundary conditions using 256×256 grids points. The time step for implementing the averaging process is 10 times coarser than the simulation time step, $\delta t = 0.002$. Figure 4 portrays the LM vorticity calculated by different methods next to the instantaneous vorticity at the end of the averaging interval for two different averaging periods. For the shorter averaging interval, all methods lead to very similar results. However, for the longer interval, GBLA with linear interpolation is less accurate than the others. The comparison between panels (c) and (h), and (d) and (i), shows that the use (2.8) instead of (2.7) has not made any discernible difference, because the interpolation error exceeds other types of error due to the relatively small time step.

To have a more quantitative comparison between different methods, we calculate the following error norm (a two-dimensional version of the norm used in (3.2)):

$$\|e\| = \sqrt{\frac{1}{N_x N_y} \sum_i (\overline{\zeta}^L(\Xi^{-1}(X_i, T), T) - \zeta(X_i, T))^2}, \tag{3.5}$$
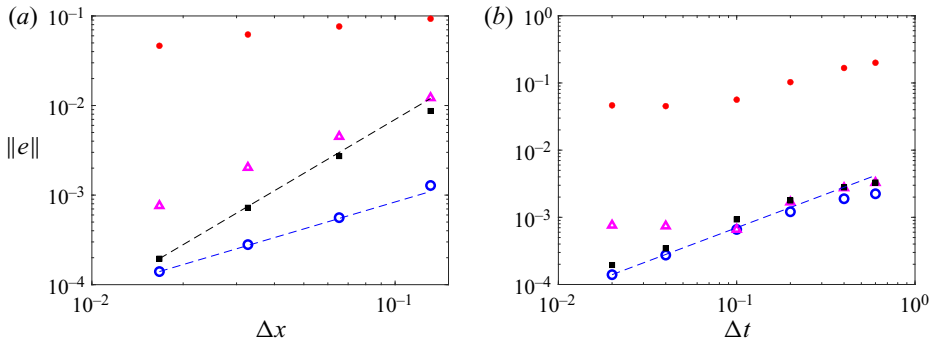
Figure 5. (*a*) Global error using different spatial resolution. (*b*) Global error using different averaging time steps. Error norm is calculated for different schemes namely: GBLA with linear interpolation and using (2.7) (red dots); GBLA with cubic interpolation and using (2.7) (black squares); particle tracking with cubic interpolation (blue circles); and GBLA with linear interpolation and using (2.8) (magenta triangles). The blue dashed lines have the slope of 1, and the black line has the slope of 2.

where the summation is carried out over all spatial grid points, and N_x and N_y are the number of grid points in each dimension. Here $\zeta(X_i, T)$ is an approximation for the exact LM, which is obtained by numerically solving (3.3*a,b*) at a high resolution of 384×384 and small time step of $\delta t = 0.002$. Here $\overline{\zeta}^L$ is calculated using different Lagrangian averaging methods with temporal and spatial distancing that are lowered to investigate their convergence and accuracy.

In figure 5(*a*) we set the averaging time step to $\Delta t = 0.02 = 10\,\delta t$, which is small enough to highlight the interpolation error (see §2.2 for discussion on different sources of error). The GBLA method with linear interpolation and using (2.7) (red dots) underperforms GBLA and particle tracking with cubic interpolation (black squares and blue circles, respectively). However, after using (2.8) even GBLA with linear interpolation (magenta triangles) yields accurate results within the same 'ballpark' as cubic interpolation and particle tracking approaches. With cubic interpolation, the convergence of GBLA is $O((\Delta x)^2)$, and that of particle tracking is $O(\Delta x)$. This is similar to the spatial convergence of error observed in figure 2(*b*) for the one-dimensional synthetic flow. For large Δx, GBLA with cubic interpolation has higher errors than its particle tracking counterpart, but by lowering Δx both methods converge to the same result.

In figure 5(*b*) the convergence of different methods with respect to Δt is investigated. For all the calculations of this plot $\Delta x = \Delta y = 2\pi/384$. Similar to the figure 5(*a*), the error of GBLA with linear interpolation and using (2.7) (red dots) is almost two orders of magnitude higher than the other methods. Use of (2.8) reduces the error of linear methods by more than one order of magnitude (compare the red dots with magenta triangles). At higher Δt, all methods seem to have a slope close to one. However, at small Δt, the methods with linear interpolation plateau as they get dominated by interpolation errors.

Considering that GBLA is an iterative algorithm, different sources of error are intertwined, and tracking their effects separately is complicated. However, from the numerical investigation of global errors in figure 5, we can draw several insightful conclusions. Different variations of GBLA converge robustly by lowering Δx and Δt. The GBLA method is more affected by the interpolation error than the particle tracking, which can be mitigated by using cubic interpolation (or other higher-order schemes). If in some

940 A21-10

applications linear interpolations are selected to lighten the computations, the choice of (2.8) over (2.7) substantially improves the result.

3.2.2. *Conservation of potential vorticity in shallow water*

To put our method to a more challenging test, we consider a fully developed turbulent flow evolving under the rotating shallow-water equations on flat topography,

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \boldsymbol{f} \times \boldsymbol{u} = -g\nabla h, \tag{3.6a}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot (h\boldsymbol{u}) = 0, \tag{3.6b}$$

where h the height field, g the gravitational acceleration and $\boldsymbol{f} = f\hat{\boldsymbol{z}}$ the Coriolis parameter. The existence of vortices of different sizes and sharp fronts in turbulent shallow water systems allow us to better examine the strengths and weaknesses of our method. For this flow, the PV,

$$q = \frac{\nabla \times \boldsymbol{u} + f}{h}, \tag{3.7}$$

is conserved on each particle trajectory. This material conservation can be used in a similar way to the previous section to assess the accuracy of Lagrangian averaging; the LM PV expressed in terms of final position should be equal to the instantaneous PV at the end of the averaging interval. Selecting $f = 0.1$, $g = 0.1$ and the mean height $H = 1$, we solve the nonlinear shallow water equation using a similar pseudo-spectral method with the same computational resolution, domain size and boundary condition as in § 3.2.1. A viscous damping of the form $\nu\nabla^2$ with $\nu = 8 \times 10^{-7}$ is included for numerical stability. We initialise the simulations with a fully developed turbulent flow which is initially in a geostrophic balance (see the initial PV in figure 6b). The time step of the simulation is $\delta t = 5 \times 10^3$, and that of GBLA $\Delta t = 2 \times 10^3$. The instantaneous PV at $t = 120$ in figure 6(a) is remarkably close to the LM PV over the period of [0, 120] in figure 6(c), where small-scale features are also captured by GBLA (see figure 6d for the difference between the two). The small disagreement is partly caused by the viscous damping that makes the conservation of PV less accurate. This is particularly the case for the sharp fronts in figure 6(c) which are dampened in figure 6(a).

