A Pressure Projection Scheme With Near-Spectral Accuracy for Nonhydrostatic Flow in Domains With Open Boundaries

K. B. Winters, Mariona Claret, M. -Pascale Lelong, Yann Ourmières

JAMES | Journal of Advances in Modeling Earth Systems

RESEARCH ARTICLE
10.1029/2023MS004040

A Pressure Projection Scheme With Near-Spectral Accuracy for Nonhydrostatic Flow in Domains With Open Boundaries

K. B. Winters[1], Mariona Claret[2,3], M.-Pascale Lelong[3], and Yann Ourmières[4]

[1]Scripps Institution of Oceanography, University of California San Diego, San Diego, CA, USA, [2]Cooperative Institute for Climate, Ocean and Ecosystem Studies, University of Washington, Seattle, WA, USA, [3]NorthWest Research Associates, Seattle, WA, USA, [4]MIO, Universitè de Toulon, Aix Marseille Univ, CNRS, IRD, Toulon, France

Abstract We describe a pressure projection scheme for the simulation of incompressible flow in cubic domains with open boundaries based on fast Fourier transforms. The scheme is implemented in **flow_solve**, a numerical code designed for process studies of rotating, density-stratified flow. The main algorithmic features of the open-boundary code are the near-spectral accuracy of the discrete differentiation and a dynamic two-dimensional domain decomposition that scales efficiently to large numbers of processors. The simulated flows are not required to be periodic or to satisfy symmetry conditions at the open boundaries owing to the use of mixed series expansions combining cosine and singular Bernoulli polynomial basis functions. These expansions facilitate the imposition of inhomogeneous boundary conditions and allow the code to be used for offline, one-way nesting within an arbitrarily embedded subdomain of a larger scale simulation. The projection scheme is designed to exploit a simple and powerful numerical engine: inversion of Poisson's equation with homogeneous Neumann boundary conditions using fast cosine transforms. Here, we describe the mathematical transformations used to accommodate the imposition of space- and time-varying boundary conditions. The utility of the approach for process studies and for nesting within submesoscale-resolving ocean models is demonstrated with simulations of wind-driven near-inertial waves in the upper ocean.

Plain Language Summary Internal gravity waves in the upper ocean are affected by the presence of features such as fronts, eddies, and geographically constrained winds and currents. Ocean models that are able to accurately capture geographic realism often lack the capacity to resolve the intricacies of the internal waves that may be generated. This paper presents a mathematical framework that combines the realism of ocean models with high-fidelity simulation of small-scale dynamics. The framework is demonstrated through a series of examples of increasing complexity, including wave trapping in a wind-perturbed anticyclonic vortex and simulation of near-inertial waves near a density front in the Gulf of Lion. The proposed methodology has the potential to take advantage of the increasing horizontal resolution and fine-scale realism of ocean models, while also relaxing the approximations that limit their ability to resolve the detailed dynamics of the internal wave field.

Key Points:

- A computational framework based on fast spectral transforms is developed for computing nonhydrostatic flow in domains with open boundaries
- Mixed basis function expansions are formulated to differentiate discrete data lacking symmetry or periodicity with near-spectral accuracy
- Nested simulations of wave trapping in a vortex and near-inertial waves near a density front in the Gulf of Lion demonstrate the approach

Supporting Information:

Supporting Information may be found in the online version of this article.

Correspondence to:

K. B. Winters,
kbwinters@ucsd.edu

Citation:

Winters, K. B., Claret, M., Lelong, M.-P., & Ourmières, Y. (2024). A pressure projection scheme with near-spectral accuracy for nonhydrostatic flow in domains with open boundaries. *Journal of Advances in Modeling Earth Systems*, *16*, e2023MS004040. https://doi.org/10.1029/2023MS004040

Received 26 SEP 2023
Accepted 1 MAR 2024

1. Introduction

In geophysical fluid dynamics it is often desired to study a particular set of dynamics that is strongly influenced by the background or ambient flow (e.g., Thomas et al., 2020; Shcherbina et al., 2015; Capet et al., 2008; DiBattista et al., 2002; Booker & Bretherton, 1967, and many others). The dynamical problem is often scale separated, that is, the characteristic length and time scales of the ambient flow are much larger than those characterizing the dynamics of particular interest. Moreover, the ambient flow may be difficult to characterize adequately with an idealization. This broad class of multiscale problems is particularly challenging to attack computationally.

Regional or basin-scale ocean models can now be configured with horizontal grid spacing on the order of a kilometer or less. Many upper ocean processes of interest, however, are not adequately captured using the hydrostatic approximation that is typically invoked. Internal waves excited in these ocean models can be sufficiently short that dispersion is significant, and nonhydrostatic equations are necessary to describe their dynamics and capture the energy transfer to higher frequencies and smaller scales. A well-studied example is the internal tide and the associated nonlinear internal solitary waves observed in the South China Sea (Simmons et al., 2011; Vitousek & Fringer, 2014). While the excitation of the internal tide can be reasonably captured by regional ocean models, its subsequent nonlinear steepening leads to nonhydrostatic wave trains of O(100) m scale that propagate

100s of kilometers. Fully non-hydrostatic simulation of such waves on the basin scale is currently not feasible (Huang et al., 2023; Simmons et al., 2011) due to the computational expense of solving the three-dimensional elliptic equation for nonhydrostatic pressure. In addition, there is increasing interest in simulations of upper ocean dynamics that can capture portions of both the inverse oceanic energy cascade as well as the forward cascade toward small, dissipative scales, which is thought to be initiated at 1–10 km horizontal scales (Balwada et al., 2022), enhanced at frontal structures (Srinivasan et al., 2023), and accomplished by nonhydrostatic internal wave energy transfers (Marino et al., 2015).

Rather than attempting to incorporate nonhydrostatic dynamics into ocean models, which can increase their computational cost by an order of magnitude (Fringer et al., 2006), we take a different approach here and show how high-fidelity spectral methods, often used in idealized studies at small scales, can be used in conjunction with lower-fidelity, large-scale models. These larger scale models, for example, regional or basin-scale ocean circulation models, are capable of computing flows with realistic features such as submesoscale fronts and eddies that coexist with local winds, topography, and large-scale currents. However, they are less capable of resolving detailed dynamics within these features. The objective here is to use the large-scale models to simulate geographically realistic ambient flows and to use a nonhydrostatic spectral model to "fill in the details" in selected subdomains encompassing ambient features of interest.

The use of spectral methods for the nested model allows for accurate integration of the equations of motion. These methods are particularly well suited for the study of internal waves, their nonlinear interactions, and their transition to turbulence. However, they would seem to be problematic in a nested setting, owing to the lack of symmetry or periodicity of the flow within a typical nested domain. The purpose of this paper is to develop an approach that combines mixed series expansions using cosine and singular Bernoulli polynomials as basis functions with a pressure projection scheme that overcomes this difficulty and retains near-spectral accuracy.

The organization of the paper is as follows. The mathematical setting and preliminaries are presented in Section 2, followed by the continuous version of the pressure projection algorithm for flow in domains with permeable boundaries in Section 3. Central to the algorithm is the mixed expansion technique to differentiate functions lacking symmetries in finite domains. This technique is described in detail in Section 4. An explicit demonstration of the projection scheme, implemented in **flow_solve** (Winters & de la Fuente, 2012), is described in Section 5. In this first example, the prescribed time-dependent boundary conditions are known analytically. A doubly nested simulation of an idealized upper ocean flow is described in Section 6. In this example, the boundary data for child domain simulations are extracted from a coarser resolution parent simulation spanning a larger domain. This example exposes a fundamental inconsistency for the algorithm presented in Section 3. A modification to the algorithm that recovers consistency is presented in Section 6.2. Finally, in Section 7, we describe the nesting of nonhydrostatic **flow_solve** within a 1/64° regional simulation of the Gulf of Lion made with the hydrostatic ocean general circulation model NEMO (Madec & Team, 2022). The **flow_solve** equations of motion and specializations for the example problem configurations are provided in Appendix A.

2. Preliminaries

We write the Eulerian equations of motion for an incompressible fluid in the domain Ω, bounded by the surface $\partial\Omega$, as

$$\frac{\partial}{\partial t}\mathbf{u} = \mathbf{F} - \nabla p \tag{1}$$

and

$$\nabla \cdot \mathbf{u} = 0, \tag{2}$$

where $\mathbf{u}$ is the fluid velocity vector, p is the pressure and t is time. $\mathbf{F}$ is an acceleration vector, for example, due to advection, buoyancy, rotation, and friction. We need not be more explicit here and simply define the vector

$$\mathbf{u}_* = \mathbf{u}^n + \int_{t^n}^{t^{n+1}} \mathbf{F}\, \mathrm{d}t, \tag{3}$$

where the superscript indicates the time value and $(t^{n+1} - t^n)$ is a small time interval Δt. The integral in Equation 3 is treated as exact in what follows, but in practice it is approximated using a discrete explicit multilevel integration scheme of desired accuracy. Similarly, we will describe our algorithmic approach primarily in continuous form, introducing discrete notation for spatial aspects as necessary. We need not consider explicit forms for coupled equations for scalars or the equation of state, other than to note that the Boussinesq approximation has been made and that the inverse of the reference density ρ_0 has been absorbed into p.

An exact solution at time t^{n+1} is

$$\mathbf{u}^{n+1} = \mathbf{u}_* - \nabla P \tag{4}$$

where

$$P = \int_{t^n}^{t^{n+1}} p \, \mathrm{d}t. \tag{5}$$

In writing this expression, we regard $\mathbf{u}_*$ as known and the scalar field P as unknown. The role of P is two-fold. It constrains $\mathbf{u}^{n+1}$ to be incompressible and satisfy Equation 2 while also ensuring that $\mathbf{u}^{n+1}$ satisfies normal-flow constraints at the boundaries. Our objective is to formulate an algorithm to solve for P that makes use of a straightforward cosine transform-based inversion of Poisson's equation with homogeneous Neumann boundary conditions as the main computational tool. This core algorithm has exponential spatial accuracy at the cost of a forward and inverse three-dimensional cosine transform.

Taking the divergence of Equation 4 and requiring that $\nabla \cdot \mathbf{u}^{n+1} = 0$ gives

$$\nabla^2 P = \nabla \cdot \mathbf{u}_* \tag{6}$$

which is to hold throughout Ω. The boundary conditions for this Poisson equation are also obtained from Equation 4, namely

$$\nabla P \cdot \hat{\mathbf{n}} = \left(\mathbf{u}_* - \mathbf{u}^{n+1}\right) \cdot \hat{\mathbf{n}} \tag{7}$$

where $\hat{\mathbf{n}}$ is the unit vector normal to the bounding surface $\partial\Omega$. The vector $\hat{\mathbf{n}}$ is uniquely defined in $\partial\Omega$, except at the seams and corners of the domain, where it can be considered undefined or multiple-valued. Here, we assume that the boundary conditions for the normal flow $\mathbf{u}^{n+1} \cdot \hat{\mathbf{n}}$ are known. For example, the normal flow vanishes at solid walls and, in principle, can be extracted from a parent simulation in a one-way nesting application. For idealized problems, it can be prescribed analytically over all or portions of the boundary and inferred as part of the solution over the remainder as necessary.

Time-stepping schemes in which an intermediate velocity vector $\mathbf{u}_*$ is corrected by solving an elliptic equation for P and projecting onto a divergence-free subspace with ∇P are known as projection methods (Chorin, 1968; Temam, 1969). These methods have been widely applied in computational fluid dynamics, yet their ability to correctly treat general boundary conditions has received relatively little attention in the literature (Guermond et al., 2006).

3. Pressure Projection With Flow-Permeable Boundaries

We begin by considering the approach for open domains with a specified time-dependent flow through the boundary $\partial\Omega$. The Neumann conditions in Equation 7 are inhomogeneous and so we are unable to use a cosine transform method directly. Rather, we take a lifting approach and reformulate the problem to expose one that has the required form, that is, a Poisson equation with homogeneous Neumann boundary conditions.

Let

$$P = \psi + \phi \tag{8}$$

and, for the moment, regard ψ and ϕ as known and unknown scalar fields respectively. Equation 6 then implies that

$$\nabla^2\phi = \nabla\cdot\mathbf{u}_* - \nabla^2\psi = \nabla\cdot(\mathbf{u}_* - \nabla\psi). \tag{9}$$

To make use of a cosine transform inversion for ϕ, we insist that

$$\nabla\phi\cdot\hat{\mathbf{n}} = 0 \quad \text{on} \quad \partial\Omega. \tag{10}$$

The field ϕ satisfies a Poisson equation with homogeneous Neumann boundary conditions and a source term that depends on both $\mathbf{u}_*$ and ψ . ϕ is uniquely defined up to an arbitrary constant that we take to be equal to zero.

Equations 7 and 8 impose a constraint on ψ that must be satisfied at the boundaries:

$$\nabla\psi\cdot\hat{\mathbf{n}} = \left(\mathbf{u}_* - \mathbf{u}^{n+1}\right)\cdot\hat{\mathbf{n}} \quad \text{on} \quad \partial\Omega \tag{11}$$

where $\mathbf{u}^{n+1}$ values are known and prescribed at the boundaries.

The scalar field ψ is not unique. It has a normal derivative constraint at each boundary point, but its behavior in the interior of Ω is unconstrained. We only require sufficient smoothness so that $\nabla^2\psi$ is well-behaved throughout Ω. A family of allowable fields ψ can be defined as solutions to a pseudo-time diffusion equation with specified flux boundary conditions.

We write

$$\frac{\partial}{\partial\tau}\psi = \kappa_x\,\psi_{xx} + \kappa_y\,\psi_{yy} + \kappa_z\,\psi_{zz} \quad \text{in} \quad \Omega. \tag{12}$$

Here, τ is the pseudo-time variable, κ_x, κ_y, and κ_z are diffusivity coefficients, and double subscripts indicate spatial partial derivatives. Anisotropic diffusion allows the accommodation of anisotropic discretization of Ω. We impose the known boundary condition (11) along with a smooth initial condition, for example, $\psi = 0$.

This problem defines a field ψ that satisfies the required boundary conditions by construction. In practice we also require that ψ be computationally inexpensive in comparison to cosine transform operations. A related requirement is that ψ be smooth enough so that Equation 12 can be solved accurately using finite-difference methods. Our strategy is to specify the diffusivities so that integration over just a few pseudo-time steps diffuses information from the boundaries into the interior over a distance of few grid spacings in each direction. In principle, the smoothness of the solution minimizes finite differencing errors and can be controlled by the choice of the integration time. If N is the total number of spatial grid points in three dimensions, each pseudo-time step in the integration of the diffusion equation for ψ is an $O(N)$ procedure. Integrating a few steps is therefore inexpensive compared to the $O(N\log N^{1/3})$ operation count for algorithmic components based on three dimensional cosine transforms.

