Fast and Accurate Computation of Vertical Modes

Jeffrey J. Early¹, M. Pascale Lelong¹, and K. Shafer Smith²

¹NorthWest Research Associates, Redmond, WA, USA, ²Center for Atmosphere Ocean Science, Courant Institute of Mathematical Sciences, New York University, New York, NY, USA

Abstract The vertical modes of linearized equations of motion are widely used by the oceanographic community in numerous theoretical and observational contexts. However, the standard approach for solving the generalized eigenvalue problem using second-order finite difference matrices produces \( O(1) \) errors for all but the few lowest modes, and increasing resolution quickly becomes too slow as the computational complexity of eigenvalue algorithms increases as \( O(n^3) \). Existing methods are therefore inadequate for computing a full spectrum of internal waves, such as needed for initializing a numerical model with a full internal wave spectrum. Here we show that rewriting the eigenvalue problem in stretched coordinates and projecting onto Chebyshev polynomials results in substantially more accurate modes than finite differencing at a fraction of the computational cost. We also compute the surface quasigeostrophic modes using the same methods. All spectral and finite difference algorithms are made available in a suite of Matlab classes that have been validated against known analytical solutions in constant and exponential stratification.

Plain Language Summary The ocean is filled with internal waves that primarily move fluid back and forth horizontally and, to a lesser degree, vertically. The amount of vertical motion for each wave depends on depth because the fluid density in the ocean also changes with depth. The density profile of the ocean varies both seasonally and geographically, which means that vertical structure of each wave is also a function of season and geographic location. In order to diagnose the sizes of the internal waves, scientists need to accurately compute the motion predicted for each wave from a given density profile. The standard methodology for computing the vertical structure of the fluid motion only works well for the biggest and slowest moving waves. This paper introduces a different methodology which is both computationally faster and more accurate than the traditional methods. All of the code for computing the vertical structure of waves is implemented in Matlab and is freely available. The paper also explores some of the most common errors that will be encountered when using ocean data to diagnose the vertical structure.

1. Introduction

Vertical modes arise as part of the separable solution to both the internal wave problem and quasigeostrophic theory. The eigenvalue problem (EVP) is treated in many introductory physical oceanography textbooks, (e.g., Cushman-Roisin & Beckers, 2011; Gill, 1982), and the resulting vertical modes describe the vertical structure of the linear solutions for a given density profile. While there are an infinite number of bases that can be used to represent ocean currents and density anomalies satisfying certain boundary conditions, the vertical modes correspond to \( O(1) \) dynamical solutions of the equations of motion and are therefore both diagnostic and prognostic. For this reason, vertical modes are the standard basis with which to represent the vertical structure of ocean currents, and it would be hard to overstate their usefulness for describing and modeling the ocean.

There are two primary uses for the vertical modes: (1) projecting a given flow field onto the vertical modes to determine its spectrum (a forward transformation) or (2) creating a dynamically consistent linear flow field from a given spectrum (an inverse transformation). For example, projection onto the vertical modes (the forward transformation) was used to construct the Garrett-Munk internal wave spectrum (Munk, 1981; Polzin & Lvov, 2011), analyze regional and global ocean simulations (Buijsman et al., 2010; Kelly et al., 2012; Zilberman et al., 2009), interpret satellite altimetry observations (LaCasce & Wang, 2015; Zhao, 2016), and to describe the vertical structure of balanced flow in the ocean (Wortham & Wunsch, 2014; Wunsch, 1997).
Examples of using the inverse transform are less common but typically involve initializing a full spectrum of internal waves (Sun & Kunze, 1999; Winters & D’Asaro, 1997).

This study was motivated by two situations where current techniques for computing vertical modes were found to have significant errors for reasonable computation times. In the first, perhaps less common situation, a numerical model needed to be initialized with a full spectrum of internal waves—a task which requires solving an EVP at each resolved wavenumber in the model. In the second situation, we sought to compare an observed horizontal velocity spectrum of internal waves to the Garrett-Munk spectrum near a very strong pycnocline. This requires computing vertical modes at high frequencies—which requires appropriately resolving the mode variability near the pycnocline.

There are a number of sources of error that arise in performing either the forward or inverse transformation. These include the following:

1. A poorly defined mean density function.
2. Measurement noise and uncertainty in the density function and in the case of a forward transform the dynamical variables.
3. Aliasing error, due to the location of the grid points of the dynamical variables (relevant only for forward transform).
4. Interpolation error, due to the location of (and lack of) data points specifying the density function.
5. Numerical truncation error of the modes generated from the vertical EVP.

Whether performing a forward or inverse transformation, these sources of error will compound in some fashion to create error in either the resulting flow field or the resulting spectrum.

The first source of error, a poorly defined mean density function, can result from lack of data or often just uncertainty in what qualifies as a mean (e.g., questions of what time and length scales to average over or even whether averaging is the correct approach for a nonlinear system). Measurement noise and uncertainty in data is a topic unto itself, so the methods here proceed without concern for measurement noise when possible. The aliasing error arises when the grid points are placed such that higher modes project onto the lower modes. This source of error is relatively easily controlled by computing the condition number of the projection matrix, which provides a fairly precise cutoff for the number of resolvable modes. On the rare occasion that a density function can be specified analytically, the interpolation error can be minimized to numerical precision. However, in the usual case where the density is given on some grid with uneven spacing, the density function must be interpolated in between those grid points. Our work suggests that density interpolation error does not dominate the error for most cases. This manuscript is therefore largely concerned with the final source of errors—arising from the numerical representation of the modes in the vertical EVP.

The standard method for solving the vertical EVP is to discretize the problem and construct the matrices using second-order finite difference matrices (e.g., Cushman-Roisin & Beckers, 2011). However, this approach produces unacceptable errors for all but the very lowest modes in the simplest stratifications. This is problematic because numerical algorithms for solving EVPs scale as $O(n^3)$, where $n$ is the number of discretization points in the vertical, so small increases in resolution come at a large computational cost.

Instead of using finite difference methods, Kelly (2016) solves the hydrostatic form of the EVP spectrally using Galerkin’s method, at a fraction of the computational cost and with much higher accuracy. Dunphy (2009) uses a Chebyshev collocation method to solve the nonhydrostatic case with fixed frequency. Here, we extend the ideas of Kelly (2016) and Dunphy (2009) to the more general nonhydrostatic cases where the EVP must be solved for each frequency in the spectrum or each resolved wavenumber in a numerical model. Our approach solves the EVP spectrally with Chebyshev polynomials to produce high-quality vertical modes, even with relatively small $n$. The same techniques are then applied to solve for surface quasigeostrophic modes, used to describe the effect of density anomalies at the ocean boundaries.

To illustrate the differences in performance of these two methods, we consider the 10th vertical mode at two different frequencies in a realistic stratification profile. The left panel in Figure 1 shows an example stratification profile taken from the Sargasso Sea during the Latmix 2011 summer field experiments (Shcherbina et al., 2015). The 10th vertical mode is computed with 512 grid points using both second-order finite difference methods (blue) and spectral methods (purple). The center panel shows that, at hydrostatic frequencies, finite difference methods perform well. However, the right panel shows the 10th mode at a nonhydrostatic
Figure 1. The left panel shows a stratification profile taken from the Sargasso Sea in the summertime (Shcherbina et al., 2015). The two vertical lines indicate a nearly hydrostatic frequency (dashed) and nonhydrostatic frequency (dotted). The 10th mode computed using finite difference methods (blue) and spectral methods (purple) is compared at the hydrostatic frequency (center panel) and nonhydrostatic frequency (right panel). Both methods use 512 evenly spaced grid points as input then interpolate to a high-resolution grid as output.

frequency (corresponding to the dashed line in the left panel) for which the finite difference method fails. Generally, we find that finite difference methods with \( N \) grid points will return about \( N/10 \) good modes, although this is stratification and grid point location dependent. For many low-mode low-frequency studies then, finite difference methods are fully up to the task. However, for nonhydrostatic frequencies or high modes, the methods described below are more suitable.

This paper begins with a derivation of the two most relevant forms of the vertical EVP that arise from the linearized equations of motion in section 2. This provides the necessary context for orthogonality relations that form the basis of the normalization of the vertical modes, which in turn shows limitations of using certain vertical modes as a basis. Section 3 demonstrates some of the problems associated with finite differencing, while section 4 shows how these can be remedied using spectral methods. Details of the numerical implementation are described in section 5, and some of the other sources of error are examined in section 6. Section 7 discusses some best practices and potential pitfalls. Finally, the appendices include the exact analytical solutions for constant and exponential stratification that are employed for unit testing, the WKB (Wentzel-Kramers-Brillouin) approximated solution used for the adaptive grid method, and the class inheritance tree of the publicly available Matlab implementation of these methods.

2. Background

The linearized equations of motion for the fluid velocity \( u(x, y, z, t) \), \( v(x, y, z, t) \), \( w(x, y, z, t) \), on the \( f \)-plane are

\[
\partial_z u - f_0 v = - \frac{1}{\rho_0} \partial_z p \\
\partial_z v + f_0 u = - \frac{1}{\rho_0} \partial_z p \\
\partial_z w = - \frac{1}{\rho_0} \partial_z p - g \frac{\rho}{\rho_0} \\
\partial_z u + \partial_z v + \partial_z w = 0 \\
\partial_z \rho + \omega \partial_z \bar{\rho} = 0
\]

where \( p(x, y, z, t) \) and \( \rho(x, y, z, t) \) are the perturbation pressure and density, respectively. These are defined such that the total pressure \( p_{tot}(x, y, z, t) = p(x, y, z, t) + p_0(z) \) and the total density \( \rho_{tot}(x, y, z, t) = \rho_0 + \bar{\rho}(z) + \)
\( \rho(x, y, z, t) \), where \( \partial_z p_r(z) = -g \hat{\rho}(z) \). All variables in the equations of motion are functions of \( x, y, z \) and \( t \), except for \( \hat{\rho} \) which is strictly a function of \( z \). The operator \( \partial_z \) is understood to reduce to \( \frac{d}{dz} \) when applied to univariate functions. We use the usual definition of the buoyancy frequency \( N^2(z) \equiv -\frac{F}{\rho_0} \partial_z \hat{\rho} \).

There are three linearly independent solutions to equations (1)–(5), assuming periodic horizontal boundary conditions and a flat bottom: two wave solutions and the geostrophic solution.

### 2.1. Wave Solutions

The positive frequency wave solution is given by

\[
\begin{bmatrix}
\rho_v \\
u_v \\
v_v \\
w_v \\
\rho_v \\
F_v
\end{bmatrix} = A \begin{bmatrix}
-\rho_0 g \frac{k}{\omega} \cos \theta, F(z) \\
\cos \alpha \cos \theta + \frac{\omega}{\omega_0} \sin \alpha \sin \theta, F(z) \\
\sin \alpha \cos \theta - \frac{\omega}{\omega_0} \cos \alpha \sin \theta, F(z) \\
K_h \sin \theta, G(z) \\
\frac{dh}{dt} \sin \theta, G(z) \\
\frac{dh}{dt} \cos \theta, G(z)
\end{bmatrix}
\]

where the functions \( F(z) \) and \( G(z) \) are the eigenfunctions with corresponding eigenvalue \( h \), to be discussed in detail below. The frequency is determined through the dispersion relation,

\[
\omega = \sqrt{gh(k^2 + l^2) + \frac{2}{\omega_0}}
\]

and the negative rotating wave solution is found by flipping the sign on the frequency, \( \omega \mapsto -\omega \). In this notation the phase angle of the wave is given by \( \alpha = \tan^{-1} \left( \frac{l}{k} \right) \) and \( K = \sqrt{k^2 + l^2} \) is the total horizontal wavenumber. The horizontal phase is given by \( \theta_x = kx + ly \pm \omega t \). The eigenvalue \( h \) is referred to as the equivalent depth and is related to the wave group velocity, \( c_g = \sqrt{gh} \). The value \( h \) can be replaced in favor of eigenfrequency \( \omega \) using the dispersion relation, but equation (6) uses both in order to avoid singularities at \( K = 0 \) and for notational compactness. Applying this solution to equations (1)–(5) leads to two coupled equations for the vertical structure functions,

\[
(N^2 - \omega^2)G = -g \partial_z F \quad \text{and} \quad F = h \partial_z G
\]

which can be combined into various second-order EVPs (see section 2.3).