3.3. *Stokes terms for standing wave in shallow water*

The difference between the Lagrangian and Eulerian mean is termed Stokes correction (see e.g. Andrews & McIntyre 1978; Bühler 2009) and can be written as

$$\overline{\phi}^S = \overline{\phi}^L - \overline{\phi} = \overline{(\boldsymbol{\xi} \cdot \nabla)\phi'} + O(\epsilon^2), \quad \phi' = \phi(\boldsymbol{x}, t) - \overline{\phi}, \tag{3.8}$$

where the first term in the right-hand side is the leading-order approximation for this quantity. Here ϵ is the amplitude of the wave term ϕ' after proper scaling, where we
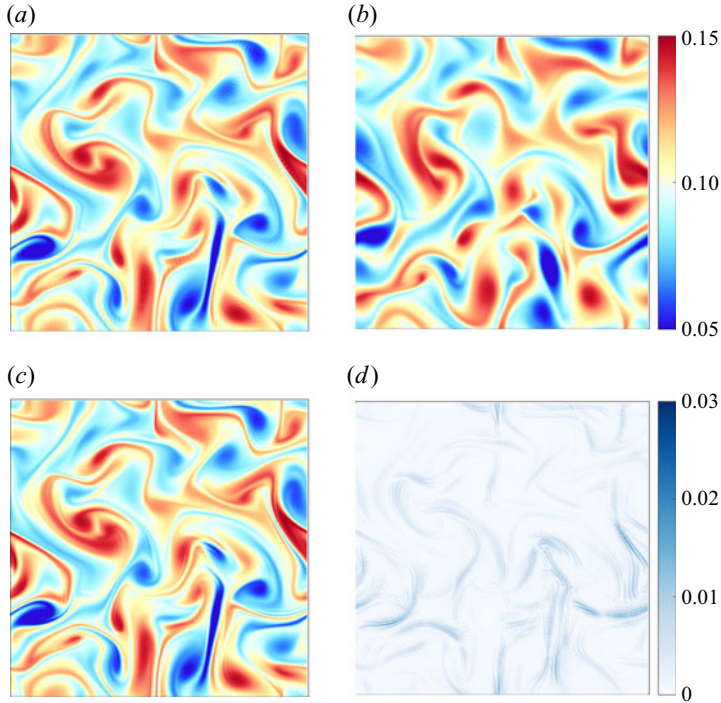
(*a*)   (*b*)

(*c*)   (*d*)

Figure 6. (*a*) The potential vorticity (PV) at the end of the averaging interval $t = 120$. (*b*) The PV at $t = 0$. (*c*) The LM PV from $t = 0$ to $t = 120$ using GBLA with a cubic interpolation scheme. (*d*) Absolute value of the difference between (*a,c*). Panels (*a–c*) share the same colour bar, which is different from that of panel (*d*).

assumed that $\epsilon \ll 1$. We consider a monochromatic standing wave of the form

$$h'(x, y, t) = A \cos(\omega t)\hat{h}(x, y), \quad \hat{h}(x, y) = \cos(kx)\cos(ly), \quad (3.9)$$

$$\boldsymbol{u}'(x, y, t) = \frac{g}{\omega^2 - f^2}(-\omega \sin(\omega t)\boldsymbol{\nabla}\hat{h} + \cos(\omega t)\boldsymbol{f} \times \boldsymbol{\nabla}\hat{h}), \quad (3.10)$$

where A is a small amplitude, is a solution of linear rotating shallow water equations provided that ω satisfies the dispersion relation

$$\omega = \sqrt{f^2 + gH(l^2 + k^2)}. \quad (3.11)$$

Thomas, Bühler & Smith (2018) showed this standing wave induces a mean flow characterised by the Stokes terms

$$\overline{h}^S = \frac{1}{2}\frac{g}{\omega^2 - f^2}\left|\boldsymbol{\nabla}\hat{h}\right|^2, \quad \overline{\boldsymbol{u}}^S = -\left(\frac{g}{\omega^2 - f^2}\right)^2 \boldsymbol{f} \times \boldsymbol{\nabla}\left(\frac{1}{2}\left|\boldsymbol{\nabla}\hat{h}\right|^2 + \frac{\omega^2 - f^2}{4gH}\hat{h}^2\right). \quad (3.12a,b)$$

The above analytical expressions provide a good test case for GBLA. We perform a numerical simulation for