While this strategy is designed to maximize smoothness, it is instructive to consider the behavior of ψ near the seams and corners of the domain where $\hat{\mathbf{n}}$ is not uniquely defined. For illustration, we consider the problem in two dimensions near a corner. Figure 1 is schematic of the constraints on ψ. The boundary values ψ_z and ψ_x are to be specified and depend on the mismatch between the $\mathbf{u}_*$ values obtained in the time integration (3) and the prescribed normal flow boundary values at time t^{n+1}. In order for $\psi(x,z)$ to be smooth, the partial derivatives ψ_{xz} and ψ_{zx} must match at the corner point, that is, the cross derivatives must be path independent in the continuous limit. If the prescribed boundary data do not satisfy this constraint, then, in particular, $\nabla^2\psi$ is not smooth in the neighborhood of the corner and the source term for Equation 9 will be poorly behaved.

Smooth boundary data at the corner implies that

$$\left(u_* - u^{n+1}\right)_z = \left(w_* - w^{n+1}\right)_x \quad \text{at the corner.} \tag{13}$$

Rearranging gives $(u_*)_z - (w_*)_x = u_z^{n+1} - w_x^{n+1}$ or, defining $\zeta = u_z - w_x$, that

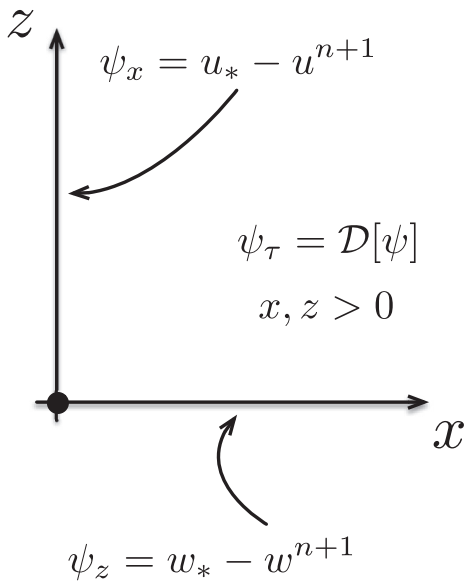

Figure 1. Schematic of the diffusion equation for $\psi(x, z)$ near a corner point $x = z = 0$. Subscripts indicate partial differentiation and $\mathcal{D}[\psi] = \kappa_x \psi_{xx} + \kappa_z \psi_{zz}$ is a two-dimensional diffusion operator.

$$\zeta_* = \zeta^{n+1} \quad \text{at the corner.} \tag{14}$$

Smoothness of the boundary data requires that the vorticity of the intermediate velocity $\mathbf{u}_*$ obtained from Equation 3 matches the vorticity of the prescribed normal-flow boundary conditions at the corner. More generally, the intermediate vorticity $(\nabla \times \mathbf{u}_*) \cdot \hat{t}$ must match the corresponding prescribed vorticity $\left(\nabla \times \mathbf{u}^{n+1}\right) \cdot \hat{t}$ at all points where two boundary planes intersect and $\hat{t}$ is the unit vector along the seam of the intersecting planes. At genuine corners, where two seams intersect, this condition must hold along both seam directions.

Taking the curl of Equation 1 and integrating from t^n to t^{n+1} gives

$$\nabla \times \mathbf{u}^{n+1} = \nabla \times \mathbf{u}^n + \int_{t^n}^{t^{n+1}} \nabla \times \mathbf{F}\, \mathrm{d}t. \tag{15}$$

The same result is obtained by taking the curl of Equation 3. This shows that, in the continuous limit, the vorticity condition (14) at the corner will be satisfied provided that the prescribed normal flow at the boundaries is exactly known. This turns out to be an important caveat that is not generally satisfied in nesting applications. For perfect boundary data, ψ, defined by Equations 11 and 12, is smooth and well-behaved throughout Ω. It is a relatively easily computed auxiliary field that absorbs the inhomogeneous boundary data, leaving ϕ to satisfy the computationally convenient Equation 9 with homogeneous Neumann conditions.

The projection scheme with perfect boundary information is summarized in Algorithm 1.

Algorithm 1. Pressure Projection in a Flow-Permeable Right Rectangular Prism

(a) Compute $\mathbf{u}_* = \mathbf{u}^n + \int_{t_n}^{t_{n+1}} \mathbf{F}\, \mathrm{d}t$ as defined in Equations 1–3.
(b) Calculate the mismatch $\left(\mathbf{u}_* - \mathbf{u}^{n+1}\right) \cdot \hat{\mathbf{n}}$ on $\partial\Omega$.
(c) Solve for ψ using Equation 12 subject to Equation 11 over a few pseudo-time steps.
(d) Calculate $\nabla\psi$ and compute $\widetilde{\mathbf{u}}_* = \mathbf{u}_* - \nabla\psi$.
(e) Calculate $\nabla \cdot \widetilde{\mathbf{u}}_*$ and solve Equation 9 for ϕ using cosine transforms.
(f) Calculate $\nabla\phi$ and project: $\mathbf{u}^{n+1} = \widetilde{\mathbf{u}}_* - \nabla\phi$.
(g) Save the pressure $p = \frac{1}{\Delta t}(\phi + \psi)$.

Once the intermediate velocity field $\mathbf{u}_*$ has been projected through the action of ∇P, the time-integrated field $\mathbf{u}^{n+1}$ satisfies the momentum Equation 1 and the incompressibility condition (2) in Ω and the prescribed normal flow boundary conditions in $\partial\Omega$. The tangential components of the flow field at the boundaries are not explicitly constrained. The projection scheme itself does not constrain the boundary values of the scalar equation(s); these must be incorporated in the treatment of the evolution equation for the scalar variable(s).

The projection algorithm takes advantage of the fast and accurate inversion of the Poisson equation for ϕ using cosine transforms. The boundary inhomogeneity inherent in the pressure field is handled by introducing a smooth auxiliary field ψ. Although ψ itself is computed using finite differencing to allow for flux boundary conditions, the errors incurred are minimized by integrating sufficiently in pseudo-time (τ) so that ψ is very smooth in the interior. Regarding ψ as accurately computed, we then insist on spectral or near-spectral accuracy for all spatial differentiation steps implied by the gradient and divergence operators in Algorithm 1.

Spectral accuracy is achieved formally if the underlying fields to be differentiated are even symmetric, that is, have zero normal derivative, at the boundary $\partial\Omega$. In this case, the field can be expanded in separable cosine series,

the spectral coefficients multiplied by an appropriate set of wavenumbers, followed by an inverse cosine transform of the result. Similar procedures using sine or Fourier series can be used if the fields have odd symmetry or are periodic, respectively. However, for flow-permeable boundaries, none of these symmetries holds. The various fields in Algorithm 1 are neither symmetric nor periodic at the domain boundaries. Though a smooth function in a finite domain can always be represented in a cosine series, naive differentiation in transform space leads to Gibbs oscillations unless the function has even symmetry.

4. Discrete Spatial Derivatives With Near-Spectral Accuracy

For flows with open boundaries, none of the computed solutions satisfy simple symmetry conditions at the boundaries, rendering a standard spectral differentiation approach based on cosine series expansions problematic. For these problems, we adopt a spatial differentiation scheme that utilizes a combination of basis functions, combining cosine series with singular, periodic Bernoulli polynomials. This mixed expansion approach retains near-spectral accuracy while relaxing the even-symmetry constraint. In this section, we develop the mixed series expansion approach to differentiation in some detail, confining the discussion to one spatial dimension for simplicity. It is straightforward to extend the ideas to separable expansions in multiple dimensions. The Bernoulli-Cosine differentiation scheme is used in the gradient and divergence operations in Algorithm 1.

For functions $f(x)$ defined in the closed interval $x \in [0, L]$, the even extension of f into the domain $[0, 2L)$ is, by construction, a continuous $2L$ periodic function. Unless the derivatives of f vanish at $x = 0$ and L, the periodic extension of f has discontinuous derivatives at $x = 0$ and $x = L$. Regardless, $f(x)$ has an $N + 1$ term cosine series expansion that matches the function values at $N + 1$ evenly spaced grid points in the closed interval $[0, L]$. The problem, however, is that $f(x)$ cannot be differentiated by the simple procedure of cosine transformation followed by wavenumber multiplication and inverse transformation due to the discontinuities in the derivative associated with the even extension to $x \in [0, 2L)$. As a discrete differentiation scheme, direct differentiation using fast cosine transforms leads to unacceptable Gibbs oscillation of the computed derivatives of functions f lacking boundary symmetry (Gottlieb & Shu, 1997; Lanczos & Boyd, 2016; Strang, 1999). The objective here is to remove the Gibbs phenomena while retaining much of the accuracy and efficiency of differentiation using fast cosine transforms.

4.1. The Mixed Bernoulli-Cosine Expansion Approach

Let the even extension of $f(x)$, known over the finite domain $0 \le x \le L$, be decomposed as follows (Eckhoff, 1993, 1998)

$$f(x) = f_Q(x) + \sum_{n=1,\text{odd}}^{Q} a_n U_n(x) + \sum_{n=1,\text{odd}}^{Q} b_n U_n(x - L),\ x \in [0,2L) \tag{16}$$

where Q is a relatively small odd integer. The functions $f(x)$ and $U_n(x)$ are $2L$ periodic and U_n is related to the Bernoulli polynomial (e.g., Costabile & Dell' Accio, 2001; Komatsu & Pita, 2016) of degree $n + 1$ defined on the closed interval [0,1] as follows:

$$U_n(x) = -\frac{(2L)^n}{(n+1)!} B_{n+1}\left(\frac{x}{2L}\right). \tag{17}$$

The two series are expansions in terms of even Bernoulli polynomials about the two singular points 0 and L. Odd-order Bernoulli polynomials are excluded due to symmetry, and $B_1(x)$ is not necessary because the function f itself is continuous at the singular points. The form of the expansions is chosen to match the continuity properties of f. For simplicity of notation, the two series in Equation 16 are denoted $S_0(x)$ and $S_L(x)$ so that

$$f(x) = f_Q(x) + S_0(x) + S_L(x). \tag{18}$$

We require that S_0 match the singular behavior of $f(x)$ near $x = 0$ while being well behaved in the interior of the domain with a vanishing derivative at $x = L$. Similarly, S_L must match the singular behavior near $x = L$.

The two series have useful properties. First, the Bernoulli polynomials can be differentiated semi-analytically, that is, in terms of Bernoulli function evaluations:

$$\frac{\mathrm{d}}{\mathrm{d}x}B_n(x) = n\,B_{n-1}(x). \tag{19}$$

Second, $U_1(x)$, and thus $S_0(x)$, has a discontinuous derivative at the left singular point $x = 0$ but is even symmetric, that is, has zero derivative at the right singular point $x = L$. Similarly, $U_1(x - L)$ and $S_L(x)$ have discontinuous derivatives at $x = L$ and zero derivatives at $x = 0$. These properties are inherited from the behavior of $B_2(x)$, the lowest-order polynomial in the expansions. All higher-order terms in the expansions are even Bernoulli polynomials and have zero derivatives at both singular points 0 and L.

The function $f_Q(x) = f(x) - S_0(x) - S_L(x)$ is therefore continuous everywhere in $0 \le x < 2L$ and Q times differentiable. Moreover, its first derivatives vanish at the singular points $x = 0$ and L. Thus, it is amenable to decomposition in a well-behaved, term-by-term differentiable cosine series over the closed domain $0 \le x \le L$.

4.2. Computing the Expansion Coefficients

Given the properties of the two series S_0 and S_L, it is straightforward to design constraints to determine the expansion coefficients $\{a_n\}$ and $\{b_n\}$. There are $M = (Q + 1)/2$ terms in each series. For the $N \gg M$ discrete grid points $x_i = iL/(N + 1)$ in $0 \le x \le L$, the constraints are simply

- $S_0(x_i) = f(x_i) \quad i = 0, 1, \ldots M - 1$
- $S_L(x_i) = f(x_i) \quad i = N - M + 1, N - M + 2, \ldots N$

that is, that the two series match the function values $f(x)$ at the first and last M points in $0 \le x \le L$.

These constraints correspond to small $(M \times M)$ linear systems for the expansion coefficients and can be written in matrix form as follows:

$$\begin{bmatrix} U_1(x_0) & U_3(x_0) & U_5(x_0) & \ldots & U_Q(x_0) \\ U_1(x_1) & U_3(x_1) & U_5(x_1) & \ldots & U_Q(x_1) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ U_1(x_{M-1}) & U_3(x_{M-1}) & U_5(x_{M-1}) & \ldots & U_Q(x_{M-1}) \end{bmatrix} \begin{bmatrix} a_1 \\ a_3 \\ \ldots \\ a_Q \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ \ldots \\ f(x_{M-1}) \end{bmatrix} \tag{20}$$

and, for $j = N - M + 1$,

$$\begin{bmatrix} U_1(x_j - L) & U_3(x_j - L) & U_5(x_j - L) & \ldots & U_Q(x_j - L) \\ U_1(x_{j+1} - L) & U_3(x_{j+1} - L) & U_5(x_{j+1} - L) & \ldots & U_Q(x_{j+1} - L) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ U_1(x_N - L) & U_3(x_N - L) & U_5(x_N - L) & \ldots & U_Q(x_N - L) \end{bmatrix} \begin{bmatrix} b_1 \\ b_3 \\ \ldots \\ b_Q \end{bmatrix} = \begin{bmatrix} f(x_j) \\ f(x_{j+1}) \\ \ldots \\ f(x_N) \end{bmatrix} \tag{21}$$

For $Q = 7$ these are 4×4 linear systems. Constructing the matrices involves evaluations of low- and even-order Bernoulli polynomials at a few near-boundary grid points. The coefficient matrices are independent of the values of f and therefore can be constructed and decomposed into LU triangular factors once during the initialization of the overall algorithm. They are then available for many different data vectors $f(x_i)$. Determining the expansion coefficients for the S_0 and S_L series for a given function f is therefore computationally inexpensive. Given the coefficients, the evaluation of the two series over the entire domain is an $O(N)$ operation.

Given the recursion relation (19), the analytical evaluation of the derivatives of S_0 and S_L over the domain is also an $O(N)$ set of Bernoulli polynomial function evaluations. The function f_Q is obtained by subtracting S_0 and S_L from $f(x)$. Because it has zero derivatives at $x = 0$ and L, it can be accurately differentiated in wavenumber space using its N term cosine series. This is an $O(N \log N)$ operation. The derivative of f is then obtained by adding the derivative estimates for the three components of Equation 16.