### 2.2. Geostrophic Solution

The geostrophic solution is given by

\[
\begin{bmatrix}
\rho_g \\
u_g \\
v_g \\
w_g \\
\rho_g \\
F_g
\end{bmatrix} = B \begin{bmatrix}
\rho_0 \cos \theta, F(z) \\
\frac{\omega}{\omega_0} l \sin \theta, F(z) \\
\frac{\omega}{\omega_0} l \sin \theta, F(z) \\
\frac{\omega}{\omega_0} l \sin \theta, F(z) \\
0 \\
\frac{dh}{dt} \cos \theta, G(z)
\end{bmatrix}
\]

where \( \theta_0 = kx + ly \). The solution can also be written entirely in terms of stream function \( \psi(x, y, z) = \frac{1}{g} \cos \theta \psi, F(z) \), where \( (u_v, v_v, \rho_v) = (\partial_y \psi, \partial_x \psi, -\rho_0 f \partial_x \psi) \). Satisfying the vertical momentum equation requires that \( N^2 G = -g \partial_z F \), but unlike the wave solutions, geostrophic solutions already satisfy continuity. For a given wavenumber \( (k, l) \) the geostrophic solution is entirely specified by a vertical profile of any one of the variables, from which all others are immediately deduced. For example, \( \hat{\rho}(k, l, z) \) determines \( G(z) \), from which \( F(z) \) is determined by integration—the thermal wind balance. There is no preferred basis for the geostrophic solution.

Although the scalings that lead to equations (1)–(5) result in the linear geostrophic solution where \( w_g = 0 \), near-geostrophic theories with a different choice of scalings, such as quasi-geostrophy (Pedlosky, 1987), have nonzero vertical velocities \( w_g \neq 0 \) and therefore require full continuity, that is, that \( F = h \partial_z G \). As with the wave solution, this requirement combined with \( N^2 G = -g \partial_z F \) results in an EVP, detailed in section 2.3.

An eigenbasis constructed with the rigid lid boundary condition \( u(0) = G(0) = 0 \) precludes nonzero density anomalies at the surface. One work-around to this limitation is to further decompose the geostrophic
solution into three parts: two parts resulting from the density anomaly at the boundaries and one part from the remaining density anomaly in the interior. Following Lapeyre and Klein (2006), the idea is then to let

$$\psi = \psi^{\text{int}} + \psi^{\text{sur}} + \psi^{\text{bot}}$$  \hspace{1cm} (10)

where both the surface and bottom components of the flow are required to have no potential vorticity in the interior,

$$\nabla^2 \psi^{\text{sur/bot}} + \frac{f_0}{N^2} \partial_z \psi^{\text{sur/bot}} = 0$$  \hspace{1cm} (11)

but account for the density anomaly at the boundaries, for example,

$$f_0 \partial_z \psi^{\text{sur}} = -\frac{g}{\rho_0} \left. \right|_{z=0}, \quad f_0 \partial_z \psi^{\text{bot}} = 0 \mid_{z=-D}$$  \hspace{1cm} (12)

The resulting modes can be solved for a given wavenumber and are referred to as the surface quasi-geostrophic (SQG) modes. This methodology has been used to construct the interior velocity field from sea surface height (SSH) and temperature data Wang et al. (2013).

Smith and Vanneste (2013) formulate a new EVP that results in modes capable of capturing surface density anomalies for quasigeostrophic flows. Taking equation (9) from Smith and Vanneste (2013) and writing it in the present notation, we have

$$\partial_z G_j - \frac{K^2 N^2}{f_0} G_j = -\frac{N^2}{g h_j} G_j$$  \hspace{1cm} (SV EVP)

with surface boundary condition $G(0) = \frac{B}{a} \partial_z G$, where $a$ is a tunable parameter. Importantly, these modes remain orthogonal, unlike the combined set of SQG modes and interior modes described above.

An alternative to both the SQG and the Smith and Vanneste (2013) approaches is to use the internal wave eigenbasis constructed with the free surface boundary condition (detailed below) which also results in orthogonal modes capable of capturing nonzero density anomaly at the ocean surface.

### 2.3. Vertical EVP

The vertical EVP is formed using the two coupled equations from section 2.1 for the vertical structure functions, $(N^2 - \alpha^2)G = -g \partial_z F$ (vertical momentum) and $F = h \partial_z G$ (continuity). In combination with the dispersion relation, one of the eigenfunctions can be eliminated, resulting in various single second-order EVPs for the vertical structure functions $F$ or $G$. The two most practical EVPs to solve are for $G(z)$ with constant $\omega$,

$$\partial_z G_j = -\frac{N^2 - \alpha^2}{g h_j} G_j$$  \hspace{1cm} (\omega-constant EVP)

and for $G(z)$ with constant $K$,

$$\partial_z G_j - K^2 G_j = -\frac{N^2 - f_0^2}{g h_j} G_j$$  \hspace{1cm} (K-constant EVP)

The near-geostrophic EVP is found by combining the vertical momentum condition $N^2 G = -g \partial_z F$ with the continuity condition $F = h \partial_z G$, which can be treated as the (\omega-constant EVP) with $\omega = 0$. Note that this is equivalent to making the hydrostatic approximation (Kelly, 2016) and removes all dependence on frequency $\omega$ and wavenumber $K$.

For the cases considered here we take the lower boundary condition at $z = -D$ to be either free slip, where $w(-D) = 0$, or no slip, where $w(-D) = 0$. These correspond to $G(-D) = 0$ and $F(-D) = 0$, respectively. These conditions can be seen as the limiting cases of sloped bottom topography (LaCasce, 2017). The surface boundary condition at $z = 0$ is taken to be either a rigid lid with $w(0) = 0$, $G(0) = 0$ or a free surface approximated as $p(x, y, 0) = \rho_0 g \eta(x, y, 0)$, where $\eta \equiv -\langle \partial_z \rho \rangle^{-1} \rho$ is the linear approximation of the isopycnal displacement. In terms of the vertical modes, the free surface boundary condition becomes $h \partial_z G(0) = G(0)$.

Finally, there are many cases where the density profile does not extend to the full depth of the ocean and no boundary conditions (beyond the EVP itself) should be added.
Solving either EVP yields a set of eigenvalues \( h_i \) that can be ordered such that \( h_1 > h_2 > h_3 > \ldots > h_n \), each with corresponding eigenfunction \( G_j \). This means that solving the \((\omega\)-constant EVP) results in wave solutions with the same frequency \( \omega \), but different wavenumbers \( k_j \), and similarly solving the \((K\)-constant EVP) results in wave solutions with the same wavenumber \( K \) and different frequencies \( \omega_j \). Although we do not implement this numerically, note that rearranging the \((K\)-constant EVP) poses the EVP for fixed group velocity, \( gh \), with eigenvalue \( K_j^2 \).

The equations for the \((\omega\)-constant EVP) and \((K\)-constant EVP) are both Sturm-Liouville problems and share the property that their eigenmodes are orthogonal. Following the procedure in Kelly (2016), the eigenmodes found with the \((\omega\)-constant EVP) satisfy

\[
G_i(0)G_j(0) + \frac{1}{8} \int_D^0 (N^2(z) - \omega^2) G_i G_j \, dz = \beta \delta_{ij}
\]

and

\[
\int_D^0 F_i F_j \, dz = \beta h_i \delta_{ij}
\]

while the eigenmodes found with \((K\)-constant EVP) satisfy

\[
G_i(0)G_j(0) + \frac{1}{8} \int_D^0 (N^2(z) - f_0^2) G_i G_j \, dz = \gamma \delta_{ij}
\]

and

\[
\int_D^0 (F_i F_j + h_i h_j K^2 G_i G_j) \, dz = \gamma h_i \delta_{ij}
\]

where \( \beta \) and \( \gamma \) are unspecified constants that depend on the chosen normalization, as discussed below. It is important to note that these orthogonality conditions only apply for a particular choice of \( \omega \) or \( K \). For example, an eigenmode \( G_i(z, k_i) \) found using \( K = k_i \) is not orthogonal to an eigenmode \( G_j(z, k_j) \) found using \( K = k_j \) if \( k_1 \neq k_2 \).

The most significant difference between the two EVPs is that eigenmodes from the \((K\)-constant EVP) often form a complete basis set for typical ocean stratification profiles, while the eigenmodes from the \((\omega\)-constant EVP) do not. The \((K\)-constant EVP) is a regular Sturm-Liouville problem when the weighting function \( w_\omega(z) \equiv N^2 - f_0^2 > 0 \) for all \( z \), a condition typically met in the ocean. We note that although it is fair to say that stratification with \( N > f_0 \) is typical of the world oceans, after examining 30,000 CTD profiles, Kunze (2017) finds that 10% of the data have \( N < 2f \) and a full 30% of the data suggest \( N < 2f \) within 380 m of the bottom. In contrast, the weighting function \( w_\omega(z) \equiv N^2 - \omega^2 \) in the \((\omega\)-constant EVP) switches sign at turning points \( z_\omega \), where \( N(z_\omega) = \omega \). Consequently, the norm of an arbitrary function defined on the domain \([-D, 0]\) and satisfying the boundary conditions is not guaranteed to be positive using the norm implied by equation (13), a necessary condition for completeness. Intuitively, this can be seen in Figure 4, which shows that the high-frequency modes have no variance beyond the turning points and are therefore incapable of representing arbitrary functions on the domain.

### 2.4. Normalization

The amplitude of each vertical mode can be scaled by an arbitrary constant, so one must choose a normalization appropriate for the problem being considered. The four most common scenarios are setting the total energy, a horizontal velocity \( (U) \), a vertical velocity \( (W) \), and the SSH of a given wave.

To set the total energy of the internal wave solution in equation 6, we use the modes from the \((K\)-constant EVP) and therefore use the norm implied by equation 15,

\[
G_i(0) + \frac{1}{8} \int_D^0 (N^2(z) - f_0^2) G_i^2 \, dz = 1 \quad \text{\( (K\)-constant norm) }
\]

where we have chosen \( \gamma = 1 \) as the normalization constant in order to keep the vertical modes unitless. Another reasonable choice is to take \( \gamma = N_0^2 D/g \), but using \( \gamma = 1 \) keeps the norm universal, rather than problem specific. Taking the total energy of the wave,

\[
E(x, y, z, t) = \frac{1}{2} \left( u^2 + v^2 + w^2 + \frac{g^2}{\rho_0 N^2} \right)
\]

(17)
and then depth integrating and averaging over space and time produces a wave with energy $P^2/2$ if we set
the coefficient $A = P/\sqrt{\lambda J} - (\omega^2 - f^2)G^2(0)/\omega^2$ in equation 6.

Setting the maximum initial eastward velocity to $U$ can be accomplished by imposing $\max F_j = 1$ and $A = U$. The maximum vertical velocity $W$ is set using the norm $\max G_j = 1$ where $A = W/(Kh_j)$, but is clearly singular for inertial waves which have no vertical velocity. The SSH is set using the pressure at the surface with $F(0) = 1$ and $A = \text{SSH} - \omega^2 h$.