$$A = 0.001, \quad f = g = H = 1, \quad k = l = 1, \quad (3.13a–c)$$

and compare $\overline{h}^L - \overline{h}$ and $\overline{\boldsymbol{u}}^L - \overline{\boldsymbol{u}}$ calculated by the grid-based averaging method with the analytical expressions in (3.12*a,b*). There is an excellent agreement between these two sets of quantities in figure 7.
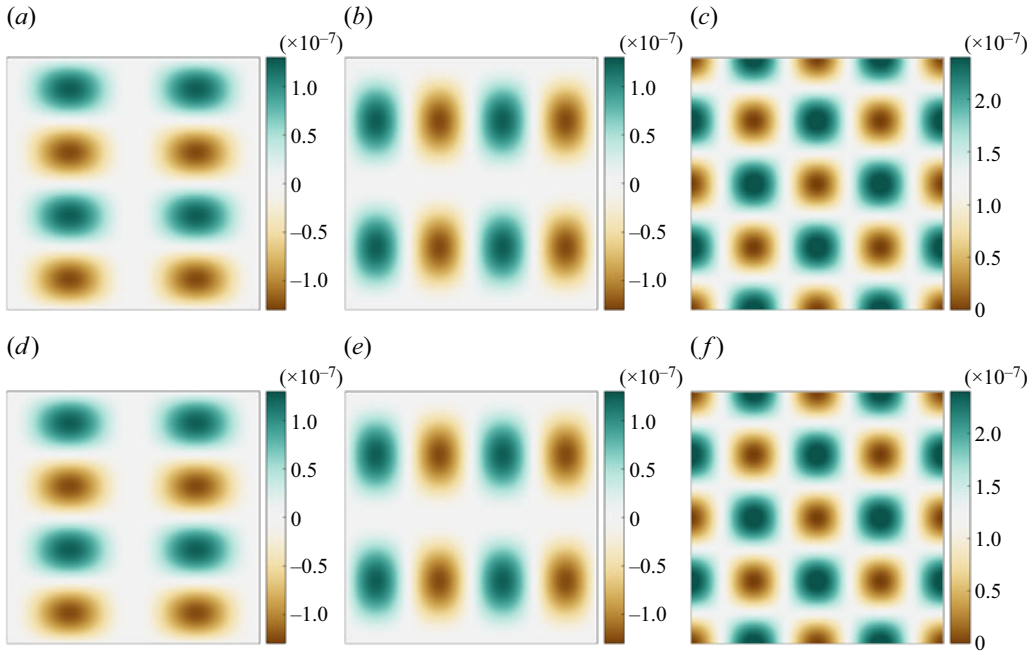
940 A21-12

Figure 7. Stokes terms computed by GBLA (*a–c*) and from analytical expression in (3.12*a,b*) (*d–f*): (*a,d*) $\overline{u}^{S}$; (*b,e*) $\overline{v}^{S}$; (*c,f*) $\overline{h}^{S}$. The averaging interval is set to $2\pi/\omega$.

4. Application: wave-averaged geostrophic balance

Geostrophic balance reliably describes the slow dynamics of large-scale geophysical flows when the waves are absent or relatively small. However, in the presence of waves that are at the same order or stronger than the mean flow, this balance breaks down even after averaging over the fast time scale at fixed points. Generalised Lagrangian-mean theories have predicted that a modified version of geostrophic balance can still hold with coexisting strong waves if it is applied to LM quantities (Moore 1970; Bühler & McIntyre 1998; Wagner & Young 2015; Xie & Vanneste 2015; Thomas *et al.* 2018). Referring to this as wave-averaged balance, Kafiabad *et al.* (2021) numerically illustrated its validity for a single barotropic vortex interacting with strong inertial waves in the context of Boussinesq equations. Instead of directly calculating the LM, they used explicit expressions of Stokes terms derived by Rocha, Wagner & Young (2018) in the limit of near-inertial waves. This approach circumvented the complications of particle tracking in parallelised Boussinesq simulation. However, for baroclinic flows and waves with $\omega \gg f$, explicit expressions for the Stokes terms are not available. Therefore, the LM has to be computed directly, and GBLA makes this computation straightforward. In this study, we touch upon this potential application of GBLA by investigating wave-averaged geostrophy in the context of rotating shallow water.

We first present a brief derivation of wave-averaged balance for the momentum equations in a form that is slightly different than the existing expressions in the literature (a more detailed derivation for shallow water can be found in Thomas *et al.* (2018) and for the Boussinesq equations in Wagner & Young (2015)). We then quantify to what extent this balance holds for waves with different parameters and compare it with Eulerian-averaged geostrophic balance. Similar to preceding studies, the underlying assumption in our

derivation is that the flow consists of fast waves with small amplitude $\epsilon \ll 1$, interacting with a slow flow of the order ϵ^2. This relative scaling of waves and slow flow is more a matter of convenience than necessity as the effect of waves on the mean flow appears at lower orders. For waves at the same order as the slow flow, similar results can be obtained with more tedious algebra and expansions to higher orders. Considering these assumptions, we expand flow variables as

$$\boldsymbol{u} = \epsilon \boldsymbol{u}_1 + \epsilon^2 \boldsymbol{u}_2 + O(\epsilon^3), \tag{4.1}$$

where $\boldsymbol{u}_1$ denotes the wave terms with $\overline{\boldsymbol{u}_1} = 0$, and the mean flow at the leading order is $\bar{\boldsymbol{u}} = \epsilon^2 \overline{\boldsymbol{u}_2} + O(\epsilon^3)$ (a similar expansion is carried out for h). Additionally, the mean flow is assumed to vary on the slow time scale of quasigeostrophic dynamics, leading to $\partial \overline{\boldsymbol{u}_2}/\partial t = O(\epsilon^2)$. At order ϵ, the expansion of (3.6a) leads to

$$\frac{\partial \boldsymbol{u}_1}{\partial t} + \boldsymbol{f} \times \boldsymbol{u}_1 = -g \boldsymbol{\nabla} h_1. \tag{4.2}$$

The next-order equation after averaging is

$$\overline{\boldsymbol{u}_1 \cdot \boldsymbol{\nabla} \boldsymbol{u}_1} + \boldsymbol{f} \times \overline{\boldsymbol{u}_2} = -g \overline{\boldsymbol{\nabla} h_2}. \tag{4.3}$$

After taking the time derivative of (2.2) and some algebraic manipulation, one can show that $\boldsymbol{u}_1 = \partial \boldsymbol{\xi}/\partial t + O(\epsilon)$ (see Andrews & McIntyre (1978) for the detailed derivation). As a result,

$$\overline{\boldsymbol{u}_1 \cdot \boldsymbol{\nabla} \boldsymbol{u}_1} = -\overline{\boldsymbol{\xi} \cdot \boldsymbol{\nabla} \frac{\partial \boldsymbol{u}_1}{\partial t}}. \tag{4.4}$$