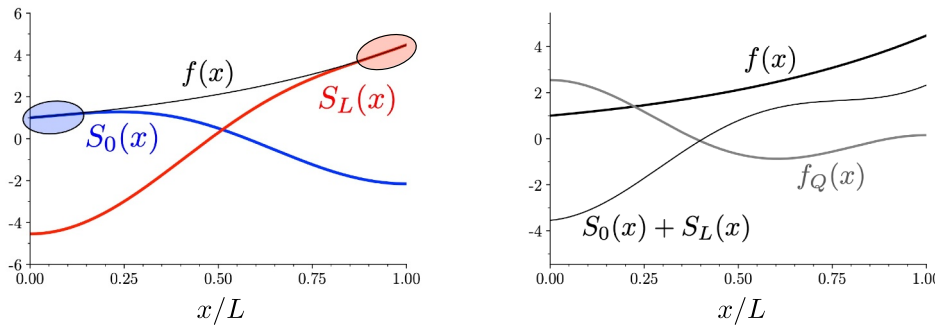

Figure 2. Explicit example for $f(x) = e^{\alpha x}$ using four term series expansions with $Q = 7$. Left: The function $f(x)$ and the two series S_0 and S_L shown separately. The expansion coefficients are computed by requiring S_0 to match f near $x = 0$ and S_L to match f near $x = L$ Right: The functions $f, S_0 + S_L$ and $f_Q = f - S_0 - S_L$. f_Q is offset vertically by -2 for clarity. The derivatives of f_Q vanish at the end points. The shaded regions in the left panel represent the regions where Bernoulli polynomials are constrained to match the function.

Because none of the fields u, v, w, $\mathbf{u}_*$ or ψ satisfy periodicity or symmetry constraints at the computational boundaries, the Bernoulli-Cosine scheme described here is needed to perform the spatial differentiation implied by the gradient and divergence operators in Algorithm 1. The only exception is ϕ, which satisfies $\nabla\phi \cdot \hat{n} = 0$ at the boundaries by construction. Its gradient can be computed by standard term-by-term differentiation of its cosine series expansion.

4.3. An Explicit Example

The approach is illustrated with a simple example. Let the domain length $L = 1$ and take

$$f(x) = e^{\alpha x} \quad \text{with} \quad \alpha = 3/2. \tag{22}$$

Let the number of grid points $N = 257$ and take $Q = 7$. The large value of N was chosen as a reminder that Gibbs oscillations are not eliminated by over-resolving the characteristic scale of variability within the finite domain. Figure 2 shows the function $f(x)$ evaluated at discrete points in $0 \le x \le L$. The function is not periodic in this domain, and its derivatives do not vanish at the end points. Consequently, the $2L$-periodic even extension of $f(x)$ has discontinuous derivatives at $x = 0$ and $x = L$.

Equations 20 and 21 are solved for the expansion coefficients a_n and b_n and these are used to construct the two series $S_0(x)$ and $S_L(x)$, shown separately in Figure 2. $S_0(x)$ matches the function f near $x = 0$ and has zero derivative at $x = L$. Conversely, $S_L(x)$ matches the function f near $x = L$ and has zero derivative at $x = 0$. The sum of these two series is shown in the right panel. Its derivatives at $x = 0$ and $x = L$ match those of f. Subtracting these two series from $f(x)$ exposes the function $f_Q(x)$ which is even symmetric over $0 \le x \le L$ and therefore suitable for term-by-term differentiation of its cosine series using a fast cosine transform.

Figure 3 shows two estimates of $f'(x)$ together with a set of discrete exact values (black circles). The first estimate (blue) is obtained simply by ignoring the discontinuity associated with the even extension of f, multiplying the cosine expansion coefficients by the wavenumber k and inverse transformation. This estimate suffers from Gibbs oscillations that are quite severe near the end points, as expected. The Bernoulli-Cosine estimate is obtained semi-analytically by differentiating the two series expansions using Equations 17 and 19 and applying the standard cosine transform differentiation method to $f_Q(x)$. The result is shown in red. The magnitude of the error in the Bernoulli-Cosine estimate, normalized by the maximum magnitude of $f'(x)$, is also shown. The accuracy of the estimate is highest in the interior of the domain and decreases toward the boundaries where the derivatives match those of the Bernoulli series to about six digits of accuracy.

In general, low-order Bernoulli polynomials are smooth throughout the domain, while the function f_Q can vary significantly. The derivative estimates in the interior are thus essentially spectral. In this sense, we refer to the accuracy as near-spectral. The right panel of Figure 3 shows the magnitude of the Fourier coefficients for $f(x)$ and

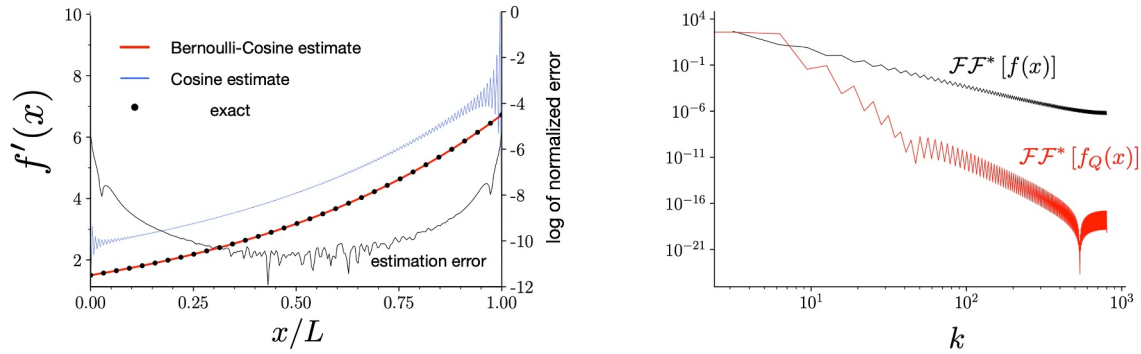

Figure 3. Explicit example for $f(x) = e^{\alpha x}$ using four term series expansions with $Q = 7$. Left: Estimates of $f'(x)$ (red) and normalized error $\hat{e}$ (thin black) using the Bernoulli-Cosine approach. The blue curve is the Gibbs contaminated estimate using the cosine transform alone, offset for clarity. Black circles are exact derivative values. The error scale is logarithmic. Right: Fourier spectra of $f(x)$ and $f_Q(x)$.

$f_Q(x)$ (both even extended). The spectrum of f_Q decays rapidly with wavenumber k, allowing its accurate differentiation using a cosine series, while the spectrum of the singular function f is almost flat near $k_{\max}$.

5. Example: Pressure Projection With Perfect Boundary Data

We now consider an example showing how the pressure projection algorithm of Section 3 can be used to compute flow in an open domain with specified boundary data that vary in both space and time. We demonstrate the approach by computing the flow associated with a large-scale internal wave mode propagating from left to right in an idealized, uniformly stratified ocean on an f-plane with rigid free-slip lids as shown in Figure 4. The exact solution in the linear, nondiffusive limit is known for this problem and is used to provide the initial and boundary data for an arbitrarily embedded subdomain. The demonstration consists of evolving the flow field within the subdomain using projection Algorithm 1.

As the internal wave mode propagates, the flow is neither symmetric nor periodic in the subdomain. The example is analogous to a one-way nesting application in which data are extracted from a large-scale, coarse-resolution parent simulation and provided as initial and boundary data for a nested child domain. In general, the boundaries of the child domain may partially coincide with those of the parent, but they are not required to do so. In particular, the embedded domain does not need to coincide with the surface or the bottom boundary, opening up the possibility of, for example, nested simulations entirely within the upper ocean thermocline.

The internal wave mode is monochromatic and has horizontal and vertical wavelengths much larger than the extents of the embedded subdomain as indicated in Figure 4. In the small-amplitude linear limit, the exact solution to the inviscid and nondiffusive Boussinesq equations of motion is periodic in time and used to test the projection

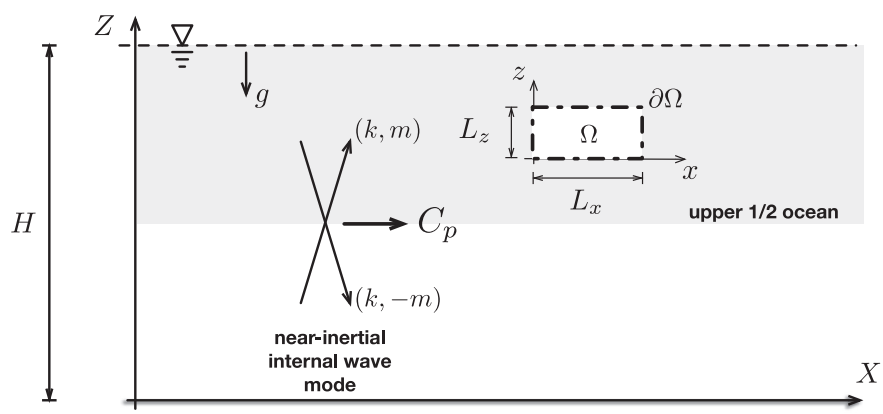

Figure 4. Schematic of parent and nested child domains. The rectangular child domain Ω is arbitrarily located within the parent domain and has flow-permeable boundaries $\partial\Omega$. A monochromatic internal wave mode propagates laterally at speed C_p with horizontal and vertical wavenumbers k and m.

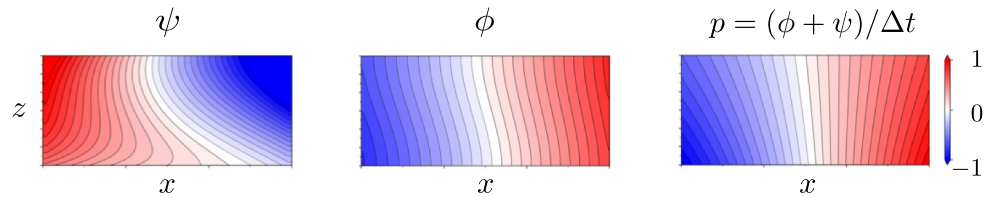

Figure 5. Snapshot of the pressure decomposition. Left: $\psi(x, z)$, middle: $\phi(x, z)$, and right: $p = (\phi + \psi)/\Delta t$. All fields are normalized by the magnitude of their maximum absolute value. The normal derivative $\nabla\phi \cdot \hat{\mathbf{n}}$ vanishes at the boundaries while the corresponding gradients of ψ and p do not.

scheme over an integration time of one wave period. In this example, the vector **F** in Equation 1 contains accelerations due to buoyancy and rotation and a coupled scalar equation for buoyancy is added. The discrete time-stepping used to estimate the integral in Equation 3 is a fourth-order explicit Adams-Bashforth method. The discretization of both space and time was chosen to be sufficiently fine so that the discretization errors are negligible. This allows us to focus on the treatment of the time-dependent, inhomogeneous boundary terms using a computational algorithm based primarily on cosine transforms.

The important parameters for the projection scheme are Q, which determines the number of terms used in the Bernoulli series, and the parameters that define the computation of ψ. In this example, we take $Q = 9$, which means that the Bernoulli series contains 5 terms. We set the diffusivity parameters in Equation 12 for ψ to be $\gamma h^2/\Delta\tau$, with h equal to the grid spacing Δx and Δz in the x and z directions, respectively. The parameter γ is adjusted to maintain stability of the pseudo-time (τ) integration. Its maximum value depends on the spatial discretization order and the time-stepping method. Here we use second-order spatial differences, an explicit Euler method for pseudo-time τ, and set $\gamma = 0.175$. During the first time step of the flow evolution, we integrate 50 pseudo-time steps with the initial condition for ψ set to zero. For all other steps, we start with the value of ψ from the previous t time step and integrate six pseudo-time steps in τ to incorporate updated boundary data. With this strategy, the nonzero values of ψ penetrate progressively deeper into the domain as the simulation evolves.

Figure 5 shows the decomposition of the pressure defined in Equation 8 after integrating in time for one wave period. ψ is computed by time-stepping Equation 12 and ϕ is computed by direct spectral (cosine) inversion. As ϕ is constructed as a sum of separable basis functions satisfying no-gradient conditions at the boundaries, it exactly satisfies homogeneous Neumann conditions. While this decomposition is computationally convenient, the physical quantity of interest is the pressure $p = (\phi + \psi)/\Delta t$ shown in the right panel. Note that p does not satisfy any symmetries at the computational boundary.

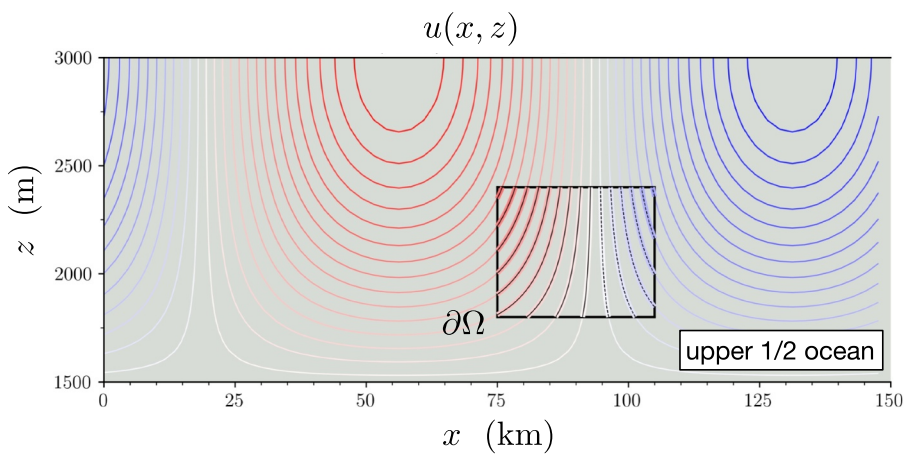

Figure 6. Snapshot of the upper half of an idealized ocean. A single internal wave mode propagates from left to right at a fixed phase speed, passing through a subdomain with a flow-permeable boundary $\partial\Omega$. Contours of the x component of the velocity u are shown. The exact solution is shown in colored contours that are thickened in the nested subdomain where the flow is computed from the initial conditions and the time-dependent boundary data. The computed solution in the subdomain is shown in thin black contour lines (negative values dashed) that coincide almost exactly with the exact solution.