To set the total energy of the interior geostrophic solution in equation 9, we assume the solution uses the typical geostrophic modes from equation $\langle \omega - \text{constant EVP} \rangle$ with $\omega = 0$ and therefore use the norm implied by equation 13,

$$\frac{1}{D} \int_{-D}^0 F_j^2 \, dz = 1 \quad (\omega - \text{constant norm})$$

where we have taken $\beta = D/h$. This produces a mode with energy $P^2$ if we let,

$$B^2 = \frac{4P^2 f^2 h}{gD} \left( \frac{1}{ghK^2 + f^2(1 + hG^2(0)/D)} \right).$$

Setting the maximum eastward velocity requires $B = U \frac{\bar{f}}{\beta}$ using $\max F_j = 1$. The SSH is set using the pressure at the surface by setting $F(0) = 1$ and $B = \text{SSH}$.

### 3. The Problem With Finite Differencing

Computing the lowest vertical modes with finite differencing methods can be relatively fast and accurate when considering a single wavenumber or frequency. Although one can encounter problems with the higher modes, this can usually be ameliorated by increasing resolution. The primary issues with finite differencing arise when needing to compute many modes across a broad range of frequencies and wavenumbers—the two scenarios that motivated the present study. To compute a complete internal wave frequency spectrum requires solving the $\langle \omega - \text{constant EVP} \rangle$ at each resolved frequency between the Coriolis frequency and the maximum buoyancy frequency, roughly $O(10^2)$ EVPs. This is especially challenging near the buoyancy frequency, where all oscillations occur in the narrow region where $N(z) < \omega$. On the other hand, initializing a numerical model with an internal wave spectrum involves solving the $\langle K - \text{constant EVP} \rangle$ for each resolved wavenumber in the model, which easily requires $O(10^4)$ computations or more.

A prerequisite to initializing a numerical model with a precise internal wave spectrum is that the modes must be computed for each unique horizontal wavenumber $K$ resolved by the model using the $\langle K - \text{constant EVP} \rangle$. If the numerical model has $(N_x, N_y)$ horizontal grid points, approximately $N_x N_y$ unique EVPs must be solved (up to another factor of 2 can be eliminated with isotropic horizontal resolution). Unfortunately, eigenvalue algorithms scale as $O(n^3)$ for $n$ by $n$ matrices. This means that initialization of an internal wave spectrum scales as $O(N_x N_y N_z^3)$, and thus, with any reasonable vertical resolution, this will quickly become a rate limiting step to a model run. In practical terms, the computation time of these EVPs takes approximately 1, 10, and 100 s for $N_z$ of 512, 1,024, and 2,048, respectively, on consumer hardware from 2015.

The problem is further exacerbated by the poor performance of finite difference methods. To demonstrate, we compare different numerical methods against an analytical solution. Consider an exponential density profile—the canonical deep ocean stratification profile which has known analytical solutions for both the nonhydrostatic internal modes (Garrett & Munk, 1972), as well as the SQG modes (LaCasce, 2012). Using a numerical method to solve the $\langle K - \text{constant EVP} \rangle$, we can compute the relative error of the numerical solution to the analytical solution. We define the relative error as

$$\text{rel. err.} = \frac{\max(|f_i - f(z_i)|)}{\max(f(z_i))}$$

where $f(z_i)$ is the true solution evaluated at the grid points $z_i$ and $f_i$ is the numerical approximation. Figure 2 shows the maximum relative error of the eigenmodes $F$, $G$ and eigenvalue $\tilde{F}$ found by solving the $\langle K - \text{constant EVP} \rangle$ using second-order finite difference methods for standard exponential stratification with 64 vertical grid points (blue). Details of the numerical implementation of the analytical solutions are given in Appendix A.
Figure 2. Relative error as a function of vertical mode number using 64 evenly spaced grid points in exponential stratification $N(z) = N_0 e^{z/b}$, where $N_0 = 3$ cph and $b = 1,300$ m at latitude $33^\circ$ in a $5,000$ m deep ocean for $K = 0.0$ (top panel) and $K = \frac{2\pi}{500}$ m (bottom panel). Shown are two second-order finite differencing methods with (1) 64 grid points (blue), (2) 631 grid points (blue dotted), and 3 spectral methods using Chebyshev polynomials with coordinates in (3) depth (red), (4) WKB scaled (purple), and (5) density (orange).

Reasonable error magnitudes are $O(10^{-2})$; however, the top panel of Figure 2 shows that no modes computed with finite differencing satisfy this condition. The situation is even worse for the $K = \frac{2\pi}{500}$ case shown in the bottom panel, where even the lowest mode has an $O(0.1)$ error. The blue dotted line in Figure 2 shows that increasing the resolution tenfold for second-order finite differencing decreases the error by a factor of 100, as would be expected. However, this comes at 1,000 times the computational cost and still barely produces any usable modes.

The accuracy of finite differencing can be increased by going to higher orders (Fornberg, 1998), since the truncation error at order $s$ scales as $O(\Delta z)^s$. The truncation error of the 10th mode in exponential stratification is shown in the top panel of Figure 3 where the blue (solid, dotted) line shows the (second and sixth)-order finite difference method converging at its predicted rate. The bottom panel shows that even with 1,024 grid points, there are only eight usable modes for the second-order finite differencing method, while sixth-order gives up to 50 modes. However, while increasing the order of the method does provide some gains in accuracy, the most efficient way to proceed is simply to use spectral methods, which promise exponentially decreasing truncation error, rather than the polynomial truncation errors offered by finite differencing. When using an analytical density function (dashed lines, Figure 3) rather than gridded data, there is no interpolation error and the spectral methods truncation errors reach a noise floor somewhere between 64 and 128 grid points. Furthermore, the number of usable modes is an order of magnitude higher than even the sixth-order finite difference method. In practical terms, the second-order finite difference method is producing about 10 good modes in 100 s, while the spectral methods are producing about 100 good modes in 1 s. The increase in truncation error at higher resolution is likely due to increasingly compounded errors of the eigenvalue solvers.

Figure 3. The top panel shows relative error as a function of resolution for the 10th mode in exponential stratification with $K = \frac{2\pi}{500}$. The density function is specified on an evenly spaced grid (solid lines) or passed directly as an analytical function (dashed lines). The bottom panel shows the number of usable modes as a function of resolution, defined as the number of modes with truncation errors less than $10^{-2}$. The convergence rates of the second-order and sixth-order finite difference methods are found to be $(\Delta z)^2$ and $(\Delta z)^5$, respectively.
4. Chebyshev Polynomials

Written in matrix form the \((K\)-constant EVP) is

\[
Av = \frac{1}{h} Bv
\]

(20)

where \(v\) is the vector representation of the normal mode \(G\) at grid points in \(z\), \(A = \partial_{zz} - K^2\) and \(B = (f^2_0 - N^2)/g\). For finite differencing, \(A = D_{zz} - K^2I\), where \(D_{zz}\) is the \(N_z \times N_z\) differentiation matrix and \(I\) is the \(N_z\) dimensional identity matrix. To use Chebyshev polynomials, we project vector \(v\) onto a Chebyshev basis using

\[
\hat{v} = T^{-1}v
\]

where \(T\) is the matrix that transforms a vector from a Chebyshev basis to the coordinate basis. In a practical sense, the columns of \(T\) are the Chebyshev polynomials. Then the EVP becomes

\[
ATT^{-1}v = \frac{1}{h} BTT^{-1}v
\]

(21)

or simply

\[
(\partial_{zz}T - K^2T)\hat{v} = \frac{1}{h} \left( f^2_0 - N^2(z) \right) g T\hat{v}
\]

(22)

The vector \(\hat{v}\) contains the coefficients needed to reconstruct eigenfunctions, and \(\partial_{zz}T\) are the second derivatives of the Chebyshev polynomials.

The optimal choice of grid for Chebyshev polynomials is a Gauss-Lobatto grid (Boyd, 2001; Canuto et al., 2006); for example, equation (31) below and thus the eigenmatrices and eigenfunctions are always created on a Gauss-Lobatto grid for any chosen coordinate. Because the basis functions are continuous functions of \(z\), the resulting vertical modes can be interpolated onto any grid at any resolution by evaluating the functions at the points of interest.

It is, however, rarely the case that density is given as an analytical function or that observations are made on a Gauss-Lobatto grid, which means that typically the density needs to be interpolated on the appropriate grid. Interpolation is performed using B-splines implemented with the numerical framework described in Early and Sykulski (2020). The advantage to using B-splines to represent gridded density data is that it is easy to accommodate arbitrary grids. Despite being a low-order method, this is generally not a limitation (see section 6). In the cases shown in Figure 2, the algorithms are given the density \((\rho)\) on a uniform grid in \(z\) of 64 points and the resulting modes are returned on the same grid (except where noted for the high-resolution finite differencing case). This is, of course, suboptimal for the spectral cases which use a Gauss-Lobatto grid on various coordinates to compute the EVP. When given an analytical function for density, these methods perform even better, as can be seen in Figure 3.

Despite the potential limitations imposed by interpolating the density with B-splines onto a Gauss-Lobatto grid, the red line in Figure 2 shows that the Chebyshev method performs extremely well, even while interpolating from an evenly spaced grid and outputting to the same grid. The first 20 and 14 modes have error less than \(O(10^{-2})\) for the \(K = 0\) and \(K = 2\pi/500\text{m}^{-1}\) cases, respectively. However, at higher horizontal resolution (larger wavenumbers \(K\)), even the spectral method’s errors grow large, because the points at which the functions are evaluated do not sufficiently capture the oscillations of the modes. This can be remedied by using a stretched coordinate, \(s\).

4.1. Stretched Coordinates

In order to find an independent coordinate better suited to capturing the structure of the eigenmodes, we rewrite the EVPs in terms of a generic coordinate \(s\) and then consider two concrete examples. Taking \(z\) to be a function of \(s\) and applying the chain rule leads to

\[
\partial_s = (\partial_s z)^{-1} \partial_z
\]

(23)

and

\[
\partial_{zz} = - (\partial_s z)(\partial_s z)^{-1} \partial_s + (\partial_s z)^{-2} \partial_{ss}
\]

(24)
Figure 4. The left panel shows a stratification profile with pycnocline taken from Cushman-Roisin and Beckers (2011). The vertical dashed line represents a frequency with two turning points in the pycnocline. The right panel shows the fourth vertical $F$ mode at that frequency.

Figure 5. Relative error as a function of vertical mode number using 64 evenly spaced grid points for the frequency and stratification shown in Figure 4. Shown is the WKB scaled spectral method (purple) as in Figure 2 and also the adaptive grid method (green) that clusters points near the regions of oscillation. Only the first 16 modes are shown.

For example, the ($K$-constant EVP) becomes

$$
(-\partial_z z)(\partial_z z)^{-3}\partial_z + (\partial_z z)^{-2}\partial_z)G - K^2G = -\frac{N^2 - f_0^2}{gh}G
$$

(25)

where now $F = h(\partial_z z)^{-1}\partial_z G$. The free surface boundary condition in these coordinates becomes $\partial_z G = G/h$, while the normalization conditions are now

$$
\frac{1}{G} \int_{s(D)}^{s(0)} (N^2(s) - f_0^2)G, G, \partial_z z \, ds = \delta_{ij}
$$

(26)

and

$$
\frac{1}{D} \int_{s(D)}^{s(0)} F, F, \partial_z z \, ds = \delta_{ij}
$$

(27)

A necessary condition for using stretched coordinates is that the function $s(z)$ must be strictly monotonic.