Substituting $\partial \boldsymbol{u}_1/\partial t$ from (4.2) in the above equation and using (3.8), we derive

$$\overline{\boldsymbol{u}_1 \cdot \boldsymbol{\nabla} \boldsymbol{u}_1} = \boldsymbol{f} \times \overline{\boldsymbol{\xi} \cdot \boldsymbol{\nabla} \boldsymbol{u}_1} + g \overline{\boldsymbol{\xi} \cdot \boldsymbol{\nabla}(\boldsymbol{\nabla} h_1)} = \boldsymbol{f} \times \bar{\boldsymbol{u}}^S + g \overline{\boldsymbol{\nabla} h}^S, \tag{4.5}$$

(4.5) and (4.3) lead to

$$\boldsymbol{f} \times \left(\bar{\boldsymbol{u}} + \bar{\boldsymbol{u}}^S \right) = -g \left(\overline{\boldsymbol{\nabla} h} + \overline{\boldsymbol{\nabla} h}^S \right), \tag{4.6}$$

where we replaced $\overline{\boldsymbol{u}_2}$ with $\bar{\boldsymbol{u}}$ (and removed the book-keeping parameter ϵ). Rewriting (4.6) in terms of LM variables gives to the final form of wave-averaged geostrophy

$$\boldsymbol{f} \times \bar{\boldsymbol{u}}^L = -g \overline{\boldsymbol{\nabla} h}^L. \tag{4.7}$$

We emphasise that $\overline{\boldsymbol{\nabla} h}^L$ is the LM of the height gradient which is different from the gradient of the LM height $\boldsymbol{\nabla} \bar{h}^L$. The right-hand side of (4.7) can also be written as $-g \boldsymbol{\nabla} (\bar{h} + \bar{h}^S/2)$ (see e.g. Thomas *et al.* (2018) or Wagner & Young (2015) for the counterpart pressure gradient in the Boussinesq equations). We prefer to keep the current form of (4.7) as expressions in terms of $\bar{h}^S/2$ unnecessarily complicate the computation and derivation.

To verify (4.7), we study two examples of slow flow interacting with strong waves. In the first example, we consider an initial condition that consists of a Gaussian anticyclone similar to what was studied by Kafiabad *et al.* (2021). The vertical vorticity of this vortex is defined as

$$\zeta(r, t = 0) = -\zeta_{max} \, \mathrm{e}^{-r^2/a^2}, \tag{4.8}$$

where r is the radial coordinate, a the vortex radius and ζ_{max} the initial maximum vorticity. The velocity fields associated with this vorticity are set to zero far from the vortex, and the
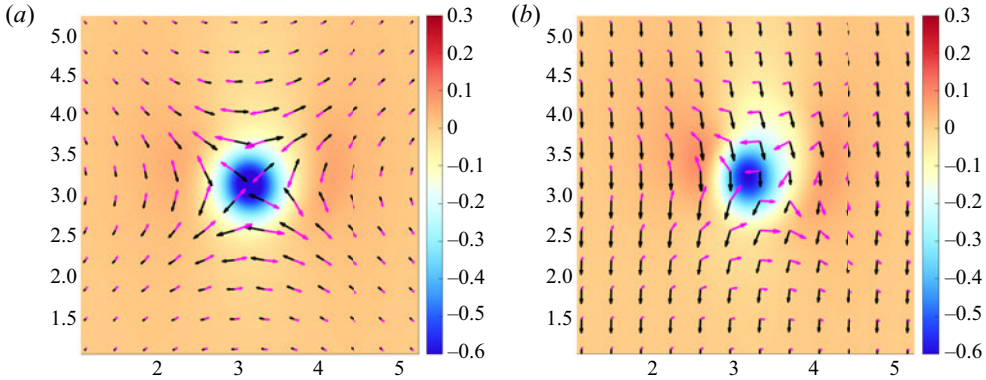
Figure 8. The vectors of Lagrangian- and Eulerian-mean Coriolis forces and height gradients (scaled by g) using GBLA: black arrows, $\boldsymbol{f} \times \overline{\boldsymbol{u}}^L$ (a) and $\boldsymbol{f} \times \overline{\boldsymbol{u}}$ (b); Magenta arrows, $g\overline{\nabla h}^L$ (a) and $g\nabla\overline{h}$ (b). The background colour shows $\nabla \times \overline{\boldsymbol{u}}^L$ (a) and $\nabla \times \overline{\boldsymbol{u}}$ (b). Only a part of resolved domain is shown.

associated height field is defined such that the geostrophic balance is initially maintained. This vortex is superimposed with a Poincaré wave of the form

$$u' = A\cos x, \quad v' = \frac{f}{\omega}A\sin x, \quad h' = H + \frac{H}{\omega}A\cos x, \qquad (4.9a\text{–}c)$$

where ω satisfies (3.11) (for $k = 1$ and $l = 0$). Selecting $a = 0.4$, $f = 20$, $g = H = 1$, $\zeta_{max} = 0.5$ and $A = 3$, we numerically solve the nonlinear shallow-water equations for three periods of the initial wave and time average the variables over this interval ($3 \times 2\pi/\omega$). The time step – which is the same for both the simulation and averaging – is set to 10^{-4}, and all other parameters are kept the same as those in § 3.2.2.

Figure 8 portrays the vectors of Lagrangian- and Eulerian-mean Coriolis forces and height gradients (scaled by g) using GBLA. It is clear that the geostrophic balance holds very well for the LM values as described by (4.7), whereas the Eulerian-mean version has broken down. In the background $\nabla \times \overline{\boldsymbol{u}}^L$ and $\nabla \times \overline{\boldsymbol{u}}$ are shown, which are slightly different fields. Unlike the initial vortex, $\nabla \times \overline{\boldsymbol{u}}$ (and to a lesser extent $\nabla \times \overline{\boldsymbol{u}}^L$) are not completely radially symmetric due to the wave feedback on the mean flow.