Figure 6 shows the embedded computational domain Ω within the upper half of an idealized ocean where a monochromatic internal wave mode propagates at fixed phase speed from left to right. The exact solution for the x component of the velocity is shown with color contour lines after one wave period of integration. The computed solution is superimposed in black on the exact solution within the subdomain. A temporal animation (not shown) of frames plotted similarly shows the coincidence of computed and exact solutions throughout the wave evolution. In this example, the propagating mode has spatial wavelengths much larger than the size of the embedded domain. The example has been repeated in the limit where the wave scales are shorter than the sides of the subdomain with similarly good agreement. The example illustrates that when the boundary data is perfectly known and the integration for $\mathbf{u}_*$ is accurate, the corner smoothness condition in Equation 14 is satisfied and Algorithm 1 works well.

A practical concern when using this algorithm for a one-way nesting of a refined resolution subdomain within a coarse, large-domain simulation is that the corner vorticity ζ^{n+1} implicit in the prescribed boundary conditions is exact. It is approximated with an accuracy that is necessarily lower than that for ζ_*, which is computed on the more finely resolved subdomain. In principle, there will always be a discrepancy between ζ_* and ζ^{n+1} because ζ_* generally has variability on scales smaller than those represented in the parent simulation. Corner smoothness constraints cannot be assumed to be exactly satisfied.

A related issue concerns the tangential flow components at the boundaries. The projection step yields these values at the boundaries without the need to appeal to known or prescribed boundary values. The projection scheme itself only requires knowledge of the normal flow components on $\partial\Omega$. On the other hand, if we know the values of the tangential components at the boundaries, either exactly as in this example or approximately from a lower resolution parent simulation, we can compare the known values with those obtained from the projection step. As with the corner smoothness, when the known values are exact, and in the limit that the computed values are also exact, the boundary values match. When the prescribed boundary information is deficient, that is, it lacks small-scale variability present in the projected fields, a mismatch will occur. In this example problem, neither corner discontinuities nor tangential flow mismatches at the boundaries present any difficulties.

However, in nesting applications, the subdomain is typically more finely resolved, and corner and boundary mismatches may be expected to present some difficulties. To this end, we consider a doubly nested simulation of an idealized upper ocean eddy subjected to variable wind forcing and test a modification of Algorithm 1 designed to mitigate these issues. The algorithmic modification can be considered an ad hoc procedure that yields acceptable approximate solutions to a slightly ill-posed problem. The problem is ill-posed because relatively low-resolution, approximate information is imposed at open boundaries instead of the exact but unknown boundary values.

6. Example: Pressure Projection With Coarse Boundary Data

The nesting approach is now used to simulate the interaction between a depth-dependent anticyclonic eddy in cyclogeostrophic balance and a spatially uniform, temporally variable stochastic wind field. This interaction is known to result in excitation, downward propagation, and trapping of internal waves of near-inertial frequency within the eddy (e.g., Asselin et al., 2020; Kunze et al., 1995; Lelong et al., 2020). The simulation suite is designed to resolve the excitation and evolution of these waves as they propagate into the eddy core. Although detailed dynamics of the waves are beyond the scope of this article, some features of the problem are salient.

First, the problem is widely scale separated. A typical eddy radius is about 20 km but the radial decay of the associated flow field is rather slow and so, simulating such an eddy generally requires a domain with a lateral extent many times larger than the typical eddy scale. In addition, waves propagating outward from the eddy will be excited. These waves may reflect from computational boundaries and contaminate the eddy core in finite time. If long study times are desired, the size of the required domain increases. These undesired effects can be mitigated with large domains and sponge regions that damp outgoing and reflected waves. The primary physics of interest, however, occurs within the much smaller confines of the eddy itself.

The strategy is to compute the flow evolution first over a very large domain, then over a subdomain moderately larger than the eddy, and finally over a domain slightly larger than the eddy core confined to a limited depth range, as shown in Figure 7. We denote the three domains as the parent, child, and grandchild domains. At each nesting level, we refine the spatial grid to compute the flow with increasing resolution. The horizontal grid spacings

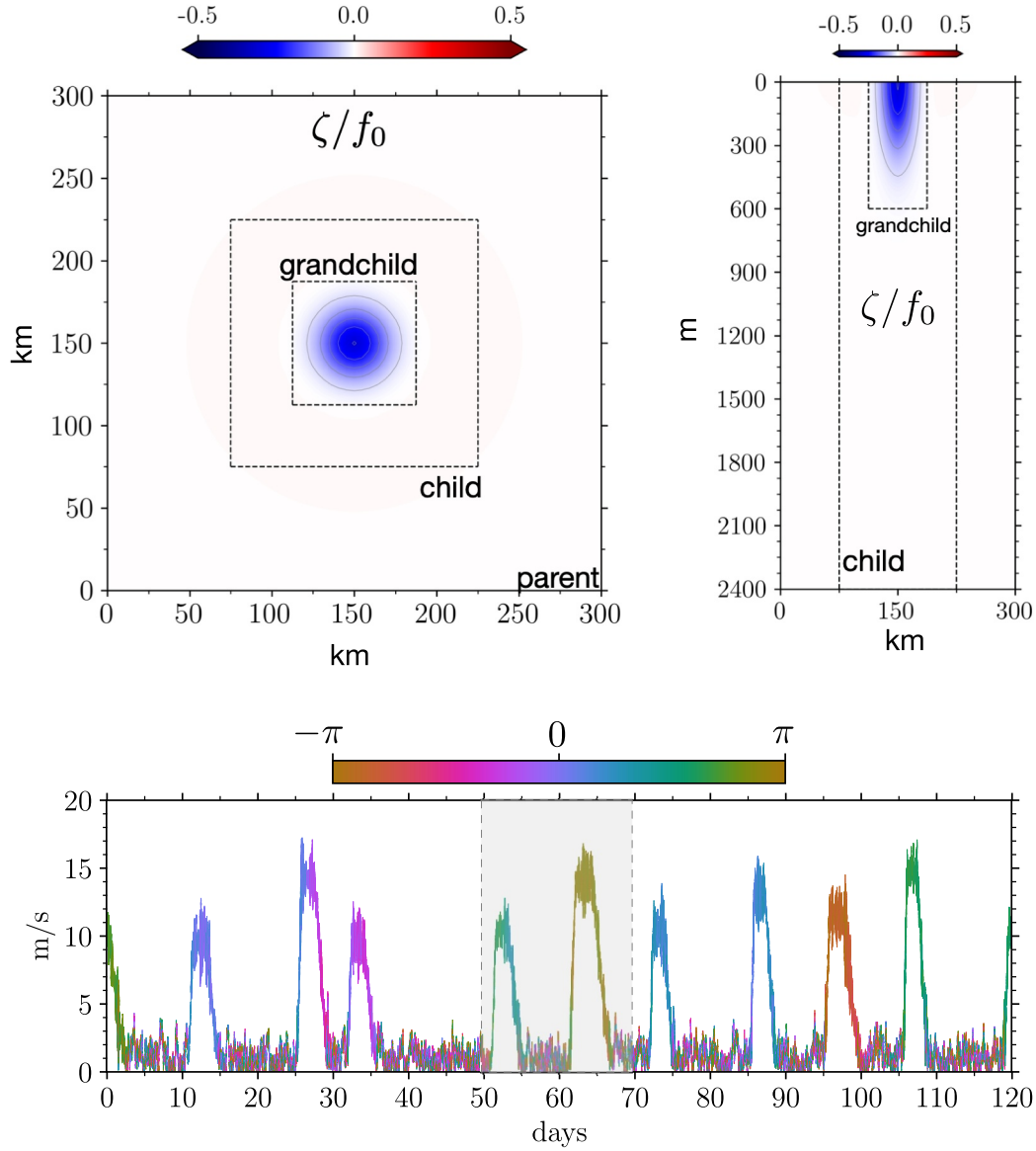

Figure 7. Schematic of the nesting strategy for the simulation of near inertial waves in an anticyclonic vortex perturbed by a stochastic surface wind field. The upper left panel shows a plan view snapshot of the initial vertical relative vorticity normalized by f_0 at the ocean surface over the parent domain. The right upper panel shows the same quantity in a vertical cut through the center of the eddy. The domains of the child and grandchild simulations are indicated by dashed lines. The lower panel shows the 10 m wind speed and direction used to derive a near-surface stress that forces the flow. The child and grandchild runs are made over the 20 days shaded period.

$\Delta x = \Delta y$ are 4.7, 1.6, and 0.59 km for the three domains, while the vertical spacings Δz are 9.4, 4.7 and 2.3 m. The parent simulation was run for a total of 120 days with the computed solutions saved on the child domain boundaries between days 50 and 70, an interval of 32.8 inertial periods $2\pi/f_0$, where f_0 is the Coriolis parameter. The boundary values and the full three-dimensional solutions at 50 days were interpolated to the finer child domain resolution and used as input for the child run. The initial and boundary values for the grandchild domain were then saved and interpolated to even finer resolution.

6.1. Buoyancy and Hydrostatic Pressure

At this point it is convenient to separate the hydrostatic component of the perturbation pressure field p_h and treat it explicitly in the time stepping. To this end, we write

$$p = \bar{p}(z) + p_h(\mathbf{x},t) + p'(\mathbf{x},t) \tag{23}$$

and

$$\rho = \bar{\rho}(z) + \rho'(\mathbf{x},t), \tag{24}$$

where $\bar{\rho}$ is an ambient density profile representative of the problem at hand. The corresponding background pressure $\bar{p}$ satisfies

$$\frac{\partial}{\partial z}\bar{p} = -g\bar{\rho} \tag{25}$$

and the perturbation hydrostatic pressure p_h satisfies

$$\frac{\partial}{\partial z}p_h = -g\rho'. \tag{26}$$

The projection algorithm described in Sections 2 and 3 is modified formally by replacing $\mathbf{F}$ with $\mathbf{F} - \nabla p_h$ and p with p'. With these small changes, the algorithmic steps are identical.

The parent simulation in the nesting strategy is computed over a domain that is large compared to the prescribed ambient vortex. Although we expect waves to radiate outward from the vortex, we do not know exact radiation conditions to apply at the lateral boundaries. An approximate radiation condition is to simply impose that

$$\nabla p' \cdot \hat{\mathbf{n}} = 0 \tag{27}$$

at the west, east, south and north lateral boundaries. This works well here because the outward radiating waves are near-inertial and therefore approximately satisfy the hydrostatic equations of motion. The time-dependent spatial gradients of p_h at the boundary due to radiating waves are not constrained but handled cleanly by the Bernoulli-Cosine treatment, while only the small correction is constrained by the imposed boundary condition. We have compared this approach to one in which a sponge region is imposed to damp outgoing and reflected waves and found it to work equally well or better.

6.2. The Approximate Pressure Projection Scheme

In the child and grandchild runs, the saved boundary data used are known with a coarser spatial and temporal resolution than the simulation itself. Therefore, we expect that the corner smoothness condition will not be exactly satisfied and the diffusion Equation 12 for ψ will be ill-behaved as a result.

To avoid this problem, we introduce an approximation to the projection algorithm. At the lateral boundaries, we impose the boundary conditions extracted from the parent run during the t^n to t^{n+1} time step, that is, we require that

$$u_* = u^{n+1}_{\text{saved}},\ v_* = v^{n+1}_{\text{saved}} \text{ and } \rho' = \rho'^{\,n+1}_{\text{saved}}. \tag{28}$$

These boundary conditions can be incorporated into the treatment of the diffusion terms.

Given that u_* and v_* satisfy prescribed boundary conditions, the pressure projection scheme only needs to incorporate boundary conditions for w at the top and bottom of the subdomain. Assuming exact boundary values, Equation 11 implies that the auxiliary field ψ should satisfy

$$\frac{\partial}{\partial z}\psi = w_* - w^{n+1}_{\text{saved}} \tag{29}$$

at the top and bottom boundary planes,

$$\frac{\partial}{\partial x}\psi = 0 \tag{30}$$

on the east and west lateral boundaries, and

$$\frac{\partial}{\partial y}\psi = 0 \tag{31}$$

on the north and south boundaries. The difficulty is that the boundary information is only approximate; it lacks the fine scales of computed solution in the interior and the corner smoothness condition Equation 14 will not be satisfied. To overcome this intrinsic problem, we slightly relax the constraints on ψ at the lateral boundaries.

Let ψ be defined explicitly as

$$\psi(x,y,z) = a(x,y)\, e^{-(L_z - z)/\gamma} + b(x,y)\, e^{-z/\gamma} \tag{32}$$

where γ is a decay scale of order Δz, the vertical grid spacing. The coefficients a and b satisfy the following.

$$a(x,y) = \gamma\big(w_*(x,y,L_z) - w^{n+1}_{\text{saved}}(x,y,L_z) \big) \tag{33}$$

and

$$b(x,y) = -\gamma\big(w_*(x,y,0) - w^{n+1}_{\text{saved}}(x,y,0) \big). \tag{34}$$

The projection step is

$$\mathbf{u}^{n+1} = \mathbf{u}_* - \nabla\psi - \nabla\phi \tag{35}$$

and at the boundaries the solution satisfies

$$\mathbf{u}^{n+1} \cdot \hat{\mathbf{n}} = (\mathbf{u}_* - \nabla\psi) \cdot \hat{\mathbf{n}} \tag{36}$$

because the normal derivatives of ϕ are zero at the boundaries. ψ is defined so that the boundary conditions for w are satisfied. The derivatives ψ_x and ψ_y are negligible everywhere on the lateral boundaries except within thin boundary layers near the upper and lower boundary planes. In these regions, ψ_x and ψ_y are nonzero, and the boundary conditions for the horizontal velocity components are slightly perturbed from the values saved from the parent simulation and imposed on u_* and v_*. These perturbations are small and can be thought of as the perturbations required to ensure that the imposed boundary conditions are such that the corner smoothness condition is satisfied. The modified projection scheme is summarized in Projection Algorithm 2.

Algorithm 2. Pressure Projection for Nested Domains With Coarse Boundary Data

(a) Compute $\mathbf{u}_* = \mathbf{u}^n + \int_{t_n}^{t_{n+1}} \mathbf{F}\, dt$ cf. Equations 1–3 with $-\nabla$ph absorbed into F and with the boundary conditions $u_* = u^{n+1}_{\text{saved}}$ and $v_* = v^{n+1}_{\text{saved}}$ imposed at the lateral boundaries.
(b) Calculate the mismatch $\left(w_* - w^{n+1}_{\text{saved}}\right)$ at the upper and lower boundaries.
(c) Construct ψ via Equation 32.
(d) Calculate $\nabla\psi$ and compute $\widetilde{\mathbf{u}}_* = \mathbf{u}_* - \nabla\psi$.
(e) Calculate $\nabla\cdot\widetilde{\mathbf{u}}_*$ and solve Equation 9 for ϕ using fast cosine transforms.
(f) Calculate $\nabla\phi$ and project: $\mathbf{u}^{n+1} = \widetilde{\mathbf{u}}_* - \nabla\phi$.
(i) Save the pressure $p = \frac{1}{\Delta t}(\phi + \psi)$.