4.2. Density Coordinates

For density coordinates $s = -\bar{\rho}/\rho_0$, equation (25) can be written as

$$
N^4\partial_z G + \partial_z (N^2) \partial_z G - K^2G = -\frac{N^2 - f_0^2}{gh}G
$$

(28)

where $F = hN^2\partial_z G$. Note that the derivatives of the density are computed on the $z$ coordinate then projected onto the $s$ coordinate in the EVP. This avoids using inverses of functions that tend towards zero and therefore has greater numerical stability. While the method does well for the high wavenumber case (Figure 2, lower panel), it performs somewhat poorly with a uniform relative error of $O(0.03)$ for all modes in the low wavenumber (upper panel), as shown by the orange line. Evidently, density coordinates cluster points inefficiently in this case.

4.3. WKB Stretched Coordinates

A compromise between depth ($z$) and density ($\bar{\rho}$) coordinates is the WKB stretched coordinate, $s = \int_{D}^{z} N(z') \, dz'$. In this case the EVP becomes

\[
\left( (\partial_z N)\partial_z + N^2\partial_\alpha \right) G - K^2G = \frac{f_0^2 - N^2}{gh}G
\]

(29)

where $F = hN\partial_\alpha G$.

The purple line in Figure 2 shows that the vertical modes computed on WKB coordinates have uniform accuracy of $O(10^{-3})$ for $K = 0$, outperforming the density coordinate case and also performing nearly as well as density coordinates in the high wavenumber case.

4.4. Adaptive Grid for $\omega$-Constant EVP

Solving the ($\omega$-constant EVP) suffers from the additional challenge that as the frequency increases and the distance between turning points decreases, the grid spacing necessary to capture the mode structure becomes ever smaller. As noted in section 1, this issue arises when considering internal waves near a pycnocline. An example stratification profile and vertical mode found at a frequency approaching the maximum frequency in the pycnocline is shown in Figure 4. The relative error as a function of mode for this example is shown in Figure 5, from which it is clear that even the WKB stretched coordinate that performed so well for the ($K$-constant EVP) does relatively poorly in this scenario. Sturm-Liouville theory tells us that the $n$th $F$ mode will have $n$ zero
crossings in the oscillatory region where \( N^2(z) > \omega^2 \) (Arfken, 1970). Thus, in order to resolve these oscillations, one would require at least \( 2n \) optimally placed points in that region, as well as additional points to capture the variance in the decay regions. Simply increasing resolution of the Chebyshev grid cannot efficiently solve the problem, as grid points will continue to be poorly placed.

To resolve this issue, we devise an ad hoc method for clustering points in regions of interest. Our approach is to partition the domain into regions where the modes are hypothesized to be nonzero, formulate the EVP for each region (using WKB stretched coordinates), then couple the equations at the region boundaries. This enables us to assign most of the points to the regions where the solution is assumed to be nonzero, and allocate a few remaining points in the other regions. A comparison of this adaptive method and the standard WKB stretched coordinates is shown in Figure 5, where the adaptive method is able to capture a few more usable modes than the standard WKB stretched coordinate method with one EVP. The value of this method becomes more pronounced as the maximum frequency is reached. The number of usable modes (error \(< 10^{-2}\)) drops to zero as the maximum buoyancy frequency is approached when using the single EVP, as shown in Figure 6. However, using the adaptive grid algorithm, we are able to guarantee a minimum number of usable modes as points cluster around the turning frequencies.

The equation boundaries are established by using the WKB approximated solution to identify the regions where the modes are expected to be nonzero. Specifically, the equation boundaries are the points where WKB solution (A40) decays to \( 10^{-5} \) of its value from the turning point. This is an adjustable tolerance, chosen to be small enough that only a few points are needed to capture the nearly zero function but large enough that the nonzero regions aren’t unnecessarily large. The gray vertical lines in Figure 6 show the number of coupled equations being used to solve the EVP. At the lowest frequencies only one equation is used and the method is identical to the WKB stretched coordinate method described in section 4.3. As the frequency increases, the algorithm eventually separates into two coupled equations: one for the top boundary and pycnocline and another for the deep region where no mode variance is expected (refer to Figure 4). At high enough frequency the region above the pycnocline is decoupled as well, and three coupled EVP problems are solved.

The EVPs are coupled by requiring that the function and its first derivative are continuous at the equation boundaries, following the procedure described in section 22.3 of Boyd (2001). The “eigenvalue rule-of-thumb” as discussed in Boyd (2001) states that roughly \( n/2 \) modes will be accurate when using \( n + 1 \) Chebyshev polynomials away from boundary layers or critical levels. Solving the EVP with turning points near the maximum buoyancy certainly does not satisfy this criterion, but the rule-of-thumb can be modified to use half of the modes with eigenvalues greater than zero. Although we make no attempt at proving the general validity of this modification, the dashed line in Figure 6 indicates that the rule-of-thumb generally does well at predicting how many modes are good quality.
5. Numerical Implementation

One of the primary products of this paper is the implementation of these methods as classes in Matlab (see Appendix C for more details). Figure 7 shows the flowchart followed by the initialization algorithm for the \textit{InternalModes} class, described in this section.

The two methods for initializing the classes are both called using

\[
\text{im} = \text{InternalModes}(\rho, z, z_{\text{Out}}, \text{latitude});
\]

where the arguments \(\rho, z\), \(z_{\text{Out}}\) are either a gridded density field at locations \(z\) or function handle valid in the domain spanned by \([\min(z) \max(z)]\). When the function handle is given, the density function is projected onto Chebyshev polynomials. If gridded data are provided, then the density is interpolated using B-splines. The argument \(z_{\text{Out}}\) specifies the grid on which all output is given, which need not span the full depth.

After initialization, all classes support setting the upper and lower boundary conditions as well as setting the normalization to any of the choices discussed in section 2.4.

The two primary functions for computing internal modes are

\[
[F, G, h, \omega] = \text{im.ModesAtWavenumber}(k);
\]

for the \((K\text{-constant EVP})\) and

\[
[F, G, h, k] = \text{im.ModesAtFrequency}(\omega);
\]

for the \((\omega\text{-constant EVP})\).

The implementation of these methods for finite differencing is straightforward—the EVP is solved either on the gridded input data as given or on a grid that matches the output grid if specified as a function. The differentiation matrices are created using the algorithms described in Fornberg (1998). However, the spectral implementations require additional choices.

The EVP being solved is

\[
[a(s)T_s + b(s)T + c(s)T]v = \frac{1}{h}d(s)Tv
\]  

(30)

where \(s\) is a generic stretched coordinate, \(a(s), b(s), c(s),\) and \(d(s)\) are referred to here as coefficient functions.
The algorithm can be separated into the three parts. First, we compute the coefficient functions for each EVP, for example, \( N^2 \) and \( \partial_z^2 N \) for equation (29). Second, the EVP is solved on the appropriate coordinate with \( n_{\text{evp}} \) points. Finally, the resulting modes are normalized and projected onto the output grid with an arbitrary number of points.

5.1. Initialization With an Analytical Function

If the `InternalModes` class is initialized with a function handle for the density, it is projected onto Chebyshev polynomials which are then used to compute the coefficient functions and, if necessary, the stretched coordinate.

To project onto Chebyshev polynomials, we define a grid with \( n_{\text{grid}} \) points using

\[
\chi_{\text{lobatto}} = \frac{z_{\text{max}} - z_{\text{min}}}{2} \left( \cos \left( \frac{\pi i}{n_{\text{grid}} - 1} \right) + 1 \right) + z_{\text{min}}
\]

where \( i \) is an integer index ranging from 0 to \( n_{\text{grid}} - 1 \). We evaluate the density function on that grid, \( \tilde{\rho}(\chi_{\text{lobatto}}) \).

The density function is then expanded in a Chebyshev polynomial basis such that

\[
\tilde{\rho}(\chi_{\text{lobatto}}) = \sum_{k=0}^{n_{\text{grid}}} \hat{\rho}_k T_k(\chi_{\text{lobatto}})
\]

where \( \hat{\rho}_k \) indicates the \( k \)th coefficient for Chebyshev polynomial defined on \( z \) coordinates. The coefficients for the derivative of the function, denoted \( \hat{\rho}_k^2 \), are then computed using a recursion formula

\[
\hat{c}_k \hat{\rho}_k^2 = \hat{\rho}_k + 2(k + 1) \hat{\rho}_k^1
\]

where \( \hat{c}_k = 2 \) for \( k = 0 \) and \( \hat{c}_k = 1 \) otherwise. Because a Gauss-Lobatto grid was used for the \( z \)-coordinate, the Chebyshev transformation is performed with a rescaled fast Fourier transformation in \( O(n_{\text{grid}} \log n_{\text{grid}}) \) operations. The differentiation requires only \( O(n_{\text{grid}}) \) operations, which means that all of the coefficient functions for the EVP can be computed on a relatively fine grid. For example, \( n_{\text{grid}} = O(2^4) \) takes a fraction of a second on consumer hardware from 2015.

The stretched coordinates implemented here are either \( s = z \), \( s = -g \tilde{\rho}/\rho_0 \), or \( s = \int_{z_{\text{min}}}^{z} N(z') \, dz' \), where the latter two cases require density to be strictly monotonic. For those two cases if \( \tilde{\rho}_z \geq 0 \) anywhere in the domain, then an error is thrown. For the WKB coordinate, \( s = \int_{z_{\text{min}}}^{z} N(z') \, dz' \), the integral is computed spectrally using equation (31).

5.2. Initialization With Gridded Data

In the more typical scenario where a user initializes the `InternalModes` class with gridded data from observations or a numerical model, B-splines are used to interpolate the data and compute the coefficient functions. The primary advantage to using B-splines in this scenario is that B-splines can be created for arbitrary grids without suffering from Runge’s phenomena at lower orders. We fit the data to sixth-order interpolating spline (with five nonzero derivatives) using the methodology and numerical implementation described in Early and Sykulski (2020).

If the method requires that density remain monotonic (e.g., for WKB and density coordinates), then the B-spline fits are constrained to be monotonic following Pya and Wood (2015). If the data are not monotonic, then this implicitly smooths to find the nearest monotonic fit in a least squares sense.

Computing the stretched coordinate and the coefficient functions from the spline interpolant requires derivatives and integrals of the B-splines, which are relatively straightforward to compute because they are just piecewise polynomials (De Boor, 1978). The WKB method requires computing the square root of the B-spline interpolant. The approach taken here is to build a new interpolating spline of the same order that interpolates between the square root of the data points.

5.3. EVP

All three Chebyshev methods solve their respective EVPs on a Gauss-Lobatto grid of their respective coordinate, that is, \( s = z \), \( s = -g \tilde{\rho}/\rho_0 \), or \( s = \int_{z_{\text{min}}}^{z} N(z') \, dz' \). The Gauss-Lobatto grid in \( s \) is defined in the usual way as

\[
S_{\text{lobatto}} = \frac{s_{\text{max}} - s_{\text{min}}}{2} \left( \cos \left( \frac{\pi \xi}{n_{\text{exp}} - 1} \right) + 1 \right) + s_{\text{min}}
\]
where the number of points \( n_{\text{evp}} \) reflects the size of the EVP and is therefore also an absolute upper bound to the number of modes that may be computed.

Once the Gauss-Lobatto grid \( \delta_{\text{lobatto}} \) is created, the corresponding value \( z(\delta_{\text{lobatto}}) \) is computed using the bisection method as implemented in Driscoll et al. (2014). The method is set to terminate with a relative error of \( O(10^{-12}) \).