To better investigate the accuracy of (4.7), we take a cross-section of this flow at $y = 0$ in figure 9 and show the profiles of Coriolis force and (scaled) height gradient. The most outstanding difference in this figure is between $-f\overline{u}$ and $g\overline{h}_y$ (the dashed lines in figure 9b), which is expected from the form of the initial wave. Let us assume a hypothetical scenario that the linear wave in (4.9a–c) does not interact with the vortex. The following Stokes expressions can be derived for this linear wave:

$$f\,\overline{\boldsymbol{\xi} \cdot \nabla u'} = f\,\overline{\frac{-A}{\omega}\sin(x - \omega t)\frac{\partial}{\partial x}A\cos(x - \omega t)} = f\frac{A^2}{2\omega} \approx 4.5, \qquad (4.10a)$$

$$f\,\overline{\boldsymbol{\xi} \cdot \nabla v'} = g\,\overline{\boldsymbol{\xi} \cdot \nabla h_x} = g\,\overline{\boldsymbol{\xi} \cdot \nabla h_y} = 0, \qquad (4.10b)$$

where the averaging is carried over one or several wave periods. In the presence of a vortex, the fast component of the flow changes in time leading to more complicated Stokes expressions than those in (4.10). Nevertheless, these values can provide a 'ballpark' estimate for the difference between the Eulerian and Lagrangian velocities. This explains the significant difference between $-f\overline{u}$ and $g\overline{h}_y$ as we expect $\overline{u}^S$ to be the largest
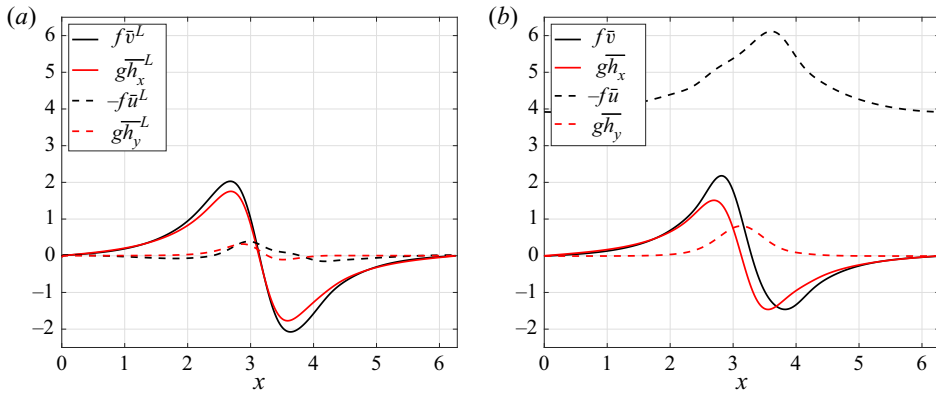
Figure 9. The profiles of different components of Coriolis force and (scaled) height gradient at $y = 0$. All the quantities in (*a*) are Lagrangian averaged, and in (*b*) Eulerian averaged.

Stokes term from (4.10). In fact, several other numerical simulations with different wave amplitudes have been performed to verify that this difference proportionally changes with $fA^2/(2\omega)$. Leaving this major difference aside, $f\bar{v}^L$ and $g\overline{h_x}^L$ are also closer to each other than their Eulerian counterparts. Even after removing the constant shift between $-f\bar{u}$ and $g\overline{h_y}$, $-f\bar{u}^L$ more closely follows $g\overline{h_y}^L$. These observations illustrate the merits of wave-averaged balance over classical geostrophic balance when strong waves are present.

Figure 10 displays the scatter plots of Lagrangian and Eulerian means for gh_x and gh_y as functions of fv and $-fu$, respectively (using all the computational grid points). For perfect geostrophic balance, one expects the data points to lie on $y = x$ (shown in black in figure 10). We consider three simulations with $g = H = 1$, $g = H = 6$ and $g = H = 10$ leading to wave frequencies of $\omega = 20.03$, 20.88 and 22.36, respectively. It is clear that the geostrophic balance holds much better for the LM quantities as the red markers are closer to $y = x$ than the green ones for all three cases. Considering these different wave frequencies implies that (4.7) is a good approximation in the presence of non-inertial waves as well as inertial waves. Unfortunately, we could not investigate much higher wave frequencies due to the formation of shocks at the high wave-amplitude, which is considered in our simulations.

In the next example, we initialise the slow flow with two opposite-sign vortices,

$$u = -\cos y, \quad v = \cos x, \quad h = -\frac{f}{g}(\sin x + \sin y). \quad (4.11a\text{--}c)$$