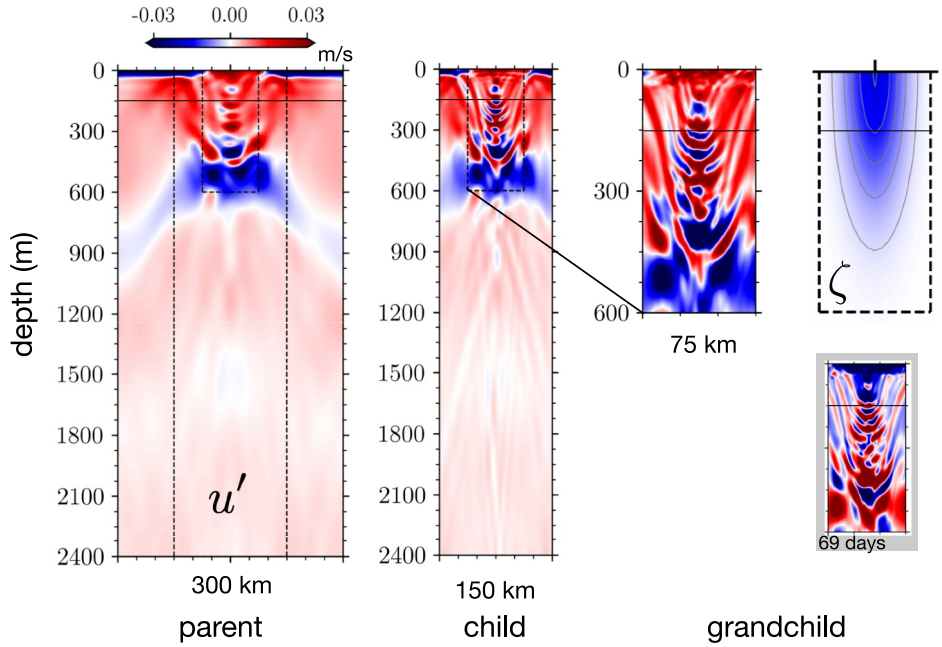

Figure 8. Snapshots of u' in a vertical slice through the eddy center in the parent, child and grandchild domains 65 days after initiation of the wind stress. Also shown are the ambient vorticity ζ for reference, the 150 m depth level (thin lines) corresponding to Figure 9, and u' in the grandchild domain four days later.

6.3. Wave Trapping in a Vortex via Nesting With Algorithm 2

For the simulation of an idealized wind-driven vortex, the velocity field $\mathbf{u}$ is decomposed into the sum of the balanced vortex flow U_V, the wind-driven flow in the mixed layer U_τ, and the perturbation velocity $\mathbf{u}'$. In each of the simulations, the hydrostatic pressure gradient $-\nabla p_h$ is included in the explicit forcing term $\mathbf{F}$. In the parent simulation, the approximate radiation condition $\nabla p' \cdot \hat{\mathbf{n}} = 0$ at the lateral boundaries is imposed. For the child and grandchild runs, time-dependent boundary values saved from the parent and child runs, respectively, are imposed via Algorithm 2.

Figure 8 shows the perturbation horizontal component u' in a vertical slice through the center of the vortex core 55 days after the initiation of the wind forcing. The wind-driven flow in the near-surface layer deforms the vortex buoyancy field at near-inertial frequency and excites downward propagating near-inertial internal gravity waves. Waves at slightly subinertial frequency are trapped within the anticyclonic eddy and become evanescent beyond the eddy core. Animation of sequential images shows upward propagation of the phase, the signature of downward energy propagation. The signals pass through the various computational boundaries without distortion. The same quantity from the grandchild run four days later is shown in the image with gray background. The finer detail captured in the trapped wave structure shows that, with increasing time, energy is cascaded toward smaller scales, and the newly resolvable spatial scales in the nested simulations become energized.

Figure 9 shows u' in plan view at 150 m depth from the doubly nested simulations. At the eddy center in the parent simulation u' is small but slightly positive while it is distinctly negative in the child and grandchild runs. From Figure 8 we see that this is a result of more finely resolving the vertical structure of the trapped waves. Animation of sequential images shows oscillations of the structure in the core and outward, spiraling radiation of near-inertial bands that pass through the computational boundaries smoothly.

7. Example: Pressure Projection With Coarse Boundary Data From an Ocean General Circulation Model

In the previous nesting examples, the boundary values applied at open boundaries were obtained from lower resolution integration of the same set of governing equations. Compared to exactly known boundary values consistent with the solutions sought, this introduced a compatibility issue and required the use of the modified

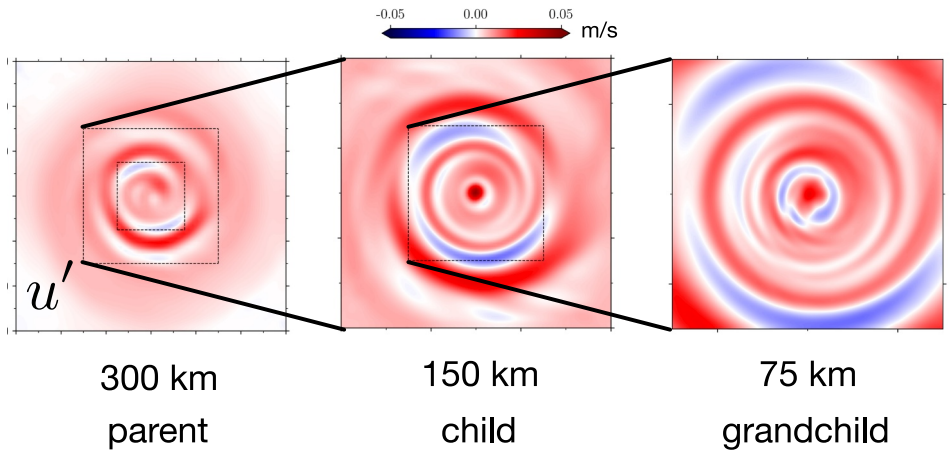

Figure 9. Snapshots of u' in plan view at 150 m depth computed in the parent, child and grandchild domains 65 days after initiation of the wind stress.

Projection Algorithm 2. Here we go a step further and explore nesting when the boundary values are obtained from a lower-resolution integration of a simplified form of the governing equations with parameterizations or closure schemes that we do not attempt to directly reproduce.

The overall nesting strategy consists of three levels, a basin scale 1/12° resolution simulation of the Mediterranean Sea and the eastern Atlantic Ocean, a 1/64° self-nesting focusing on the Gulf of Lion (Madec & Team, 2022) (the parent model here), and a finite depth nonhydrostatic simulation at 500 m resolution centered on a semi-persistent density front using **flow_solve** (the child model) as shown in Figure 10.

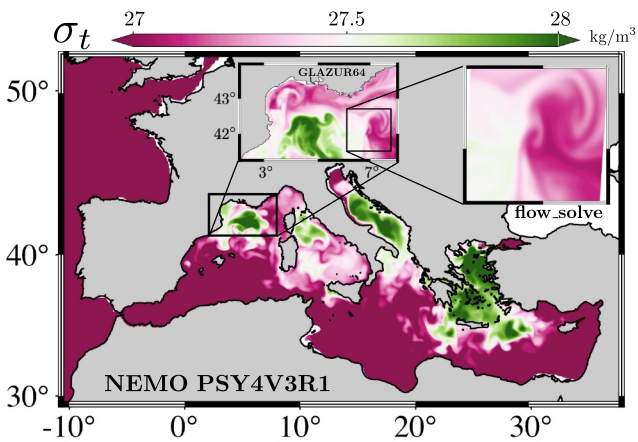

Figure 10. Map of σ_t showing the Gulf of Lion nesting strategy. The outermost simulation models the Mediterranean Sea and the eastern Atlantic Ocean at 1/12° resolution and 50 vertical z-levels using NEMO. The GL64 simulation is a NEMO self-nesting focusing on the Gulf of Lion at 1/64° × 1/64° resolution (about a 1.2 km mesh on a Mercator isotropic projection) with 130 z levels. The nonhydrostatic **flow_solve** domain is a 128 km by 128 km portion of the Gulf roughly centered on a semipersistent density front. The **flow_solve** domain extends from a few meters below the free surface to a depth of 2,052 m, shallower than the approximately 2,700 m water depth of the region. Time (1 November 2009 at 12p.m. GMT) and depth (24 m, within the mixed layer) are the same for all panels.

7.1. The Parent Model

For the parent simulation, we use the 1/64° × 1/64° GLAZUR64 configuration (hereafter denoted GL64) of the NEMO Ocean General Circulation Model (Madec & Team, 2022) that focuses on the Gulf of Lion in the northwest Mediterranean during November 2009.

The vertical grid spacing Δz is 1 m above the upper 30 m to allow the resolution of the sharp pycnocline and decreases exponentially below to $\Delta z = 30$ m near the seabed with a total of 130 z-levels. GL64 is forced at the surface by 0.1° 3-hr atmospheric data (wind, heat fluxes, evaporation) from the Météo-France model ARPEGE with ad hoc data assimilation (Bouyssel et al., 2022), which is sufficient to generate inertial oscillations at the surface. GL64 has two open boundaries (south and east, see Figure 10) where daily heat, salinity and momentum transfers were obtained from the global operational NEMO configuration PSY4V3R1, a reference product for realistic ocean forecasting that is freely available through the Copernicus Marine Service (https://marine.copernicus.eu/). PSY4V3R1 covers the global ocean, including the Mediterranean Sea, with 1/12° resolution and 50 vertical z-levels, clustered at a spacing of 1 m at the surface and 450 m at the bottom. This operational product uses a sequential data assimilation system (SAM2V1) based on the Kalman filter with SEEK formulation and bias correction (3D-Var) with incremental analysis update (IAU) (Ourmières et al., 2006). The data assimilated are satellite sea surface temperature and sea level anomaly, and in situ temperature and salinity profiles.

The GL64 configuration employs parameterizations from the suite implemented in NEMO and has been used for realistic marine applications such as jellyfish stranding forecasting (Berline et al., 2013), floating litter transport simulations (Ourmières et al., 2018), as well as studies of realistic mesoscale dynamics such as boundary current intrusions (Barrier et al., 2016) and eddy generation (Guihou et al., 2013).

7.2. The Child Model

The goal of the nested simulations is to take advantage of the regional realism of the large-scale circulation and forcing, the surface vorticity field, and the sophisticated nature of the representation of surface forcing and mixed layer dynamics, but to "replace" the internal gravity waves that are generated to study their dynamics with higher fidelity as they propagate into the ocean interior.

The nested **flow_solve** domain is embedded in the GL64 domain roughly centered on a persistent density front that is continuously perturbed by mixed-layer inertial currents. The domain size is 128 km × 128 km by 2,048 m in the vertical. The upper boundary is located four m below the mean free-surface position in the GL64 model, while the lower boundary, at a depth of 2,052 m, is well above the ocean bottom, which varies in depth from about 2,600 to 2,750 m in this region. Using the velocity and density values from the GL64 run on these boundaries allows us to avoid directly modeling the free surface and the flux of atmospheric properties across it, implementing a coupled mixed-layer model, and explicit treatment of variable bathymetry while still leveraging the representation of these effects within NEMO.

The nested simulation is run for a 4-week period beginning on 5 November 2009. During this time interval, the GL64 solutions exhibit variability on slow time scales much longer than the inertial frequency and at fast near-inertial scales. For example, there is a slow overall cooling of the surface mixed layer in addition to energetic, wind-driven near-inertial currents. Similarly, slow large-scale currents advect large-scale density variability through the domain at depth, while hydrostatic, low vertical mode internal waves produce fast, near-inertial variability.

The modeling strategy adopted relies on three characteristics of the GL64 solutions:

1. The horizontal gradients of the free surface height induce accelerations that are comparable to those arising from the Coriolis terms.
2. The depth of the base of the surface mixed layer varies between about 25 and 50 m and on time scales of less than a day. Within the surface mixed layer, the modeled vertical diffusivity is very large, and both momentum and thermodynamic tracers are well mixed.
3. On a mid-depth horizontal cross section, the GL64 solutions for w are the signatures of horizontally localized, low-vertical-mode internal waves that are solutions to the hydrostatic equations of motion but are not solutions to the nonhydrostatic equations. Under nonhydrostatic dynamics, these wave modes rapidly disperse. The ubiquitous and persistent presence of these modes in the GL64 solutions contaminates the saved boundary data at near-inertial frequencies.

7.3. Modeling the Upper flow_solve Boundary

The GL64 simulation has a free surface characterized by displacements $\eta(x, y, t)$ and a corresponding variable surface pressure field $p_s = g\eta$. Gradients of η induce barotropic horizontal acceleration $g\nabla\eta$. Focusing on the region of the density front in the **flow_solve** domain, the free surface drops about 6 cm in 50 km corresponding to an induced acceleration of about 10^{-5} m/s^2. Taking the Coriolis parameter $f_0 \approx 10^{-4}$ s^{-1} and a characteristic horizontal speed of 0.1 m/s yields the same magnitude for the Coriolis terms. To include this forcing in the **flow_solve** run, we position the upper surface of the domain at 4 m depth. Saved values $\eta(x, y, t)$ are interpolated to the **flow_solve** grid and used to calculate the surface pressure $p_s(x, y, t)$ at each time step. p_s is then added as an integration constant during the computation of the hydrostatic pressure p_h using Equation 26. The normal flow speed $w(x, y, t)$ at 4 m depth from the GL64 simulation is saved and imposed on the **flow_solve** solutions.