The coefficient functions in equation (30) are now simply evaluated onto the \( \delta_{\text{lobatto}} \) grid using the interpolant (either a B-spline or Chebyshev). The Chebyshev polynomials and their derivatives \( (T, T_s, \text{and } T_{ss}) \) are computed using the standard recursion formulas and then multiplied by the coefficient functions to create eigenvalue matrices. The boundary conditions are implemented by replacing the first and last rows of the matrices \( A \) and \( B \) in (20).

The EVP is then solved using the standard generalized EVP solver. This is typically the rate limiting step in the process, taking \( O(n_{\text{evp}}^3) \) operations. The resulting eigenvectors now contain coefficients to the Chebyshev polynomials defined on the stretched coordinate.

### 5.4. Adaptive Grid

The adaptive grid is created by locating regions in the domain where the solution is expected to be small and then allocating fewer points (and therefore fewer Chebyshev polynomials) to those regions. After identifying the turning points \( z_T \) where \( N^2(z_T) = \omega^2 \), the WKB solution (A40) is used to identify the equation boundaries \( z_{\text{end}} \), the points where \( F^{WKB}(z_{\text{end}}, \omega)/F^{WKB}(z_T, \omega) = 10^{-5} \). The WKB solution is assumed to work locally in the stratification and is therefore applied at the turning point, \( z_T \), in the direction of decaying variance. The eigenvalue (A42) is assumed to be set globally by integration over all oscillatory regions. If \( m \) equation boundaries \( z_{\text{end}} \) are identified, they delineate \( m + 1 \) regions: the “decay” regions, where the solution is anticipated to be small and governed by the decaying exponential and “oscillatory” regions where the solution is expected to dominate and include the oscillatory solution. These decaying and oscillating regions are necessarily alternating.

The \( m+1 \) regions are coupled using the same technique described in section 22.3 of Boyd (2001) by requiring continuity across boundaries for \( G \) and \( \partial_z G \). The key benefit to this algorithm is that the decay regions are allocated as few as 6 points each, while oscillatory regions are apportioned the remaining points relative to their WKB length, \( L^m = \int N(z')dz' \). While we find that 6 points appear to be sufficient for the decay regions, in practice, we do in fact apportion 1/16th of the total points evenly between the decay regions as a hedge that this ad hoc method will fail for some unforeseen cases.

The adaptive grid algorithms use low-order interpolation to identify \( z_T \) and \( z_{\text{end}} \) because high accuracy is not required, and this keeps the number of computations \( O(n_{\text{grid}}) \).

### 5.5. Normalization

The final step is to normalize the resulting eigenvectors and project them onto \( z_{\text{out}} \). Normalization using the two integral conditions can be performed exactly invoking the fact that the integral of each Chebyshev polynomial \( T^k(z) \) is exactly known

\[
I_k = \int_{-1}^{1} T^k(z) \, dz = \begin{cases} \frac{(-1)^{k+1} - 1}{1-k^2} & k \neq 1 \\ 0 & k = 1 \end{cases}.
\] (35)

We have defined \( I_k \) such that it can be summed with a vector of Chebyshev coefficients to produce the integral. In other words, if \( I_k \) is a Chebyshev coefficient vector, then \( I = \sum I_k I_k \) is the definite integral. The integrands in equation (26) and (27) are computed pseudospectrally before integrating (by transforming to the spatial domain, multiplying, then transforming back Chebyshev coefficients).

Computing the max U and and max W norm is more problematic because the function extrema do not necessarily lie on the grid points. For the implementation, we simply take the maximum at the resolved grid points, but if higher accuracy is required, one could locate the extrema using the methods in Boyd (2014).

Finally, the normalized eigenmodes are projected onto the output grid using the slow Chebyshev transforms,

\[
v(z_{\text{out}}) = \sum_{k=0}^{n_{\text{evp}}} I_k T^k(s(z_{\text{out}}))
\] (36)
If a large number of output points are requested, this operation could dominate the total computation time. The algorithm flowchart for the mode computation is shown in Figure 8.

5.6. SQG Modes
The two functions for computing the SQG modes are

\[ \psi = \text{im.SurfaceModesAtWavenumber}(k); \]

and

\[ \psi = \text{im.BottomModesAtWavenumber}(k); \]

where \( k \) is an array of wavenumbers.

The SQG modes are found from equation (B2) using a linear solver after replacing the top and bottom points of the matrix with the boundary conditions. As noted in Tulloch and Smith (2009a), the SQG modes require a high density of grid points near the boundaries, a task well suited to the Gauss-Lobatto grid in equation (31). The number of Chebyshev polynomials is chosen so that the Gauss-Lobatto grid captures at least 10 points over the \( e \)-folding scale. The \( e \)-fold scale for constant stratification (see Appendix B.1) is

\[ \Delta z_{\text{fold}} = \frac{f_0}{K N_0}, \]

and the distance between the first two points in a Lobatto grid, equation (31), is

\[ \Delta z_{\text{boundary}} = \frac{D}{4} \left( \frac{\pi}{(n_{\text{grid}} - 1)} \right)^2. \]

(37)

where we have defined \( D = z_{\text{max}} - z_{\text{min}} \) as the depth of the domain. Setting \( \Delta z_{\text{boundary}} = \frac{1}{10} \Delta z_{\text{fold}} \), we find that we need

\[ n_{\text{grid}} = 1 + \frac{\pi}{2} \sqrt{\frac{10DKN_0}{f_0}} \]

(38)

points (and therefore also \( n_{\text{grid}} \) polynomials) to sufficiently capture the SQG mode. The resulting SQG modes are projected onto the output grid using equation (36).

5.7. Unit Testing
In order to ensure that each of the algorithm implementations is correct, the numerically generated eigenmodes and eigenvalues are compared against the analytical solutions for constant stratification and exponential stratification shown in Appendix A. The comparison is performed across a range of wavenumbers and frequencies for both surface boundary conditions and all four norms.

The computed SQG modes are also compared against analytical solutions for constant and exponential stratifications, shown in Appendix B.
6. Other Sources of Error

In section 1 we noted five sources of errors that contribute to the total error when computing the forward or inverse transformation, but this manuscript has primarily focused on one source of error: the numerics of accurately representing the vertical modes in the EVP. We now discuss these other sources of error and how they are dealt with in the numerical implementation.

6.1. Aliasing Error

When performing a forward transformation, where a given dynamical field is projected onto the vertical modes, the data grid will determine how many modes are resolvable. As with a Fourier transformation, higher-frequency modes alias into the lower-frequency modes. Unlike the Fourier transformation, however, the optimal grid for performing a transformation is not an evenly spaced grid but depends on the stratification profile, and therefore, the EVP being solved. Here we show that there is a relatively easy method for determining the number of resolvable modes for a given grid using the condition number of the resulting matrix.

To show the effect of different grid choices on the forward transformation, we use the analytical solution of the vertical modes in exponential stratification in combination with an imposed spectrum to generate stochastic isopycnal displacement profiles typical of the world oceans. In particular, we use the Garrett-Munk spectrum.

\[ H(j) = \frac{H_0}{(j + J)^p} \]  

where \( j \) is the roll-off mode, usually set to 3 but possibly as high as 20, \( p \) is the slope which is usually very nearly 5/2, and \( H_0 \) normalizes sum over \( j = 1..\infty \) to unity. For each stochastically generated set of coefficients, \( m^j \), a profile \( \eta(z) = \sum_{j=1}^{N} G'(z)m^j \) is created. The profile is then evaluated on three different grids: \( z_{even}^j \), \( z_{lobatto}^j \), and \( z_{quadrature}^j \), where \( z_{quadrature}^j \) is the grid of Gaussian quadrature points, determined by the roots of a mode one higher than is trying to be used (Boyd, 2001; Press et al., 1992). Using the first \( n \) modes, we then attempt to recover the coefficients using least squares—in practice this is Matlab’s \texttt{mldivide} (\( \setminus \)) operator. For example, the first \( n \) coefficients of the evenly spaced grid are determined by

\[ \hat{m}^j_{even} = G'(z_{even}^j) \setminus \eta(z_{even}^j) \]  

where \( j = 1..n \). The root-mean-square error (rmse) is defined as the error in the sum of recovered and missing coefficients

\[ \text{rmse}^2 \equiv \frac{\sum_{j=1}^{n} (m^j - \hat{m}^j)^2 + \sum_{j=n+1}^{N} (m^j)^2}{\sum_{j=1}^{N} (m^j)^2} \]  

Figure 9 shows the result of trying to recover the mode coefficients, \( \hat{m}^j \) using successively more modes for the three different grids. In all three cases the rmse decreases as modes are added until a dramatic increase occurs, correlating with a similarly dramatic increase in the condition number of the matrix. The quadrature grid, defined as the roots of the \( G^{N-1} \) mode plus the boundaries, performs best, as expected. Including the boundaries in this definition means that the top and bottom boundary points provide no useful information, and therefore, only \( N - 2 \) modes are recoverable for \( G \) and \( N - 1 \) for \( F \) with a rigid lid. This definition is chosen so that the forward and inverse transform for constant stratification coincides with the discrete sine and cosine transform (and their associated grids).

While the condition number of the matrix is clearly controlled by the grid being used, the choice of norm also affects the condition number. Generally speaking, the \( K \)-constant norm performs well for transformations with the \( G \) modes and the \( \omega \)-constant norm with the \( F \) modes. The exception to this is when the free surface boundary condition is used, the barotropic mode has a substantially different \( L^2 \) norm and should be rescaled.

6.2. Mode Error

Here, we test whether or not the truncation error in the vertical modes is a limitation in recovering the coefficients of the spectrum, \( m^j \). To this end, we created profiles on a quadrature grid with variance distributed using \( H(j) \) for a range of parameters including white noise (large \( j \)), and very smooth (large \( p \)), as described...
Figure 9. The top panel shows the root-mean-square error as a function of total modes used to recover the coefficients with an inverse transformation. Vertical dashed lines are the predicted cutoffs for the different grids, based on the matrix condition number at which the modes are no longer resolvable. The condition number as a function of total modes is shown in the bottom panel.

Figure 9. The top panel shows the root-mean-square error as a function of total modes used to recover the coefficients with an inverse transformation. Vertical dashed lines are the predicted cutoffs for the different grids, based on the matrix condition number at which the modes are no longer resolvable. The condition number as a function of total modes is shown in the bottom panel.

above, and recorded how many mode coefficients were recoverable for some error tolerance across all the different numerical methods in this paper. We found that discarding modes with truncation errors exceeding the requested error tolerance of the modes guarantee coefficients recovered with the requested error tolerance. In other words, if one wants relative errors in coefficients of less than $10^{-2}$, one needs modes with errors less than $10^{-2}$. The only exception is for very steep spectra (e.g., $p = -10$), where the coefficients of the higher modes are indistinguishable from zero. The reverse of this is not true—in fact, including modes with truncation errors exceeding the error tolerance can often return coefficients within the bounds of the error tolerance. Evidently, the errors in these ordered, orthogonal bases work systematically in our favor.

6.3. Interpolation Error

Another source of error may arise from interpolation of the density function. This issue is treated separately from a poorly defined or noisy mean density function and therefore assumes that the data given are gridded and without error. For gridded data, our method uses a relatively low order B-spline to interpolate between grid points which is then evaluated for the coefficient functions where needed. Does this low-order interpolation method limit the mode recoverability described in the previous section?

To address this question, we compute the vertical modes for exponential stratification from an evenly spaced density function with variable number of grid points. These modes are then used to recover the coefficients $\tilde{m}_j$, as above, on a high-resolution quadrature grid. We find that with as few as 16 grid points for the density function, we are able to recover 90 mode coefficients with less than 1% error.