A similar form of wave to (4.9*a*–*c*) is superimposed with these vortices. We set the flow and wave parameters as $f = 100$, $g = 100$, $H = 4$ and $A = 20$. All other parameters and the numerical set-up are kept the same as the previous example. In figure 11, we observe that the vectors of $f \times \bar{u}^L$ and $g\overline{\nabla h}^L$ balance each other very well (figure 11*a*), whereas the balance for their Eulerian counterparts breaks down (figure 11*b*). The background fields indicate that $\nabla \times \bar{u}$ is more affected by the wave signal than $\nabla \times \bar{u}^L$ is. Further assurance comes from figure 12 where the LM Coriolis force components much more closely follow the LM scaled height-gradient components. Figure 13 also verifies that the Eulerian-mean quantities are farther from $y = x$ and wider spread.
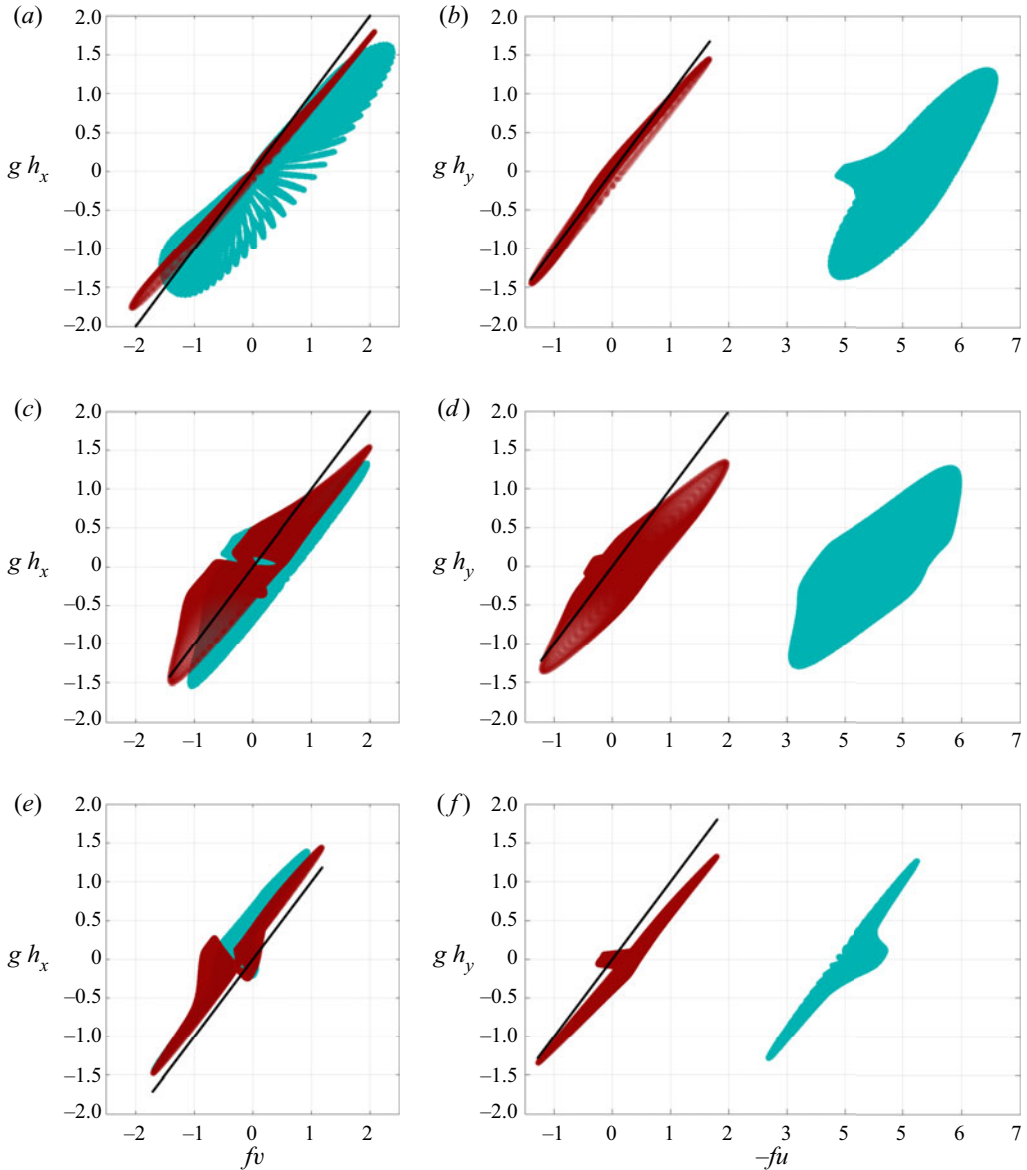
940 A21-16

Figure 10. The scaled height gradient as a function of the Coriolis force after Lagrangian averaging (in red) and Eulerian averaging (in green). The black line is $y = x$, which represents the geostrophic balance. Every marker represents the mean values for a computation grid point in space: (a,b) $g = H = 1$; (c,d) $g = H = 6$; (e,f) $g = H = 10$. All other parameters are kept the same as those in figures 8 and 9.

5. Discussion

Through several examples, we find that GBLA reliably calculates the LM of flow variables and has several advantages over its particle tracking counterparts. Particle-tracking methods are performed either forward or backward in time to calculate LM fields. The forward-time tracking cannot guarantee that the particles at a later time are uniformly distributed. In fact, in regions of confluence the particles get cluttered in a small part of domain. The backward particle tracking does not suffer from this issue but requires storing times series of several fields: the instantaneous value of the desired field, its
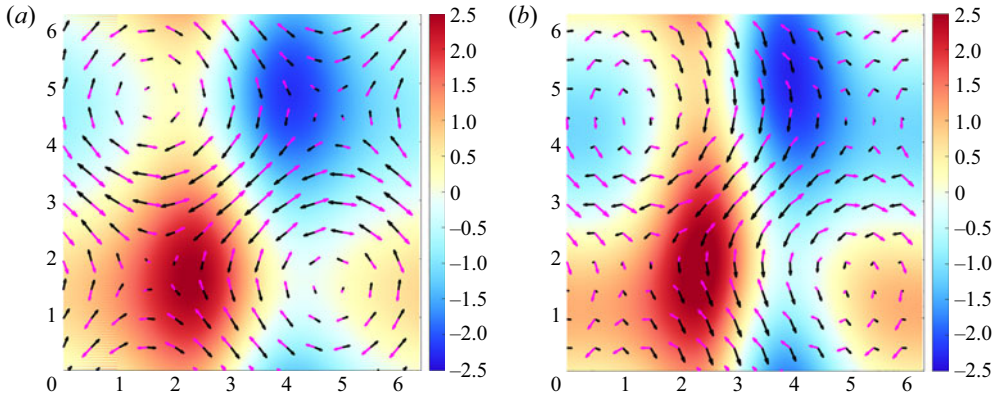
Figure 11. Similar figure to figure 8 for two opposite-sign vortices described in (4.11*a–c*). The entire computational domain is shown in this figure.
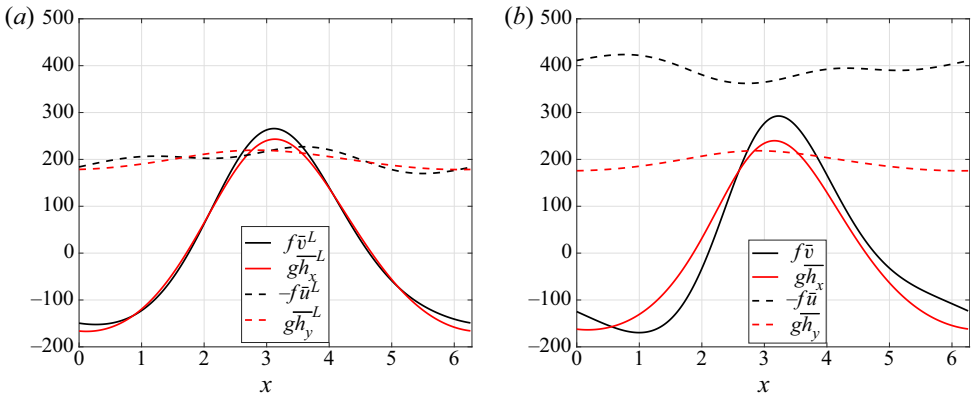