7.4. Modeling the Mixed Layer

The vertical grid spacing in GL64 is approximately 1 m in the mixed layer but increases rapidly near and below the mixed layer base, increasing from about 5 m at 50 m depth to about 30 m at and below 300 m. Approximately 30% of the GL64 grid, and thus the computational effort, is focused on the surface mixed layer. The mixed layer is

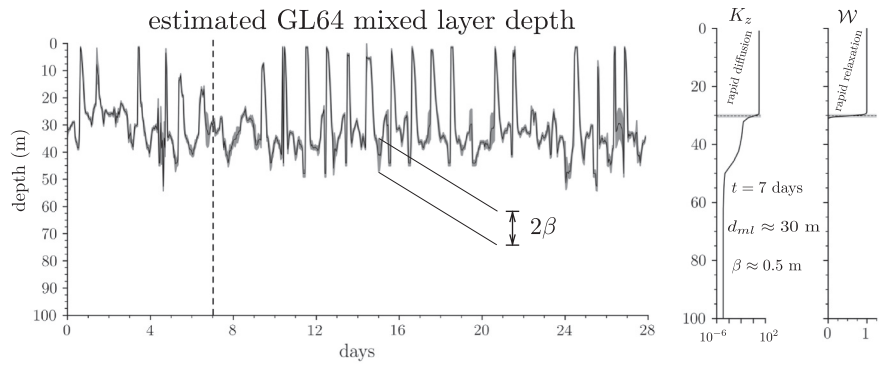

Figure 11. Left: Estimated depth of the base of the mixed layer in GL64 as a function of time. The transition layer of thickness 2β is shaded. Middle: The logarithm of the modeled diffusivity K_z from the GL64 TKE closure scheme at the start of day 7. Mixed layer values are 10 m^2/s. Right: The corresponding relaxation window $\mathcal{W}(z)$ used in the **flow_solve** simulation.

driven by assimilated atmospheric data imposed as stress and flux boundary conditions and distributed vertically through the NEMO Turbulent Kinetic Energy (TKE) closure scheme. Because our primary focus is on internal waves at depth, we chose to exploit the GL64 mixed-layer solutions without directly reproducing the surface boundary conditions or the TKE closure scheme. To better resolve this physics, the **flow_solve** simulation incorporates finer spatial resolution beneath the mixed layer, higher-order numerics (near-spectral vs. second-order finite difference), and the richer nonhydrostatic equations of motion.

To model the mixed layer itself, we take advantage of the fact that the TKE closure in GL64 produces mixed layer diffusivities K_z on the order of 10 m^2/s and that these are essentially uniform throughout the mixed layer. Taking a nominal thickness of the mixed layer $h_{ml} = 40$ m, this implies a diffusive time scale of $T_{\text{mix}} = h_{ml}^2/K_z$ of 160 s and near homogenization of the mixed layer on a time scale of a few T_{mix}. We take advantage of these characteristics by extracting time-dependent vertical profiles of K_z from GL64. For each profile we estimate the depth of the base of the mixed layer z_{ml} and the thickness of the transition layer β over which the diffusivity rapidly decays, as illustrated in Figure 11.

At each time step in the **flow_solve** run, z_{ml} and β are used to define a relaxation window $\mathcal{W}$ as

$$\mathcal{W}(z) = \frac{1}{2}(1 + \tanh\{(z - z_{ml})/\beta\}.) \tag{37}$$

$\mathcal{W}$ is equal to one within the estimated mixed layer and decays to zero below over the scale β. The equations of motion are then modified by adding rapid relaxation terms for the horizontal components of velocity and density anomaly ρ'. For example, the relaxation term on the right-hand side of the x momentum equation is

$$-\frac{1}{T_{\text{relax}}}\mathcal{W}(z)(u(x,y,z,t) - u_{GL64}(x,y,z_0,t)) \tag{38}$$

where $u_{GL64}(x,y,z_0,t)$ is the GL64 value saved at the upper open boundary of the **flow_solve** domain and T_{relax} is a relaxation time scale set to 240 s. As the GL64 mixed layer deepens and shallows over time scales faster than but comparable to a day, the **flow_solve** solutions are constrained to produce a nearly well mixed layer with matching thickness and properties.

Figure 12 shows a representative snapshot of the **flow_solve** mixed layer flow embedded within the GL64 flow field at the same depth and time. The simple relaxation scheme allows us to avoid reproducing the surface forcing and closure schemes of NEMO while still enabling temporal and geographical realism of the computed mixed layer flow and thus "realistic" generation and downward propagation of wind-driven near inertial internal waves.

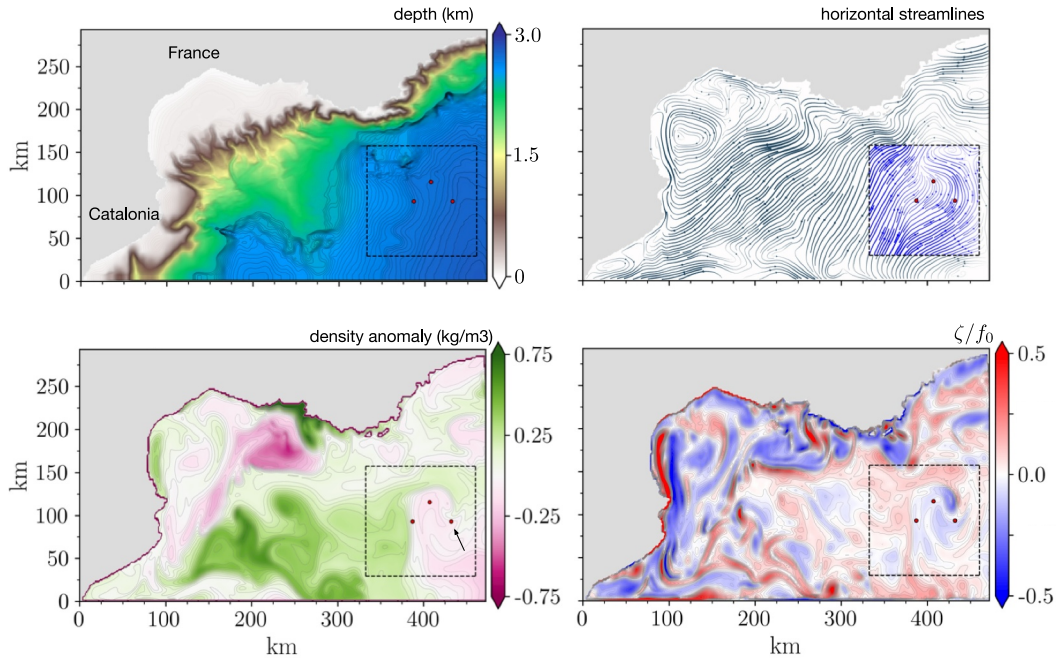

Figure 12. GL64 and embedded **flow_solve** domains. The upper left panel shows the bathymetry of the Gulf of Lion, the **flow_solve** subdomain (dashed lines), and the location of three synthetic moorings (red dots). The other panels show snapshots within the surface mixed layer (24 m depth) from the **flow_solve** simulation within its subdomain and the outer GL64 solution 7 days after the start of the nested run. The density and streamlines reveal the overall flow patterns in the basin. The instantaneous density anomaly and vorticity fields show fronts and filaments with spatial scales on the order of 10 km. The location of the mooring used in Figure 13 is indicated by an arrow.

7.5. Modeling the Background Diffusivity

Diffusion in **flow_solve** is modeled using separable, diffusion operators $\mathcal{D}$ of the form

$$\mathcal{D}(u) = (-1)^{p-1}\,\kappa_*\,\frac{\partial^{2p}}{\partial x^{2p}}\,u \tag{39}$$

where p is the half order of the operator and κ_* is a coefficient with dimensions $\mathrm{m}^{2p}\,\mathrm{s}^{-1}$ (e.g., Winters, 2016). These operators are applied to both the momentum and the scalars in each coordinate direction. With increasing $p > 1$, the action of these operators is increasingly confined toward the smallest resolvable scales. The time scale T_{diff} over which variability at the grid scale is removed is given by

$$T_{\mathrm{diff}} = k_{\mathrm{max}}^{-2p}\,\kappa_*^{-1} \quad \text{where} \quad k_{\mathrm{max}} = \frac{\pi}{\Delta} \tag{40}$$

and Δ is the grid spacing.

The parameter values were set to $p = 3, 4$ and $T_{\mathrm{diff}} = 20$ s and 1 hr in the horizontal and vertical directions, respectively. The very weak vertical diffusivity was chosen to approximately match the explicit vertical diffusivity in GL64 where, at depth, $K_z = \mathrm{O}(10^{-5})\ \mathrm{m}^2\,\mathrm{s}^{-1}$ and the characteristic diffusion distance over a day is about 1 m. The characteristic horizontal diffusion scale over a day $(\kappa_*\,T_{\mathrm{day}})^{1/6} \approx 640$ m, comparable to the grid spacing Δx.

7.6. Modeling the Lower Boundary

To highlight near inertial waves, the slowly evolving flow, defined by a third order Butterworth low-pass filter with a cut-off frequency of $f/2$, is subtracted from the horizontal velocity components at each depth. The

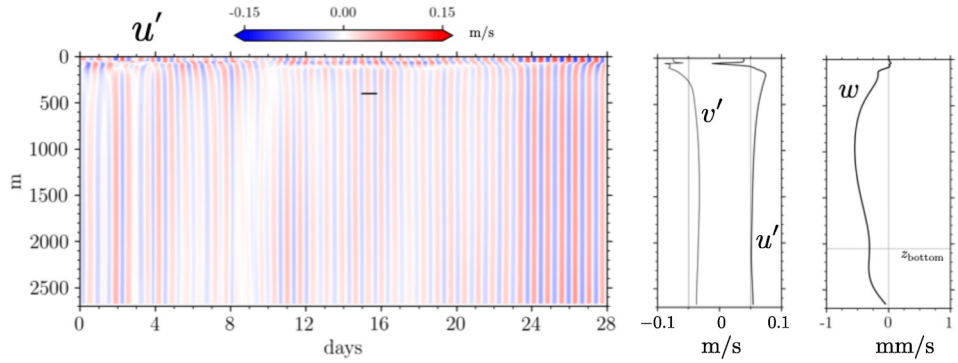

Figure 13. Left panel: Depth-time plot of the zonal component of the fast velocity from the GL64 run at the mooring position indicated in Figure 12. The horizontal line at 400 m depth spans an inertial period. Middle panel: Instantaneous profiles of the zonal and meridional flow components at $t = 7$ days, offset for clarity. Right panel: The corresponding w profile. The position of the bottom boundary of the **flow_solve** domain is indicated.

resulting "fast" components are denoted u' and v'. Figure 13 shows the features of the near-inertial internal waves resolved in the GL64 simulation at the mooring location indicated in Figure 12. The waves are fairly energetic, with horizontal speeds often exceeding 0.1 m/s. They have frequencies close to the inertial frequency f and little vertical structure beneath the mixed layer. Looking at the corresponding values saved on one of the vertical sections used as boundary values (not shown), we see that these waves take the form of spatially localized columnar structures with horizontal wavelengths of 10–20 km. The structures are coherent in time and can be visually tracked in animations of the boundary data. These waves are the responses to convergence and divergence of the mixed-layer flow that perturbs both the free surface and the pycnocline.

Under hydrostatic dynamics, these wave structures do not significantly disperse. The range of horizontal wavenumbers needed to form a localized structure propagate with the same horizontal speed. However, these waves disperse significantly more when allowed to evolve under nonhydrostatic dynamics. This can be understood by noting that the aspect ratio of these features kH, where k is the horizontal wavenumber and H is the approximate ocean depth, is order one. The hydrostatic approximation formally assumes a vanishing aspect ratio but is generally invoked whenever the ratio is expected to be asymptotically small. In GL64, fast motions at small horizontal scale in the mixed layer force a nearly non-dispersive small-scale, low-mode near-inertial response. In contrast, the nested **flow_solve** runs undergo an initial transient phase characterized by the rapid dispersion of the spatially coherent near-inertial waves present in the initial conditions extracted from GL64.

This introduces a fundamental problem at the lower open boundary of the **flow_solve** domain. The GL64 boundary values of w are primarily associated with coherent wave modes that do not exist in **flow_solve** after the short initial transient phase. For illustration, see the value of w at the bottom of the **flow_solve** domain in Figure 13.

Mathematically, we are free to impose whatever values $w(x, y, t)$ we want using Projection Algorithm 2. For example, the values imposed in the grandchild run just beneath the anticyclonic vortex (Section 6.3, Figure 9) were low-resolution versions of the values needed to allow seamless propagation of the downward propagating waves through the boundary. However, in GL64 nesting, the saved boundary values w have amplitudes and phases of waves that are dynamically inconsistent with the dispersive waves computed in the interior. If imposed, these boundary conditions would act both to excite upward-propagating waves and to reflect downward-propagating waves.

Although we are obligated to prescribe $w(x, y, t)$ at the lower boundary, we must admit that we do not know the values that should be imposed. The correct normal flow boundary values are those that allow outgoing signals to radiate without reflection, while permitting incoming signals to enter the domain cleanly. In the previous examples, the required boundary values were known analytically (Section 5) or were reasonably approximated by saved values from coarser resolution simulations of the same governing equations (Section 6).

In the present case, the hydrostatic parent model does not provide these values, particularly at the bottom open boundary. For this reason, we place the boundary much deeper than about 500 m, the depth of our main region of

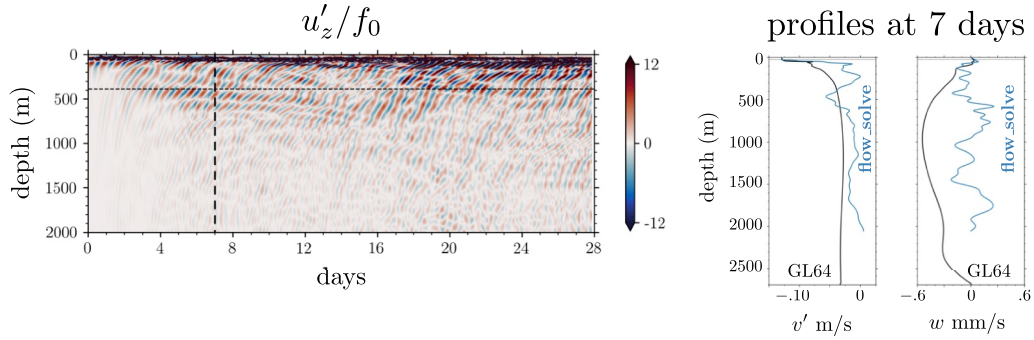

Figure 14. Left panel: Depth-time plot of the zonal shear u_z' normalized by f_0. Middle: Profiles of meridional velocity v' at 7 days from GL64 and **flow_solve** simulations. Right: Corresponding profiles of w.

interest, and impose that $w = 0$ there. The consequences of this choice are that downward propagating waves are reflected from the lower boundary and the lowest modes excited will have reduced vertical scale (e.g. $H = 2{,}052$ m rather than ≈2,700 m).

An alternative approach might be to introduce a sponge layer near the bottom computational boundary. This would reduce or eliminate reflection, but, as a consequence, prevent the formation of low vertical modes. Because low vertical modes, which are a superposition of downward-propagating waves and their upward reflections, are a prominent feature of the GL64 solutions, we choose to allow the possibility of their formation by prescribing the reflective boundary condition $w = 0$. In effect, we compute the wave response to the GL64 mixed layer forcing in a shallower ocean. From an asymptotic perspective, the effective depth of the ocean remains much larger than the thickness of the mixed layer, that is, for $h_{ml} \approx 40$ m, the ratio h_{ml}/H only increases from about 0.015 to 0.02 when H is reduced.