6.4. Poorly Defined or Noisy Mean Density

It is often the case that the mean density function, $\bar{\rho}(z)$, is not easily defined. Averaging over a mooring time series of density will not necessarily result in a monotonic density function—and averaging often removes the sharp gradients that exist in individual profiles, which may not be desirable. Even output from numerical models can suffer from these same issues, depending on the boundary conditions. Noisy data, where errors in the observed value of $\bar{\rho}(z)$ stem from instrument errors, also effectively constitute a poorly defined mean.

One way to frame these issues is to ask how a misspecified mean density affects our ability to infer the vertical spectrum of a given flow.

First, we note that all methods, as implemented, will proceed without error for noisy data. However, the most notable difference between methods is that the WKB and density coordinate methods use a density function constrained to the nearest monotonic spline fit, as previously described. It is not clear that this implicit smoothing is necessarily "better" than using the unaltered density function with the $z$-coordinate method for noisy data. This decision, and how to deal with measurement noise in general, is beyond the scope of this manuscript. Additional smoothing of the density data can be done using many techniques, including the constrained smoothing splines described in Early and Sykulski (2020).

In order to test the effects of misspecifying a mean density profile, we generate density profiles in exponential stratification that follow the GM spectrum, as above, but we attempt to recover the coefficients using vertical modes computed from a noisy mean density profile. The results are consistent with section 6.2. For example, as long as the modes from the noisy profile have errors less than $10^{-2}$ relative to the modes generated from the true profile, the recovered coefficients $\tilde{m}_j$ will have errors less than $10^{-2}$ relative to $m_j$. The relative error of the vertical modes increases as a function of mode number until eventually the mode errors and therefore coefficient errors reach $O(1)$. Exactly where mode errors reach $O(1)$ depends on the details of the noise, or misspecification, of the mean density profile.

While accurately recovering the coefficients $m_j$ of the spectrum becomes impossible with a noisy mean density profile, we are able to accurately infer the spectrum from which $m_j$ was generated by either ensemble
averaging over additional synthetic profiles or bin averaging nearby modes. One way to see why this might be true is to note that the coefficient errors never exceed $O(1)$—so although the exact coefficient is incorrect, the magnitude is correct on average. In practice, using modes from the misspecified mean density profile causes variance that should be associated with mode $j = 33$, for example, to be assigned to the variance of nearby modes. That the spectrum is recoverable despite a noisy mean is consistent with previous analysis methods. In Polzin and Lvov (2011), internal wave spectra are found using WKB approximated modes and a WKB stretched grid. It is also important to note that computing the spectrum with a misspecified mean density function still requires an orthogonal set of modes, and therefore, fast and accurate mode computation is still helpful.

7. Discussion

The spectral methods presented here solve the most relevant forms of the vertical eigenvalue and surface quasigeostrophic mode problems efficiently and accurately. The methods also include an algorithm for computing modes in challenging stratification profiles at high frequencies near turning points. The algorithms are implemented in a publicly available Matlab suite (https://github.com/JeffreyEarly/GLOceanKit/) using the class hierarchy described in Appendix C, and the implementations are validated against known analytical solutions, under a wide range of conditions. However, the methods do not always perform well under all conditions.

Poorly resolved features and discontinuities in the density profile will produce the Gibbs phenomenon, where “ringing” occurs in the vicinity of the discontinuity. In one example, we found that a narrow 5-m-wide pycnocline in a 5,000-m-deep ocean produced strong spurious oscillations unless the pycnocline was sufficiently well resolved with enough grid points. In another example, we defined an analytical profile with a discontinuity in $\tilde{\rho}_{zz}$ (the highest derivative used in the EVP) and this also produced the Gibbs phenomenon. Interestingly, in these cases lower-order finite differencing produced better modes for the same vertical resolution, because these methods implicitly smooth the derivatives. A logical extension of this work would be to apply the “splitting” algorithms in chebfun (Driscoll et al., 2014) to handle such discontinuities.

Our recommendations for projecting observed or modeled fields onto the modes are as follows:

- When possible, use a quadrature grid for the fields and modes, for example, the function GaussQuadaturePointsForModesAtWavenumber in the InternalModesSpectral class computes the Gauss quadrature points for $G$. This is a relatively expensive operation (it involves solving the EVP) but provides near-optimal point placement.
- In the usual case where there is no freedom to choose grid points, compute the condition number as a function of mode as described in section 6.1 and limit the number of modes to a low condition number. Alternatively, if computation time is not a limitation, one can perform the least squares fit of the fields to successively more modes until the coefficients are no longer stable.

Many of the errors described in section 6 may still be a concern but may be quantified with some of the techniques in section 6, by using the density function and spectrum specific to the problem.

Appendix A: Internal Mode Solutions

We present analytical solutions for the internal mode eigenvalue problem in three different scenarios: constant stratification, exponential stratification, and the WKB approximated solution for arbitrary stratification. These solutions are used to validate the numerical implementations.

A.1. Constant Stratification

The internal baroclinic modes in constant stratification are given as

\[ G_j^{\text{const}}(z) = A \sin \left( m_j (z + D) \right) \]

\[ F_j^{\text{const}}(z) = Ah_j m_j \cos \left( m_j (z + D) \right) \]

with eigendepth $h_j$ and vertical wavenumber $m_j$ given by

\[ m_j = \frac{j \pi}{D} + \frac{\xi}{D} \]
Table A1
Inverse of the Normalization Constants ($A^{-1}$) for All Combinations of Solutions and Norms Considered Here

|                          | Trigonometric | Linear  | Hyperbolic |
|--------------------------|---------------|---------|------------|
| $K$-constant              | $(-1)^j \sqrt{\sin^2 (m_j D) + \frac{(N_0^2 - f_0^2) D}{2g} \left(1 - \frac{\sin(2m_j D)}{2m_j D}\right)}$ | $D \sqrt{1 + \frac{(N_0^2 - f_0^2) D}{2g} \left(1 - \frac{\sin(2m_j D)}{2m_j D}\right)}$ | $\sqrt{\sinh^2 (m_0 D) + \frac{(N_0^2 - f_0^2) D}{2g} \left(1 - \frac{\sinh(2m_0 D)}{2m_0 D}\right)}$ |
| $\omega$-constant         | $(-1)^j h_j m_j \sqrt{\frac{1}{2} + \frac{\sin(2m_j D)}{4m_j D}}$ | $D$ | $h_j m_0 \sqrt{\frac{\sinh(2m_0 D)}{4m_0 D}} + \frac{1}{2}$ |
| $U_{max}$                 | $(-1)^j h_j m_j$ | $D$ | $h_j m_0 \cosh(m_0 D)$ |
| $W_{max}$                 | $(-1)^j \sin(m_0 D)\text{for } j = 0$ | $D$ | $\sinh(m_0 D)$ |

where we have assumed that $w = 0$ at the lower boundary $z = -D$. In the case of a rigid lid ($w = 0$ at $z = 0$), the correction to the vertical wavenumber is $\xi = 0$. However, if the linearly approximated free surface boundary condition is used, $h_j G_j(0) = \tilde{G}_j(0)$, then $\xi$ is nonzero. The equations for $\xi$ are transcendental and are therefore solved with a numerical root finding algorithm. The equations for $\xi$ are written in a form conducive for finding the desired root:

- For fixed wavenumber, $k$, the vertical wavenumber correction $\xi$ is found by solving
  \[
  (\xi + j\pi) \left(N_0^2 - f_0^2\right) D \cos(\xi) - g \left(k^2 D^2 + (\xi + j\pi)^2\right) \sin(\xi) = 0
  \]
  (A4)

near $\xi = 0$ and the eigendepth $h_j$ is given by
  \[
  h_j = \frac{N_0^2 - f_0^2}{g k^2 + m_j^2}
  \]
  (A5)

- For fixed frequency, $\omega$, the vertical wavenumber correction $\xi$ is found by solving
  \[
  D(N_0^2 - \omega^2) - g(\xi + j\pi) \tan(\xi) = 0
  \]
  (A6)

near $\xi = 0$ and the eigendepth $h_j$ is given by
  \[
  h_j = \frac{N_0^2 - \omega^2}{g m_j^2}
  \]
  (A7)

The normalization for these modes is given in the first column of Table A1.

In the case of the free surface boundary condition, there also exists a barotropic mode ($j = 0$), the solution of which changes from trigonometric to hyperbolic at $\omega = N_0$, or $k = k_*$, where
  \[
  k_* \equiv \sqrt{\frac{N_0^2 - f_0^2}{g D}}
  \]
  (A8)

In these cases then, the vertical wavenumber reduces to $m_0 = \xi/D$.

- Trigonometric case, $k < k_*$ or $\omega < N_0$
  The solution is exactly the same as the baroclinic solutions given above in equations (A1) and (A2), but now equation (A4) is solved when $j = 0$. As a practical matter, it is numerically more stable to solve
    \[
    (N_0^2 - f_0^2) D - g k_*^{-1} \left(k_*^2 D^2 + \xi^2\right) \tan(\xi) = 0
    \]
    (A9)
  for the positive root near $\xi = \sqrt{(N_0^2 - f_0^2) D/g - k_*^2 D^2}$. For fixed frequency, equation (A6) can be used to find the root near $\xi = \sqrt{D(N_0^2 - \omega^2)/g}$.
- Linear case, $k = k_*$ or $\omega = N_0$
  The solution is given by
    \[
    G_0^{\text{const}}(z) = A(z + D)
    \]
    (A10)
    \[
    F_0^{\text{const}}(z) = AD
    \]
    (A11)

where $h_0 = D$. 
• Hyperbolic case, \( k > k_0 \) or \( \omega > N_0 \) The solution is given by
\[
G_j(z) = A \sinh (m_0(z + D)) 
\]
(12)
\[
F_j(z) = Ah_0m_0 \cosh (m_0(z + D)) 
\]
(13)

– For fixed wavenumber, \( k \), the vertical wavenumber correction \( \xi \) is found by solving
\[
(N_0^2 - f_0^2) D - g\xi^{-1} (k^2D^2 - \xi^2) \tanh(\xi) = 0
\]
(14)
for the root near \( \xi = \sqrt{k^2D^2 - (N_0^2 - f_0^2)D/g} \) and the eigendepth \( h_0 \) is given by
\[
h_0 = \frac{1}{g} \frac{N_0^2 - f_0^2}{k^2 + m_0^2} 
\]
(15)

– For fixed frequency, \( \omega \), the vertical wavenumber correction \( \xi \) is found by solving
\[
(\omega^2 - N_0^2) D - g\xi \tanh(\xi) = 0
\]
(16)
for the root near \( \xi = \sqrt{D(\omega^2 - N_0^2)/g} \) and the eigendepth \( h_0 \) is given by
\[
h_0 = \frac{1}{g} \frac{\omega^2 - N_0^2}{m_0^2} 
\]
(17)

### A.2. Exponential Stratification

Exponential stratification serves as the \( O(1) \) representation of the ocean stratification away from the poles and continental boundaries. As formulated and first solved in Garrett and Munk (1972), we take the stratification to be \( N^2 = N_0^2 e^{2z/b} \), where \( N_0 \) is buoyancy frequency and \( b \) is the thermocline depth (with canonical values of three cycles per hour and 1,300 m).