Figure 12. Similar figure to figure 9 for two opposite-sign vortices described in (4.11*a–c*).

interpolated values at particle positions, particle positions along the trajectories and velocity components. The GBLA method does not need to store any time series; it only stores a few fields such as partial means at two time steps. As a result, it can significantly reduce the memory consumption if a large number of grid points or time steps are employed. For the LM of any field other than velocity, GBLA does not require interpolation of velocities, whereas the particle tracking methods do. For instance, assume that the LM temperature needs to be calculated. For particle tracking, in addition to the temperature, the velocity field needs to be interpolated at the particles' instantaneous positions to advect them backward or forward. However, for GBLA there is no need for interpolating velocities, and only one set of interpolation for temperature partial means is needed; as a result, the amount of interpolation in GBLA is at least half of those in particle tracking. Depending on the spatial dimension, it could be even less than half. For example in three-dimensional flows, it is only a quarter of the interpolations required by particle tracking.

All the advantages said, we caution that the GBLA results could be affected by interpolation errors for long averaging periods or coarse spatial resolutions; better interpolation schemes and finer time steps can significantly reduce these errors.
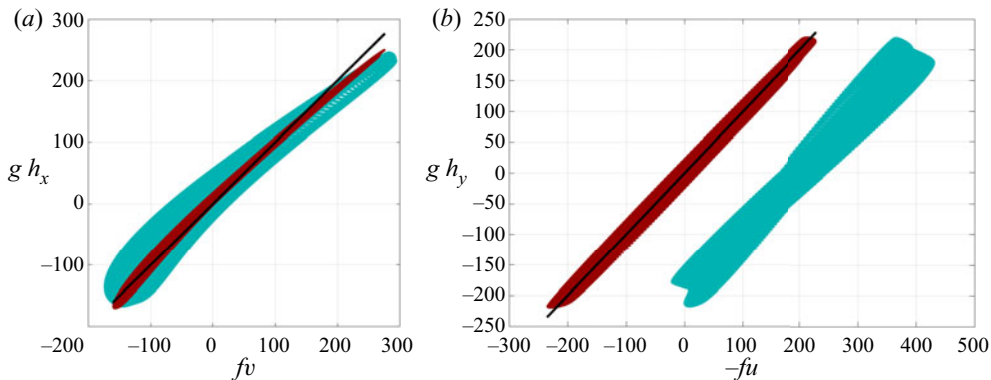
940 A21-18

Figure 13. Similar figure to figure 10 for two opposite-sign vortices described in (4.11*a*–*c*).

Our numerical investigations establish a robust spatial and temporal convergence of GBLA even for linear interpolation. Hence, by using finer grids one obtains reliable LM fields.

In this study, we explained the averaging process only for a single interval, but in some applications the evolution of mean flow or wave field in time is desired. This can be achieved by a moving time-window that frees up the memory from completed intervals and allocates it to the new intervals. Note that GBLA does not require the discretisation in time or space to be uniform. Hence, it can be improved by adaptive time stepping or a more advanced interpolation scheme. Another direction of improvement is by using more advanced filtering schemes. Many studies in geophysical fluids employ low-pass filtering in Fourier space (see e.g. Shakespeare & Hogg 2017; Shakespeare *et al.* 2021), which can be more effective than simple averaging used in (2.3). This filtering (as well as other advanced filtering schemes) can be achieved by using GBLA, if an appropriate kernel is included in the definition of mean in (2.3). Considering other definitions of mean/perturbation than the one used in (2.3) and (2.4) (Soward & Roberts 2010; Gilbert & Vanneste 2018), it is an open question to the author whether a similar approach to GBLA can be used to derive these forms of mean.

We mainly focused on the description and validation of our method and touched upon one application of GBLA. We leave other applications of GBLA for future work. Recent studies such as Shakespeare & Hogg (2017, 2018, 2019) have demonstrated that Lagrangian filtering is more effective in extracting wave signals from the flow. The GBLA method can make Lagrangian filtering more accessible by making the parallel implementation easier and reducing the memory footprint. Another application of GBLA is the study of the reduced models described for LM quantities (e.g. Xie & Vanneste 2015; Wagner & Young 2015), which was briefly considered in this paper. Studying these models in the context of Boussinesq equations is often challenging due to complications of particle tracking in high-resolution parallelised solvers. The GBLA method can also be applied to observations or more realistic global circulation models to gain more insight about the ocean and climate dynamics.

Ultimately, GBLA can improve the parameterisation schemes used in ocean and weather models that need to dissipate energy at much larger scales than viscous scales. Such dissipation – which is required for numerical stability and closing the energy budget – reduces the effective resolution of these models (see e.g. Skamarock 2004). Several studies have shown that the unbalanced and fast processes such as waves dynamics initiate the downscale flux of energy toward dissipation range in the ocean and atmosphere

(Waite & Snyder 2009; Barkan, Winters & McWilliams 2017; Kafiabad & Bartello 2018; Taylor & Straub 2020; Xie 2020). Hence, one can argue that the dissipation of the fast component after removing the LM preserves the slow dynamics better than the less selective dissipation of all fields or their Eulerian high-pass part.

Declaration of interests. The author reports no conflict of interest.

Author ORCIDs.

Hossein A. Kafiabad https://orcid.org/0000-0002-8791-9217.

REFERENCES

ANDREWS, D.G. & MCINTYRE, M.E. 1978 An exact theory of nonlinear waves on a Lagrangian-mean flow. *J. Fluid Mech.* **89**, 609–646.

ASSELIN, O. & YOUNG, W.R. 2020 Penetration of wind-generated near-inertial waves into a turbulent ocean. *J. Phys. Oceanogr.* **50** (6), 1699–1716.

BACHMAN, S.D., SHAKESPEARE, C.J., KLEYPAS, J., CASTRUCCIO, F.S. & CURCHITSER, E. 2020 Particle-based lagrangian filtering for locating wave-generated thermal refugia for coral reefs. *J. Geophys. Res.: Oceans* **125** (7), e2020JC016106.