7.7. Near Inertial Waves Beneath the Mixed Layer

Figure 13 shows that there is little vertical structure and consequently little near-inertial shear beneath the mixed layer in the GL64 simulation. Despite an energetic mixed-layer flow with near-inertial variability over a range of horizontal scales, there is almost no penetration of near-inertial shear to greater depths.

In contrast, forced by a nearly identical mixed layer flow, the **flow_solve** simulation captures a near inertial wave response with significantly enhanced vertical structure. Figure 14 shows the episodic generation and penetration of near-inertial shear to depths greater than 1,000 m. During the first few days there is very little near-inertial shear below about 150 m. After about a week, the shear has penetrated to all depths. Throughout the simulation, beam patterns emanate episodically from the base of the mixed layer and propagate downward. Vertical profiles of the fast velocity components are shown in the figure. The **flow_solve** solutions exhibit significantly more small-scale structure.

Figure 15 shows the frequency spectra calculated from the vertical shear of the fast horizontal flow components u' and v' at a depth of 388 m (dashed line in Figure 14). In both simulations, the peak frequency response is close to the inertial frequency f_0, though shifted slightly higher in the **flow_solve** simulation. However, at all frequencies, there is substantially more shear in the **flow_solve** wave field than in GL64. The GL64 spectra are narrowly peaked near f_0 while the **flow_solve** spectra decay less rapidly toward both sub- and superinertial frequencies. This is consistent with a higher rate of energy transfer from the forced inertial frequency to higher-frequency internal waves and lower-frequency motions. The canonical decay rate of -2 for the Garrett and Munk internal wave energy spectrum (Garrett & Munk, 1972, 1975) is shown for reference.

Snapshots of u at 21 days, 388 m depth within the nested domain are shown in Figure 16 for both the GL64 and the **flow_solve** simulations. These fields match at the boundaries but differ in the interior. The GL64 flow fields are smooth, whereas the **flow_solve** solutions display significantly enhanced small-scale structure. The enhanced structure results from a combination of finer spatial resolution, higher-order numerical methods, and less restrictive dynamical equations. The bottom two images show the same quantity u but with a colormap chosen to emphasize the enhanced small-scale structure resolved in the nested simulation.

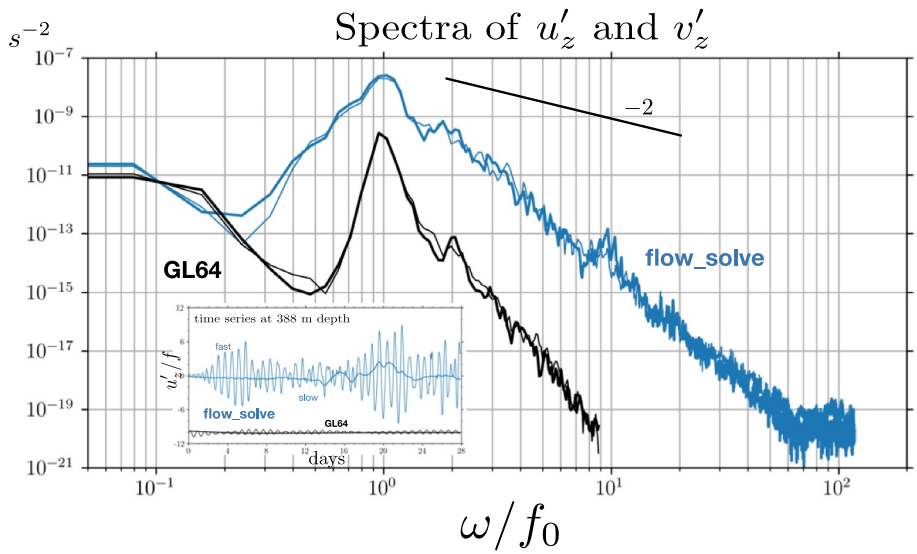

Figure 15. Frequency spectra of zonal (thick) and meridional (thin) shear at approximately 388 m depth for the GL64 (black) and **flow_solve** (blue) simulations. The inset shows the corresponding fast (thin) and slow (low-pass filtered, thick) time series for the zonal component u. The GL64 time series data is offset for clarity.

8. Discussion

In this paper, we have attempted to illustrate, if not formally quantify, the additional physics that a properly implemented nonhydrostatic child run brings to a (somewhat arbitrary) high resolution NEMO run that is likely to excite nonhydrostatic internal waves given the opportunity, that is, when subject to nonhydrostatic dynamics. The first step is to demonstrate that the algorithm is properly implemented. Section 5 shows that with exact boundary data supplied, Algorithm 1 produces the exact interior solution within an arbitrarily embedded subdomain. Section 6 shows that, even when the parent and child runs are subject to the same nonhydrostatic physics, there is an issue related to consistency that can be overcome by using Algorithm 2. Finally, Section 7 applies this algorithm to the GL64 nesting case, where additional issues related to the mixing of hydrostatic boundary data with nonhydrostatic equations arise. We demonstrate that the simple, nearly nondispersive hydrostatic solutions produced by NEMO disperse rapidly when evolved under nonhydrostatic dynamics, that the resulting fields exhibit enhanced shear radiating downward from the surface, and that a 28-day simulation transfers internal wave shear variance to frequencies that are outside the range of validity of the hydrostatic approximation. Direct comparison of a comparable-resolution, self-nested NEMO run is not attempted due to feasibility constraints.

Our approach is built upon a continuous, mixed Bernoulli-Cosine expansion technique that allows the differentiation of discrete data (specified at N equally spaced grid points) lacking boundary symmetries with near-spectral accuracy. Differentiation of the expansion consists of two parts: an O(N) set of Bernoulli polynomial evaluations, and an O($N \log N$) trigonometric transform and inversion. For large N, the extra computation for the mixed expansion compared to a trigonometric expansion for data with appropriate symmetries is negligible.

The expansions are used as the basis functions for the dependent variables in fluid flow simulations with open boundaries. The algorithm exploits separable, parallelized fast Fourier (and sine and cosine) transforms, and thus is constrained to cubic spatial domains with regularly spaced grid points.

In the nonhydrostatic equations of fluid motion Equations 1 and 2, the pressure Equation 6 is elliptic and thus requires boundary conditions. In a typical projection method, the boundary data required, Equation 7, constrain the normal pressure gradient at the boundaries. In our approach, the pressure is decomposed into components ϕ and ψ that satisfy homogeneous and inhomogeneous Neumann conditions at the boundaries, respectively.

The auxiliary field ψ absorbs the boundary inhomogeneity and is defined so that it can be easily and inexpensively computed. We propose two simple methods to do so. The source term for the Poisson equation for ϕ is modified by $\nabla^2\psi$ and the ϕ inversion is computed accurately using fast cosine transforms. Owing to the lack of flow

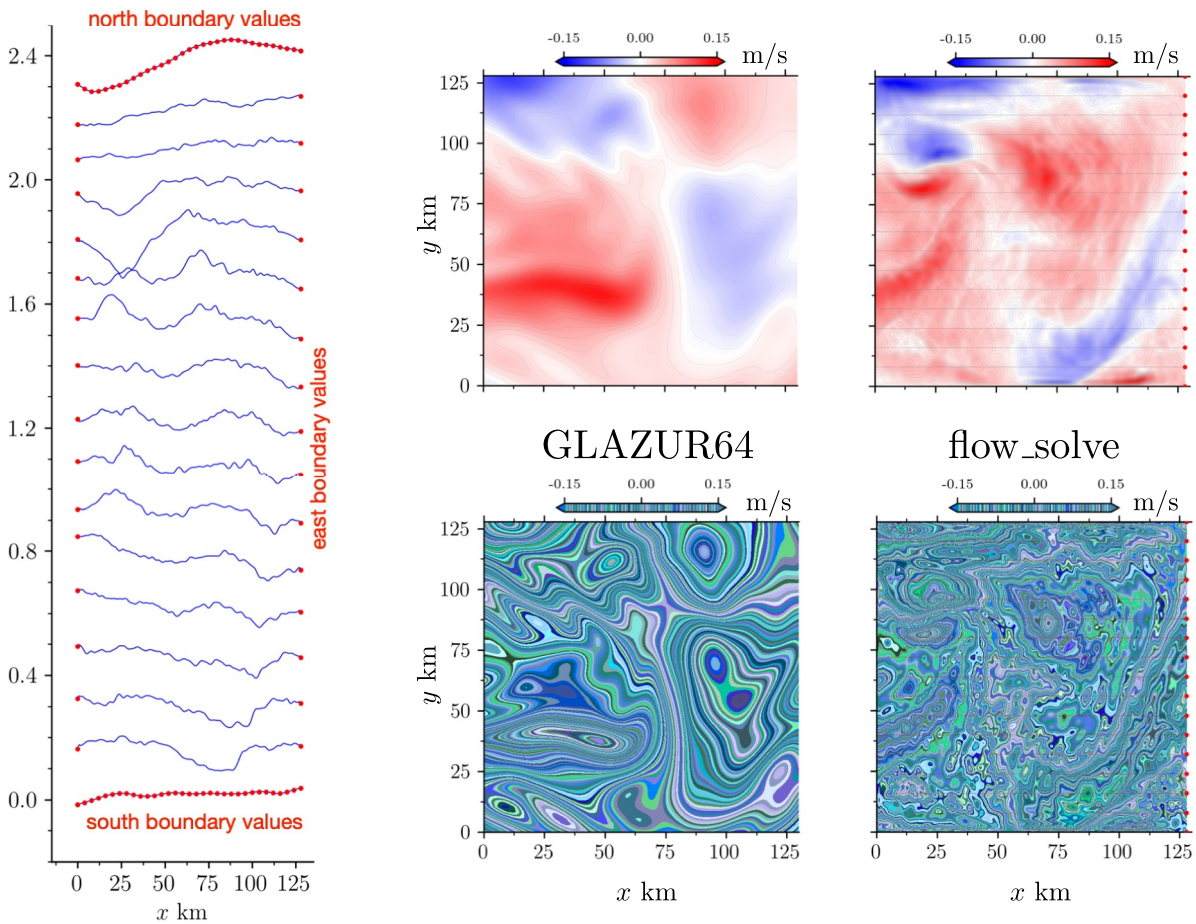

Figure 16. Left: $u(x, y_j)$ at 388 m depth and 21 days along zonal transects at several meridional locations are shown in blue. The imposed boundary values at the corresponding time and locations in the GL64 simulation are shown with red circles. Right: $u(x, y)$ at the same depth and time from the GL64 simulation (left) and from **flow_solve** (right). The color scheme in the lower panels emphasizes the enhanced spatial structure captured with the nested simulation. The red circles and thin lines in the right panels show the meridional locations of the zonal transects in the left panel.

symmetries, the Bernoulli-Cosine differentiation method is required to accurately compute the various gradients needed in the algorithm.

When the boundary data are accurately known, for example, when solving an analytically posed boundary value problem, a straightforward application of the approach (Algorithm 1) works well. When the boundary data are obtained from a coarse resolution outer or parent simulation, an incompatibility arises, and an approximation (Algorithm 2) is introduced to mitigate the problem. The approximation resolves the incompatibility by perturbing boundary conditions within thin boundary layers adjacent to the upper and lower boundaries. Simulations demonstrating the approach are provided in Sections 5 and 6 for a propagating low-mode internal wave and near-inertial wave trapping in an anticyclonic vortex.

Nesting a nonhydrostatic model within a hydrostatic parent simulation introduces modeling issues rather than strictly mathematical algorithmic issues. The boundary data extracted from the parent are not only more coarsely resolved, but are an approximate solution to dynamically reduced equations. In the example of Section 7, the parent data at near inertial frequencies is contaminated by artificially (nearly) non-dispersive wave modes with little vertical structure beneath the mixed layer. At the lateral boundaries, the high frequency component of the boundary data is a relatively small perturbation of the low frequency flows passing through the open boundaries. Imposing the parent data on the normal flow at these boundaries appears to be a reasonable approach. At the bottom boundary, however, the fast component of the parent values of w dominate and are not solutions of the nonhydrostatic equations. They have been discarded as discussed in Section 7.5.

The left panel of Figure 16 shows zonal transects of u at several meridional locations at 388 m depth after 21 days. The blue curves are the values computed in the nested run and the red circles are the imposed boundary values saved from GL64. The lowest and uppermost transects coincide with the southern and northern **flow_solve** boundaries and match the saved boundary data, which has significantly less spatial structure. Outward-propagating small-scale waves will necessarily reflect from the open boundaries. The upper right plan view image shows the locations of the transects (red circles, thin lines).

Some reflection may not be entirely bad. Reflected waves could be considered as a crude model of the dispersive waves that should be entering the domain, about which we have no information to impose. Though the saved boundary data is contaminated at fast inertial time scales, there is important information in the low frequency component. Large-scale thermohaline variability carried by slowly evolving currents can only enter the child domain by imposition of parent boundary conditions for the normal flow and density. When such slow variability is important, it appears that some wave reflection is an inconvenient but unavoidable cost.

In summary, we have proposed, analyzed, and demonstrated a methodology by which the ever-increasing realism of the submesoscale dynamics captured by regional-scale ocean models can be utilized to provide the forcing and geographical context for nonhydrostatic simulations of wind-driven internal gravity waves. We have illustrated single- and doubly-nested strategies and embedded **flow_solve** within the ocean general circulation model NEMO (Madec & Team, 2022) to simulate near-inertial waves in the Gulf of Lion. For this application, we have adopted a simple mixed-layer treatment that takes advantage of the high diffusivity inherent in the GL64 mixed layer. In principle, this could be avoided by implementing a mixed-layer model directly within **flow_solve**. Given the variety of vertical mixing parameterizations and configuration options available, we have not prioritized this inclusion. At present, the main limitation of the proposed approach appears to be the high-frequency contamination of the hydrostatic boundary data from the parent simulation.

Effective strategies for addressing the problem are likely to depend on the application and specific objectives of a given study. Here, we recognize that the direct imposition of deficient boundary data is partially reflective, and focus our analysis on an interior region where the flow appears to be little affected (see Movie S2). As demonstrated in Section 6.3, a second nesting level could be introduced. The saved boundary values in this case would come from dynamically consistent equations, so we expect nearly transparent boundary behavior, as shown in Figure 8. Moreover, a second nesting level with dynamically consistent boundary data would allow for a much shallower lower boundary, enabling further enhanced resolution or computational efficiency as desired. Nonhydrostatic, nested simulations focused on the upper ocean are a feasible approach to obtain both geographical context and the spatial and temporal resolution to resolve internal wave energy transfers from near-inertial waves into the higher frequency internal wave continuum.