Letting the stretched coordinate \( s = N_0 e^{z/b} \) as in section 4.1, the \( (\omega\text{-constant EVP}) \) becomes
\[
s^2G_{ss} + sG_s + \frac{b^2}{c_s^2} (s^2 - \omega^2) G = 0 
\]
(18)
which has solution
\[
G_j^{\text{exp}}(z) = J_v \left( \frac{bN_0}{c_j} e^{z/b} \right) - \alpha_j Y_v \left( \frac{bN_0}{c_j} e^{z/b} \right) 
\]
(19)
where
\[
\alpha_j \equiv J_v \left( \frac{bN_0}{c_j} e^{-D/b} \right) /Y_v \left( \frac{bN_0}{c_j} e^{-D/b} \right) 
\]
(20)
is chosen to satisfy the lower boundary condition \( G(-D) = 0 \). The Bessel function \( Y_v(s) \) has a singularity at \( s = 0 \), so for many choices of \( \omega \) and \( c_j, \alpha_j \ll 1 \) and the second term needs to be neglected for stable numerical evaluation. The order of the Bessel function is set by the frequency \( v = \frac{bN_0}{c_j} \), or, using the dispersion relation wavenumber \( v = \sqrt{\frac{\omega^2}{c_s^2 + \omega^2}} \), the \( F \) modes are found by taking the derivative
\[
F_j^{\text{exp}}(z) = \frac{N_0}{2g} e^{z/b} \left[ J_{v-1} \left( \frac{bN_0}{c_j} e^{z/b} \right) - J_{v+1} \left( \frac{bN_0}{c_j} e^{z/b} \right) + \alpha_j \left( Y_{v-1} \left( \frac{bN_0}{c_j} e^{z/b} \right) - Y_{v+1} \left( \frac{bN_0}{c_j} e^{z/b} \right) \right) \right] 
\]
(21)
The discretization into modes is a result of applying the second boundary condition, in this case either a rigid lid \( G(0) = 0 \) or a free surface \( G(0) = F(0) \) and then finding the eigenmode speeds \( c_j = \sqrt{g h_0} \) that satisfy those conditions. The \( c_j \)’s are therefore determined by the roots of Bessel functions, for which there is no general closed form solution. The challenge then becomes finding bounds for the roots and writing the equation in a form suitable for a root finding algorithm.
In practice, it is easiest to find the inverse of the \( c_s \), so we let \( x = \frac{\hbar N_0}{c_s} \) and then write the root equation, \( f(x) = 0 \), in terms of parameter function \( \nu(x) \) and \( s(x) \). Again, stable numerical evaluation requires that we work around the singularity of \( Y_n \), so if \( \nu(x) < s(x)e^{-b/x} \), then we take

\[
f_{\text{small}}(x) = A(x)J_{\nu(x)}(e^{-b/x}s(x)) + B(x)Y_{\nu(x)}(e^{-b/x}s(x))
\]  
(A22)

and when \( \nu(x) > s(x)e^{-b/x} \) we take

\[
f_{\text{big}}(x) = A(x)J_{\nu(x)}(e^{-b/x}s(x))/Y_{\nu(x)}(e^{-b/x}s(x)) + B(x)
\]  
(A23)

where in either case the rigid-lid condition requires

\[
A(x) = Y_{\nu(x)}(s(x))
\]  
(A24)

\[
B(x) = -J_{\nu(x)}(s(x))
\]  
(A25)

and the free surface requires

\[
A(x) = Y_{\nu(x)}(s(x)) - \frac{\alpha}{s(x)}(Y_{\nu(x)-1}(s(x)) - Y_{\nu(x)+1}(s(x)))
\]  
(A26)

\[
B(x) = -J_{\nu(x)}(s(x)) + \frac{\alpha}{s(x)}(Y_{\nu(x)-1}(s(x)) - Y_{\nu(x)+1}(s(x)))
\]  
(A27)

• \( \omega \)-constant solution

The parameter functions are defined as

\[
\nu(x) = \frac{\omega x}{\eta}
\]  
(A28)

\[
s(x) = \frac{N_0 x}{\eta}
\]  
(A29)

where the scaling factor \( \eta \) is chosen from the WKB approximated solution (Desaubies, 1973),

\(-\omega > N_0 e^{-b/x}\)

The root equation must be taken to be \( f_{\text{big}}(x) \) and

\[
\eta \pi = \sqrt{N_0^2 - \omega^2 - \omega \cos^{-1} \left( \frac{\omega}{N_0} \right)}
\]  
(A30)

(from Desaubies, 1973, equation 2.18); the first \( n \) modes are found within \([0.5, n+1]\).

–otherwise

The root equation must be taken to be \( f_{\text{small}}(x) \) and

\[
\eta \pi = \sqrt{N_0^2 - \omega^2 - \omega \cos^{-1} \left( \frac{\omega}{N_0} \right)}
\]  
(A31)

\[
-\sqrt{N_0^2 e^{-2b/x} - \omega^2 + \omega \cos^{-1} \left( \frac{\omega e^{-b/x}}{N_0} \right)}
\]

(from Desaubies, 1973 equation 2.19); the first \( n \) modes are found within \([0.5, n+1]\).

• \( K \)-constant solution

Unlike the \( \omega \)-constant solution, the order of the Bessel function changes for each root, and therefore, the root equation being used may have to change during evaluation.

The parameter functions are defined as

\[
\nu(x) = \sqrt{\frac{\omega^2 x^2}{\epsilon^2} + \lambda^2}
\]  
(A32)

\[
s(x) = x
\]  
(A33)

where \( \epsilon = \frac{\omega}{N_0} \) and \( \lambda = bk \). Desaubies (1973) provides low-frequency (lf) and high-frequency (hf) bounds for the eigenvalues, which we can be written in terms of wavenumber

\[
x_{\text{lf}}(j) = \left( j - \frac{1}{4} \right) \pi + \lambda \frac{\pi}{2}
\]  
(A34)
We transition at

\[ x_i = \frac{\lambda}{\sqrt{S^2 e^{2D/b} - \epsilon^2}} \]  

(A36)

To set the search bounds for the root algorithm, we want

\[ x_{\text{lower}} = \begin{cases} 
  x_{\text{lf}}(1) & \lambda < 2(1 - \frac{1}{4}) \cdot 10^{-1} \\
  x_{\text{hf}}(1) & \text{otherwise} 
\end{cases} \]  

(A37)

and

\[ x_{\text{upper}} = \begin{cases} 
  x_{\text{lf}}(1.1n) & \lambda < n - \frac{1}{4} \\
  x_{\text{hf}}(5n) & \text{otherwise} 
\end{cases} \]  

(A38)

so the first \( n \) modes are found within \([x_{\text{lower}}, x_{\text{upper}}]\).

Normalization is performed by direct evaluation of the mode functions for the \( U_{\max} \) and \( W_{\max} \) and by numerical integration for the \( K \)-constant norm and \( \omega \)-constant norm norms.

**A.3. WKB Solution**

Desaubies (1973) found the WKB solution for stratification with at most one turning point \( z_T \). Simplified and using the notation of this manuscript, the WKB solution with single turning point \( z_T \) is

\[ G_{j}^{\text{WKB}}(z, \omega) = A \sqrt{\pi} \left( \frac{\xi}{\omega^2 - N^2} \right)^{\frac{i}{4}} Ai(\xi) \]  

(A39)

\[ F_{j}^{\text{WKB}}(z, \omega) = A \sqrt{\pi} \left( \frac{\xi}{\omega^2 - N^2} \right)^{\frac{i}{4}} \xi \omega Ai'(\xi) \]  

(A40)

where \( \xi = \text{sgn}(\omega^2 - N^2) \left( \frac{2}{3} q \right)^{\frac{i}{2}} \) and we have defined

\[ q(z) = \frac{1}{\sqrt{gh_j}} \left| \int_{z_T}^{z} \sqrt{|N^2 - \omega^2|} \, dz \right| \]  

(A41)

so that it goes to zero at the turning point but is positive everywhere else. The eigenvalue is proportional to the integral of the stratification over the oscillatory region

\[ \sqrt{gh_j} = \frac{1}{\left( j - \frac{1}{2} \right) \pi} \left| \int_{z_T}^{z} \sqrt{|N^2 - \omega^2|} \, dz \right| \]  

(A42)

while normalization coefficient for the \( K \)-constant norm is

\[ A(\omega) = \left( -1 \right)^{j} \sqrt{2} \left[ \int_{z_T}^{z} \frac{N^2 - \omega^2}{\sqrt{N^2 - \omega^2}} \, dz \right]^{-\frac{1}{2}} \]  

(A43)

It is useful to note that away from the turning point, the Airy function \( Ai(\xi) \) can be approximated as sinusoidal above the turning point and exponentially decaying below the turning point

\[ G_{j}^{\text{WKB}}(z, \omega) = \frac{1}{\sqrt{2}} \frac{A(\omega)}{\sqrt{|N^2 - \omega^2|}} \begin{cases} 
  \frac{1}{\sqrt{2}} e^{-q}, & \text{for } z > z_T \\
  \sin q + \cos q, & \text{for } z < z_T 
\end{cases} \]  

(A44)
In the case of no turning point \((\omega < N(z), \forall z \in [-D, 0])\) the WKB solution is
\[
G^{\text{WKB}}_{j}(z, \omega) = \frac{A(\omega)}{\sqrt{N^2 - \omega^2}} \sin q
\]
(A45)

\[
F^{\text{WKB}}_{j}(z, \omega) = -\frac{A(\omega)N^2 h_j}{4(\omega^2 - N^2)^{3/2}} \sin q
\]
(A46)

\[
+ A(\omega) \frac{h_j}{g} \sqrt{N^2 - \omega^2} \cos q
\]
(A47)

where we have taken \(z_T = -D\) in (A41). The eigenvalue is now
\[
\sqrt{gh_j} = \frac{1}{j \pi} \int_{-D}^{0} \sqrt{|N^2 - \omega^2|} \, dz
\]
(A48)

Note that this solution does reduce to the exact solution for constant stratification.

**Appendix B: SQG Mode Solutions**

The SQG modes are computed by taking the Fourier transform of \(\psi\) in equation (11) and scaling by the Fourier transformed boundary condition \(\tilde{\rho}(k, l, z = 0)\) such that \(\psi(x, y, z) = \int \int \left( \frac{-g}{N} \tilde{\rho} \right) \phi(k, l, z) e^{i(kx + ly)} \, dk \, dl\). Equation (11) then becomes
\[
0 = -K^2 \phi + \frac{\partial}{\partial z} \left( \frac{f_0^2}{N^2} \frac{\partial \phi}{\partial z} \right)
\]
(B1)

with surface boundary condition \(f_0 \partial_z \phi = 1\) and bottom boundary boundary condition \(\partial_z \phi = 0\) (and vice versa for the bottom boundary modes).

To solve this problem numerically, we project \(\phi\) onto a Chebyshev basis where \(v = \tilde{T} \tilde{v}\) is the vector representation of \(\phi\). The equation for the SQG modes becomes
\[
\left[ N^2 \partial_z T - \partial_z (N^2) \partial_z T - \frac{K^2 N^4}{f_0} T \right] \tilde{\psi} = 0
\]
(B2)

in the interior with boundary conditions \(f_0 \partial_z T \tilde{\psi} = 1\) and \(f_0 \partial_z T \tilde{\psi} = 0\) for the surface mode.

The numerical implementation of equation (B2) is validated against the known analytical solutions.