BARKAN, R., WINTERS, K.B. & MCWILLIAMS, J.C. 2017 Stimulated imbalance and the enhancement of eddy kinetic energy dissipation by internal waves. *J. Phys. Oceanogr.* **47** (1), 181–198.

BRETHERTON, F.P. 1971 The general linearized theory of wave propagation. In *Mathematical Problems in the Geophysical Sciences* (ed. W.H. Reid), Lectures in Applied Mathematics, vol. 13, pp. 61–102. American Mathematical Society.

BÜHLER, O. 2009 *Waves and Mean Flows*. Cambridge University Press.

BÜHLER, O. & MCINTYRE, M.E. 1998 On non-dissipative wave–mean interactions in the atmosphere or oceans. *J. Fluid Mech.* **354**, 301–343.

CHARNEY, J.G. 1971 Geostrophic turbulence. *J. Atmos. Sci.* **28** (6), 1087–1095.

GILBERT, A.D. & VANNESTE, J. 2018 Geometric generalised Lagrangian-mean theories. *J. Fluid Mech.* **839**, 95–134.

GRIMSHAW, R. 1975 Nonlinear internal gravity waves in a rotating fluid. *J. Fluid Mech.* **71** (3), 497–512.

KAFIABAD, H.A. & BARTELLO, P. 2018 Spontaneous imbalance in the non-hydrostatic Boussinesq equations. *J. Fluid Mech.* **847**, 614–643.

KAFIABAD, H.A., VANNESTE, J. & YOUNG, W.R. 2021 Wave-averaged balance: a simple example. *J. Fluid Mech.* **911**, R1.

MCINTYRE, M.E. & NORTON, W.A. 2000 Potential vorticity inversion on a hemisphere. *J. Atmos. Sci.* **57**, 1214–1235, corrigendum: **58**, 2001, 949–949.

MOHEBALHOJEH, A.R. & DRITSCHEL, D.G. 2001 Hierarchies of balance conditions for the f-plane shallow water equations. *J. Atmos. Sci.* **58**, 2411–2426.

MOORE, D. 1970 The mass transport velocity induced by free oscillations at a single frequency. *Geophys. Fluid Dyn.* **1**, 237–247.

NAGAI, T., TANDON, A., KUNZE, E. & MAHADEVAN, A. 2015 Spontaneous generation of near-inertial waves by the Kuroshio front. *J. Phys. Oceanogr.* **45** (9), 2381–2406.

ROCHA, C.B., WAGNER, G.L. & YOUNG, W.R. 2018 Stimulated generation: extraction of energy from balanced flow by near-inertial waves. *J. Fluid Mech.* **847**, 417–451.

SHAKESPEARE, C.J., GIBSON, A.H., HOGG, A.M., KEATING, S.R., BACHMAN, S.D. & VELZEBOER, N. 2021 Lagrangian filtering: a novel method for separating internal waves from non-wave flows in high-resolution simulations. In *Earth and Space Science Open Archive*, https://doi.org/10.1002/essoar.10507016.1.

SHAKESPEARE, C.J. & HOGG, A.M. 2017 Spontaneous surface generation and interior amplification of internal waves in a regional-scale ocean model. *J. Phys. Oceanogr.* **47** (4), 811–826.

SHAKESPEARE, C.J. & HOGG, A.M. 2018 The life cycle of spontaneously generated internal waves. *J. Phys. Oceanogr.* **48** (2), 343–359.

SHAKESPEARE, C.J. & HOGG, A.M. 2019 On the momentum flux of internal tides. *J. Phys. Oceanogr.* **49** (4), 993–1013.

SKAMAROCK, W.C. 2004 Evaluating mesoscale NWP models using kinetic energy spectra. *Mon. Weath. Rev.* **132** (12), 3019–3032.

SOWARD, A.M. & ROBERTS, P.H. 2010 The hybrid Euler–Lagrange procedure using an extension of Moffatt's method. *J. Fluid Mech.* **661**, 45–72.

SUTHERLAND, B.R., ACHATZ, U., COLM-CILLE, P.C. & KLYMAK, J.M. 2019 Recent progress in modeling imbalance in the atmosphere and ocean. *Phys. Rev. Fluids* **4** (1), 010501.

TAYLOR, S. & STRAUB, D. 2020 Effects of adding forced near-inertial motion to a wind-driven channel flow. *J. Phys. Oceanogr.* **50** (10), 2983–2996.

THOMAS, J., BÜHLER, O. & SMITH, K.S. 2018 Wave-induced mean flows in rotating shallow water with uniform potential vorticity. *J. Fluid Mech.* **839**, 408–429.

VALLIS, G.K. 2017 *Atmospheric and Oceanic Fluid Dynamics*. Cambridge University Press.

VANNESTE, J. 2013 Balance and spontaneous wave generation in geophysical flows. *Annu. Rev. Fluid Mech.* **45**, 147–172.

WAGNER, G.L. & YOUNG, W.R. 2015 Available potential vorticity and wave-averaged quasi-geostrophic flow. *J. Fluid Mech.* **785**, 401–424.

WAGNER, G.L. & YOUNG, W.R. 2016 A three-component model for the coupled evolution of near-inertial waves, quasi-geostrophic flow and the near-inertial second harmonic. *J. Fluid Mech.* **802**, 806–837.

WAITE, M.L. & SNYDER, C. 2009 The mesoscale kinetic energy spectrum of a baroclinic life cycle. *J. Atmos. Sci.* **66** (4), 883–901.

WARN, T. 1997 Nonlinear balance and quasi-geostrophic sets. *Atmos. Ocean* **35** (2), 135–145.

XIE, J.-H. 2020 Downscale transfer of quasigeostrophic energy catalyzed by near-inertial waves. *J. Fluid Mech.* **904**, A40.

XIE, J.-H. & VANNESTE, J. 2015 A generalised-Lagrangian-mean model of the interactions between near-inertial waves and mean flow. *J. Fluid Mech.* **774**, 143–169.