Appendix A: flow_solve Equations of Motion

The forced equations of motion for a three-dimensional, stratified fluid in the Boussinesq limit on an f-plane are:

$$\frac{\partial}{\partial t}\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + f\,\hat{z}\times\mathbf{u} - b\hat{z} = -\nabla p + \boldsymbol{\mathcal{F}}_u + \mathcal{D}(\mathbf{u}) \tag{A1}$$

$$\frac{\partial}{\partial t}b + \mathbf{u}\cdot\nabla b = \mathcal{F}_b + \mathcal{D}(b) \tag{A2}$$

$$\nabla\cdot\mathbf{u} = 0. \tag{A3}$$

Here, t is time, f is the Coriolis parameter, and g is the gravitational acceleration. The buoyancy $b = -g(\rho/\rho_0)$, where ρ is the deviation of the density from the reference value ρ_0, and p is the pressure scaled by ρ_0. $\boldsymbol{\mathcal{F}}_u$ and $\mathcal{F}_b$ are external forcing terms for the momentum and buoyancy equations respectively. The equations are solved over a cubic domain of size $(L_x \times L_y \times L_z)$ on a plaid grid with anisotropic spacing $\Delta x = \Delta y >> \Delta z$. The operator $\mathcal{D}$ is the hyperdiffusion operator defined in Equation 39.

The equations are specialized for the problems discussed in Sections 5, 6 and 7 as follows:

A1. Internal Wave Propagation (Section 5)

This problem is unforced and so $\mathcal{F}_u$ and $\mathcal{F}_b$ are set equal to zero. Because the linearized, inviscid and nondiffusive equations of motion can be solved exactly for a uniform depth ocean with free-slip conditions at the bottom and top, we set $\mathcal{D}(\mathbf{u}) = \mathcal{D}(b) = 0$ and solve the equations in the linear (small amplitude) limit. The initial conditions correspond to a snapshot of the exact solution evaluated within an open, interior subdomain. Exact boundary values are imposed as described in Algorithm 1.

A2. Wave Trapping in an Idealized Vortex (Section 6)

The ambient stratification for this problem is prescribed as a 40 m thick mixed layer overlying a constant N ocean. At $t = 0$, we prescribe the initial flow field as an idealized anticyclonic baroclinic vortex in cyclogeostrophic balance. In the absence of additional forcing, the vortex would remain unchanged except for the negligible effects of diffusion.

The flow is then forced for $t > 0$ using a idealized wind stress model. A time series of a spatially uniform 10 m wind field with stochastically varying wind speed and direction was constructed as an Ornstein-Uhlenbeck process. The time series was then augmented by stronger pulse-like wind events with nominal 10 m speeds of 12 m s^{-1} that occur on average every 12 days with a mean duration of 36 hr. The effect of the stochastic wind on the upper ocean is modeled as a time-dependent horizontal body force that exerts uniform stress on the mixed layer flow via the term $\mathcal{F}_u$.

The interaction between the stochastic wind forcing and the ambient vortex excites near-inertial gravity waves that radiate downward and outward and are amplified at the base of the vortex. The nesting levels have progressively finer resolution and correspondingly reduced diffusion. The simulations are run using Algorithm 2.

A3. Nesting Within the NEMO GL64 Simulation (Section 7)

In this run, the initial and boundary data for buoyancy was generated from the temperature and salinity fields of the GL64 simulation using the UNESCO 1980 Equation of State (Jackett & McDougall, 1995; McDougall, 1987). The buoyancy and velocity fields were then linearly interpolated from the spatially irregular NEMO grid onto the regular spatial and temporal flow solve grids. The forcing terms $\mathcal{F}_u$ and $\mathcal{F}_b$ were configured to impose the relaxation discussed in Section 7.4. The overall modeling strategy is described in Section 7 and the simulation was run using Algorithm 2.

Data Availability Statement

The **flow_solve** code, the initial and boundary data, and the simulation output used to make the figures are available at Zenodo via https://doi.org/10.5281/zenodo.8253770 with Creative Commons International 4.0 license (Winters et al., 2024).

Acknowledgments
Funding provided by the US National Science Foundation under Grants OCE-2045270 (KBW, MPL), OCE-2045399 (KBW), OCE-1851572 (MPL) and OCE-1850761 (MC) is gratefully acknowledged. KBW also acknowledges funding from the US Office of Naval Research (Grant N00014-18-1-2803). The GL64 NEMO calculations were performed using GENCI-IDRIS resources (Grant A0110101707). The GL64-**flow_solve** nesting runs were facilitated by the computational, storage, and networking infrastructure of the Hyak supercomputer system at the University of Washington. We also thank three anonymous reviewers for constructive and helpful feedback.

References

Asselin, O., Thomas, L. N., Young, W. R., & Rainville, L. (2020). Refraction and straining of near-inertial waves by barotropic eddies. *Journal of Physical Oceanography*, *50*(12), 3439–3454. https://doi.org/10.1175/JPO-D-20-0109.1

Balwada, D., Xie, J.-H., Marino, R., & Feraco, F. (2022). Direct observational evidence of an oceanic dual kinetic energy cascade and its seasonality. *Science Advances*, *8*(41), eabq2566. https://doi.org/10.1126/sciadv.abq2566

Barrier, N., Ourmières, Y., & Petrenko, A. (2016). Intrusions of the Northern Mediterranean Current on the eastern Gulf of Lion: Insights from in-situ observations and high resolution numerical modelling. *Ocean Dynamics*, *66*(3), 313–327. https://doi.org/10.1007/s10236-016-0921-7

Berline, L., Zakardjian, B., Molcard, A., Ourmières, Y., & Guihou, K. (2013). Modeling jellyfish Pelagia noctiluca transport and stranding in the Ligurian Sea. *Marine Pollution Bulletin*, *70*(1–2), 90–99. https://doi.org/10.1016/j.marpolbul.2013.02.016

Booker, J. R., & Bretherton, F. P. (1967). The critical layer for internal gravity waves in a shear flow. *Journal of Fluid Mechanics*, *27*(3), 513–539. https://doi.org/10.1017/S0022112067000515

Bouyssel, F., Berre, L., Bénichou, H., Chambon, P., Girardot, N., Guidard, V., et al. (2022). The 2020 global operational NWP data assimilation system at Météo-France. In S. Park & L. Xu (Eds.), *Data assimilation for atmospheric, oceanic and hydrologic applications* (Vol. IV, pp. 645–664). Springer. https://doi.org/10.1007/978-3-030-77722-7_25

Capet, X., McWilliams, J. C., Molemaker, M. J., & Shchepetkin, A. F. (2008). Mesoscale to submesoscale transition in the California Current system. Part I: Flow structure, eddy flux, and observational tests. *Journal of Physical Oceanography*, *38*(1), 29–43. https://doi.org/10.1175/2007JPO3671.1

Chorin, A. J. (1968). Numerical solution of the Navier-Stokes equations. *Mathematics of Computation*, *22*(104), 745–762. https://doi.org/10.1090/s0025-5718-1968-0242392-2

Costabile, F., & Dell' Accio, F. (2001). Expansion over a rectangle of real functions in Bernoulli polynomials and applications. *BIT*, *41*(3), 451–464. https://doi.org/10.1023/A:1021958910686

DiBattista, M. T., Majda, A. J., & Marshall, J. (2002). A statistical theory for the "patchiness" of open-ocean deep convection: The effect of preconditioning. *Journal of Physical Oceanography*, *32*(2), 599–626. https://doi.org/10.1175/1520-0485(2002)032⟨0599:ASTFTP⟩2.0.CO;2

Eckhoff, K. S. (1993). Accurate and efficient reconstruction of discontinuous functions from truncated series expansions. *Mathematics of Computation*, *61*(204), 745–763. https://doi.org/10.1090/S0025-5718-1993-1195430-1

Eckhoff, K. S. (1998). On a high order numerical method for functions with singularities. *Mathematics of Computation*, *67*(223), 1063–1087. https://doi.org/10.1090/s0025-5718-98-00949-1

Fringer, O., Gerritsen, M., & Street, R. (2006). An unstructured-grid, finite-volume, nonhydrostatic, parallel coastal ocean simulator. *Ocean Modelling*, *14*(3), 139–173. https://doi.org/10.1016/j.ocemod.2006.03.006

Garrett, C., & Munk, W. (1972). Oceanic mixing by breaking internal waves. *Deep-Sea Research and Oceanographic Abstracts*, *19*(12), 823–832. https://doi.org/10.1016/0011-7471(72)90001-0

Garrett, C., & Munk, W. (1975). Space-time scales of internal waves: A progress report. *Journal of Geophysical Research*, *80*(3), 291–297. https://doi.org/10.1029/JC080i003p00291

Gottlieb, D., & Shu, C.-W. (1997). On the Gibbs phenomenon and its resolution. *SIAM Review*, *39*(4), 644–668. https://doi.org/10.1137/S0036144596301390

Guermond, J., Minev, P., & Shen, J. (2006). An overview of projection methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, *195*(44), 6011–6045. https://doi.org/10.1016/j.cma.2005.10.010

Guihou, K., Marmain, J., Ourmières, Y., Molcard, A., Zakardjian, B., & Forget, P. (2013). A case study of the mesoscale dynamics in the North-Western Mediterranean Sea: A combined data-model approach. *Ocean Dynamics*, *63*(7), 793–808. https://doi.org/10.1007/s10236-013-0619-z

Huang, H., Song, P., Qiu, S., Guo, J., & Chen, X. (2023). A nonhydrostatic oceanic regional model, ORCTM v1, for internal solitary wave simulation. *Geoscientific Model Development*, *16*(1), 109–133. https://doi.org/10.5194/gmd-16-109-2023

Jackett, D. R., & McDougall, T. J. (1995). Minimal adjustment of hydrographic profiles to achieve static stability. *Journal of Atmospheric and Oceanic Technology*, *12*(2), 381–389. https://doi.org/10.1175/1520-0426(1995)012⟨0381:MAOHPT⟩2.0.CO;2

Komatsu, T., & Pita, C. (2016). Several explicit formulae for Bernoulli polynomials. *Mathematical Communications*, *21*, 127–140.

Kunze, E., Schmitt, R. W., & Toole, J. M. (1995). The energy balance in a warm-core ring's near-inertial critical layer. *Journal of Physical Oceanography*, *25*(5), 942–957. https://doi.org/10.1175/1520-0485(1995)025⟨0942:TEBIAW⟩2.0.CO;2

Lanczos, C., & Boyd, J. (2016). *Discourse on Fourier series*. Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9781611974522

Lelong, M.-P., Cuypers, Y., & Bouruet-Aubertot, P. (2020). Near-inertial energy propagation inside a Mediterranean anticyclonic eddy. *Journal of Physical Oceanography*, *50*(8), 2271–2288. https://doi.org/10.1175/JPO-D-19-0211.1

Madec, G., & Team, N. S. (2022). *NEMO ocean engine (Computer software manual No. 27)*. Zenodo. https://doi.org/10.5281/zenodo.1464816

Marino, R., Pouquet, A., & Rosenberg, D. (2015). Resolving the paradox of oceanic large-scale balance and small-scale mixing. *Physical Review Letters*, *114*(11), 114504. https://doi.org/10.1103/PhysRevLett.114.114504

McDougall, T. J. (1987). Neutral surfaces. *Journal of Physical Oceanography*, *17*(11), 1950–1964. https://doi.org/10.1175/1520-0485(1987)017⟨1950:NS⟩2.0.CO;2

Ourmières, Y., Brankart, J.-M., Berline, L., Brasseur, P., & Verron, J. (2006). Incremental analysis update implementation into a sequential ocean data assimilation system. *Journal of Atmospheric and Oceanic Technology*, *23*(12), 1729–1744. https://doi.org/10.1175/JTECH1947.1

Ourmières, Y., Mansui, J., Molcard, A., Galgani, F., & Poitou, I. (2018). The boundary current role on the transport and stranding of floating marine litter: The French Riviera case. *Continental Shelf Research*, *155*, 11–20. https://doi.org/10.1016/j.csr.2018.01.010

Shcherbina, A. Y., Sundermeyer, M. A., Kunze, E., D'Asaro, E., Badin, G., Birch, D., et al. (2015). The LatMix summer campaign: Submesoscale stirring in the upper ocean. *Bulletin of the American Meteorological Society*, *96*(8), 1257–1279. https://doi.org/10.1175/BAMS-D-14-00015.1

Simmons, H., Chang, M.-H., Chang, Y.-T., Chao, S.-Y., Fringer, O., Jackson, C. R., & Ko, D. S. (2011). Modeling and prediction of internal waves in the South China Sea. *Oceanography*, *24*(4), 88–99. https://doi.org/10.5670/oceanog.2011.97

Srinivasan, K., Barkan, R., & McWilliams, J. C. (2023). A forward energy flux at submesoscales driven by frontogenesis. *Journal of Physical Oceanography*, *53*(1), 287–305. https://doi.org/10.1175/JPO-D-22-0001.1

Strang, G. (1999). The discrete cosine transform. *SIAM Review*, *41*(1), 135–147. https://doi.org/10.1137/s0036144598336745

Temam, R. (1969). Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (II). *Archive for Rational Mechanics and Analysis*, *33*, 377–386. https://doi.org/10.1007/BF00247696

Thomas, L. N., Rainville, L., Asselin, O., Young, W. R., Girton, J., Whalen, C. B., et al. (2020). Direct observations of near-inertial wave ζ-refraction in a dipole vortex. *Geophysical Research Letters*, *47*(21), e2020GL090375. https://doi.org/10.1029/2020GL090375

Vitousek, S., & Fringer, O. B. (2014). A nonhydrostatic, isopycnal-coordinate ocean model for internal waves. *Ocean Modelling*, *83*, 118–144. https://doi.org/10.1016/j.ocemod.2014.08.008

Winters, K. B. (2016). The turbulent transition of a supercritical downslope flow: Sensitivity to downstream conditions. *Journal of Fluid Mechanics*, *792*, 997–1012. https://doi.org/10.1017/jfm.2016.113

Winters, K. B., Claret, M., Lelong, P., & Ourmières, Y. (2024). Codes and data for a pressure projection scheme with near-spectral accuracy for nonhydrostatic flow in domains with open boundaries [Software]. *Zenodo*. https://doi.org/10.5281/zenodo.8253770

Winters, K. B., & de la Fuente, A. (2012). Modelling rotating stratified flows at laboratory-scale using spectrally-based DNS. *Ocean Modelling*, *49–50*, 47–59. https://doi.org/10.1016/j.ocemod.2012.04.001