**B.1. Constant Stratification**

The surface quasigeostrophic modes for constant stratification can be found in Tulloch and Smith (2009b). For the upper boundary this is
\[
f_0 \phi^\text{const}_{\text{sur}}(z) = \frac{1}{\lambda} \frac{\cosh (\lambda(z + D))}{\sinh (\lambda D)}
\]
(B3)

and the lower boundary
\[
f_0 \phi^\text{const}_{\text{bot}}(z) = -\frac{1}{\lambda} \frac{\cosh (\lambda z)}{\sinh (\lambda D)}
\]
(B4)

where \(\lambda = \frac{KN}{f_0}\). These solutions cannot be evaluated numerically for all wavenumbers and depths because the sinh function may overflow. Instead, we use
\[
f_0 \phi^\text{const}_{\text{sur}}(z) = \frac{1}{\lambda} \frac{e^{i\lambda z} + e^{-i\lambda(z+2D)}}{1 - e^{-2\lambda D}}
\]
(B5)

and
\[
f_0 \phi^\text{const}_{\text{bot}}(z) = -\frac{1}{\lambda} \frac{e^{i\lambda(z-D)} + e^{-i\lambda(z+D)}}{1 - e^{-2\lambda D}}
\]
(B6)

for reliable numerical evaluation.
B.2. Exponential Stratification

The surface SQG modes for exponential stratification are first solved by LaCasce (2012). Defining the scale $\eta \equiv \frac{N_0k_b}{2\pi}$, the surface mode is given by

$$
\phi^\text{surf}_\text{exp} = \frac{e^{z\eta}}{N_0K} K_0 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right) I_1 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right) + I_0 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right) K_1 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right)
$$

(B7)

while the bottom mode can be shown to be

$$
\phi^\text{bot}_\text{exp} = \frac{e^{z\eta}}{N_0K} K_0 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right) I_1 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right) + I_0 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right) K_1 \left( 2\eta e^{-\frac{2\eta z}{2\eta}} \right)
$$

(B8)

Bessel functions $K_0$ and $K_1$ should not be confused with the wavenumber $K$.

B.3. WKB Solution

The WKB solution for the SQG mode does not appear to have been previously derived. Assuming a solution of the form $\phi = e^{\theta(z)}$ and inserting this into equation (11), we have that

$$
q'' + q'^2 - 2(\ln N)' q' - N^2 K^2 = 0
$$

(B9)

We then assume a series expansion $q = q_0 + eq_1 + ...$ combined with the assumption that both $q''$ and $N'$ are $O(\epsilon)$. The two lowest order equations are therefore

$$
q_0'' + 2q_0'q_1' - 2(\ln N)' q_0' = 0O(1)
$$

(B10)

$$
q_0'' + 2q_0'q_1' - 2(\ln N)' q_0' = 0O(\epsilon)
$$

(B11)

which has solution

$$
\phi(z) \approx \sqrt{\frac{N(z)}{N_0}} \left[ A e^{kz} f_{0,0}^{\text{N(z)}} + Be^{\frac{z}{\eta}} f_{1,0}^{\text{N(z)}} \right]
$$

(B12)

Applying boundary condition $f_0 \phi^\text{surf}_\text{C}(-D) = 0$ and $f_0 \phi^\text{surf}_\text{bot}(0) = 1$, the surface mode is

$$
\phi^\text{WKB}\text{surf}_\text{C}(z) \approx \frac{1}{KN_0} \sqrt{\frac{N(z)}{N_0}} \left[ A e^{\theta(z)} + e^{-\theta(z) + 2\pi i b}\right]
$$

(B13)

where $s(z) \equiv \frac{k}{f_0} \int_0^z N(z) \, dz$ in the integral from the surface to depth, while $a = \frac{f_0N_0(-D)}{2KN^2(-D)}$, $b = \frac{f_0N_0(0)}{2KN^2_0}$, and $\alpha = \frac{1-a}{1+b}$.

For constant stratification, this reduces to the result in Appendix B.1.

Appendix C: Class Hierarchy

The algorithms in this manuscript are implemented as classes in Matlab in order to take advantage of class-based inheritance. The class hierarchy is shown in Figure C1, where $\text{InternalModeBase}$ is the abstract superclass which defines the primary interface. A class cluster $\text{InternalModes}$ (not part of the hierarchy) is included to provide a single interface from which to initialize all the concrete subclasses.

The primary $\text{Spectral}$ class uses depth ($z$) coordinates from section 4 to compute the eigenvalue problem, while the $\text{DensitySpectral}$ and $\text{WKB}$ $\text{Spectral}$ subclasses use the stretched coordinates described in sections 4.2 and 4.3, respectively. The $\text{AdaptiveSpectral}$ class overrides its superclass when the frequency is high enough, as described in section 4.4. The $\text{ConstantStratification}$ class implements the analytical solution from Appendices A.1 and B.1, while the $\text{ExponentialStratification}$ class implements the analytical solution from Appendices A.2 and B.2. The $\text{WKB}$ class implements the WKB approximated solution from Appendices A.3 and B.3 and inherits from the $\text{Spectral}$ class in order to use...
and 1536747. Science Foundation Awards 1658564 80NSSC18K0070, and the National Office of Navy Research Grant N00014-15-1-2465, NASA grant. Thanks to Jonathan Lilly for his careful comments on an early draft of this manuscript. This work was supported by the Lilly for his careful comments on an early draft of this manuscript. This work was supported by the Office of Navy Research Grant N00014-15-1-2465, NASA grant 80NSSC18K0070, and the National Science Foundation Awards 1658564 and 1536747.

Acknowledgments

All classes available online (https://github.com/JeffreyEarly/GLOceanKit/). Thanks to Jonathan Lilly for his careful comments on an early draft of this manuscript. This work was supported by the Office of Navy Research Grant N00014-15-1-2465, NASA grant 80NSSC18K0070, and the National Science Foundation Awards 1658564 and 1536747.

Figure C1. Class hierarchy for the Matlab implementation of the algorithms in this manuscript. InternalModesBase is an abstract class.

the spectrally computed stratification function, $N^2$. The FiniteDifference class uses finite difference matrices of arbitrary order, on arbitrary grids using the algorithm described in Fornberg (1998).

References

Arkten, G. (1970). Mathematical methods for physicists (Second). Burlington, MA: Academic Press.

Boyd, J. P. (2014). Solving transcendental equations: The Chebyshev polynomial proxy and other numerical rootfinders, perturbation series and oracles. Philadelphia: SIAM.

Buijsman, M. C., McWilliams, J. C., & Jackson, C. R. (2010). East-west asymmetry in nonlinear internal waves from Luzon Strait. Journal of Geophysical Research, 115, C10057. https://doi.org/10.1029/2009JC006004

Canuto, C., Hussaini, M. Y., Quarteroni, A., & Zang, T. A. (2006). Spectral methods: Fundamentals in single domains. Cambridge: Cambridge University Press.

Cushman-Roisin, B., & Beckers, J.-M. (2011). Introduction to geophysical fluid dynamics: Physical and numerical aspects (Vol. 101). Amsterdam: Academic Press.

De Boor, C. (1978). A practical guide to splines (Vol. 27). New York: Springer-Verlag.

DeSaubies, Y. J. F. (1973). Internal waves near the turning point. Geophysical Fluid Dynamics, 5(1), 143–154.

Desaubies, Y. J. F. (1973). Internal waves near the turning point. Geophysical Fluid Dynamics, 5(1), 143–154.

Dunphy, M. (2009). The influence of mesoscale eddies on the internal tide (Master’s Thesis). University of Waterloo.

Early, J. J., & Sukulski, A. M. (2020). Smoothing and interpolating noisy GPS data with smoothing splines. Journal of Atmospheric and Oceanic Technology. https://10.1175/JTECH-D-19-0087.1

Fornberg, B. (1998). Calculation of weights in finite difference formulas. SIAM Review, 40(3), 685–691.

Garrett, C., & Munk, W. (1982). Atmosphere-ocean dynamics. San Diego: Academic press.

Kelly, S. M. (2016). The vertical mode decomposition of surface and internal tides in the presence of a free surface and arbitrary topography. Journal of Physical Oceanography, 46(12), 3777–3788.

Kelly, S. M., Nash, J. D., Martini, K. I., Alford, M. H., & Kunze, E. (2012). The cascade of tidal energy from low to high modes on a continental slope. Journal of Physical Oceanography, 42(7), 1217–1232.

Kelly, S. M., Nash, J. D., Martini, K. I., Alford, M. H., & Kunze, E. (2012). The cascade of tidal energy from low to high modes on a continental slope. Journal of Physical Oceanography, 42(7), 1217–1232.

Kunze, E. (2017). The internal-wave-driven meridional overturning circulation. Journal of Physical Oceanography, 47(6), 1325–1345.

Kunze, E. (2017). The internal-wave-driven meridional overturning circulation. Journal of Physical Oceanography, 47(6), 1325–1345.

LaCasce, J. H. (2012). Surface quasigeostrophic solutions and baroclinic modes with exponential stratification. Journal of Physical Oceanography, 42(4), 569–580.

LaCasce, J. H. (2012). Surface quasigeostrophic solutions and baroclinic modes with exponential stratification. Journal of Physical Oceanography, 42(4), 569–580.

LaCasce, J. H. (2017). The prevalence of oceanic surface modes. Geophysical Research Letters, 44, 11,097–11,105. https://doi.org/10.1002/2017GL075430

LaCasce, J. H., & Wang, J. (2015). Estimating subsurface velocities from surface fields with idealized stratification. Journal of Physical Oceanography, 45(9), 2424–2435.

Lapeyre, G., & Klein, P. (2006). Dynamics of the upper oceanic layers in terms of surface quasigeostrophic theory. Journal of Physical Oceanography, 36(2), 165–176.

Munk, W. (1981). Internal waves and small-scale processes, Evolution of physical oceanography (pp. 264–291). London: MIT Press.

Pedlosky, J. (1987). Geophysical fluid dynamics. New York, NY: Springer-Verlag.

Polzin, K. L., & Lvov, Y. V. (2011). Toward regional characterizations of the oceanic internal wavefield. Reviews of Geophysics, 49, RG4003. https://doi.org/10.1029/2010RG000329

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). Numerical recipes in C (2nd ed.)., The Art of Scientific Computing. Cambridge: Cambridge University Press.

Pya, N., & Wood, S. N. (2015). Shape constrained additive models. Statistics and Computing, 25(3), 543–559.

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.

Smith, K. S., & Vanneste, J. (2013). A surface-aware projection basis for quasigeostrophic flow. Journal of Physical Oceanography, 43(3), 548–562.
Sun, H., & Kunze, E. (1999). Internal wave-wave interactions. Part I: The role of internal wave vertical divergence. *Journal of Physical Oceanography*, 29(11), 2886–2904.

Tulloch, R., & Smith, K. S. (2009a). A note on the numerical representation of surface dynamics in quasigeostrophic turbulence: Application to the nonlinear Eddy model. *Journal of the Atmospheric Sciences*, 66(4), 1063–1068.

Tulloch, R., & Smith, K. S. (2009b). Quasigeostrophic turbulence with explicit surface dynamics: Application to the atmospheric energy spectrum. *Journal of the Atmospheric Sciences*, 66(2), 459–467.

Wang, J., Flierl, G. R., LaCasce, J. H., McClean, J. L., & Mahadevan, A. (2013). Reconstructing the ocean’s interior from surface data. *Journal of Physical Oceanography*, 43(8), 1611–1626.

Winters, K. B., & D’Asaro, E. A. (1997). Direct simulation of internal wave energy transfer. *Journal of Physical Oceanography*, 27(9), 1937–1945.

Wortham, C., & Wunsch, C. (2014). A multidimensional spectral description of ocean variability. *Journal of Physical Oceanography*, 44(3), 944–968.

Wunsch, C. (1997). The vertical partition of oceanic horizontal kinetic energy. *Journal of Physical Oceanography*, 27(8), 1770–1794.

Zhao, Z. (2016). Internal tide oceanic tomography. *Geophysical Research Letters*, 43, 9157–9164. https://doi.org/10.1002/2016GL070567

Zilberman, N. V., Becker, J. M., Merrifield, M. A., & Carter, G. S. (2009). Model estimates of M2 internal tide generation over mid-Atlantic ridge topography. *Journal of Physical Oceanography*, 39(10), 2635–2651.