Quadratic Spline Galerkin Method for the Shallow Water Equations on the Sphere

Anita T. Layton[1]*, Christina C. Christara[2]†, and Kenneth R. Jackson[2]†

[1] *Department of Mathematics, University of North Carolina, Chapel Hill, North Carolina, USA.*
[2] *Department of Computer Science, University of Toronto, Toronto, Ontario, Canada.*

E-mail: [1]layton@amath.unc.edu, [2]{ccc, krj}@cs.toronto.edu

Abstract

Currently in most global meteorological applications, the spectral transform method or low-order finite difference/finite element methods are used. The spectral transform method, which yields high-order approximations, requires Legendre transforms. The Legendre transforms have a computational complexity of $\mathcal{O}(N^3)$, where N is the number of subintervals in one dimension, and thus render the spectral transform method unscalable. In this study, we present an alternative numerical method for solving the shallow water equations (SWEs) on a sphere in spherical coordinates. In this implementation, the SWEs are discretized in time using the two-level semi-Lagrangian semi-implicit method, and in space on staggered grids using the quadratic spline Galerkin method. We show that, when applied to a simplified version of the SWEs, the method yields a neutrally stable solution for the meteorologically significant Rossby waves. Moreover, we demonstrate that the Helmholtz equation arising from the discretization and solution of the SWEs should be derived algebraically rather than analytically, in order for the method to be stable with respect to the Rossby waves. These results are verified numerically using Boyd's equatorial wave equations [2] with initial conditions chosen to generate a soliton.

Keywords: numerical weather prediction; finite element; semi-Lagrangian semi-implicit; Rossby stability; staggered grids.

Subject classifications: 65N30; 76B65; 86A10.

---

*Part of this work was done while the author was at the Department of Computer Science, University of Toronto.

†Partly supported by Natural Sciences and Engineering Research Council (NSERC) of Canada.

1 Introduction

Weather prediction is a science with a long history. Its objective is the description and prediction of the behaviour of the atmosphere, ocean water and sea ice. The accuracy of weather prediction depends on many factors, including the accuracy of the knowledge of the state of the atmosphere at the initial time, the numerical methods applied, and the resolution used in these methods. Weather prediction computations are very time-consuming. Therefore, there is a lot of interest among the scientific community in studying accurate and efficient methods for weather prediction. One way to achieve high accuracy in weather prediction computations is to consider high-order discretization methods.

The spatial discretization schemes that are commonly used in meteorological simulations are finite difference schemes, spectral schemes, and finite element schemes. For instance, the models used by the National Center for Atmospheric Research (NCAR) and the European Centre for Medium-Range Weather Forecasts (ECMWF) are based on the spectral transform method [8, 17, 18], while the model developed by the Canadian Meteorological Centre in partnership with the Meteorological Research Branch (CMC-MRB) uses a variable-resolution cell-integrated finite element scheme [3, 4]. Although the spectral transform method is popular and offers superior accuracy at a given resolution, the efficiency of the method at high resolutions is reduced, because the cost of performing spectral transforms increases rapidly with spatial resolution. In the case of Fourier transforms in the longitudinal direction, fast Fourier Transforms (FFTs) may be used and their computational cost grows as $\mathcal{O}(N^2 \log(N))$, where N is the number of spatial subintervals in one dimension. An efficient method for performing Legendre transforms, analogous to FFTs, has not yet been developed. Thus, the Legendre transforms in the latitudinal direction are often performed by summation and their costs escalate at a rate of $\mathcal{O}(N^3)$. Therefore, finite element methods will likely offer a competitive alternative in the future, since, compared to the spectral transform method, their computational complexity grows at a slower rate. Provided that a sufficiently efficient solver is applied for the solution of the linear system arising from the Helmholtz problem, the computational costs of finite difference and finite element methods applied to the shallow water equations (SWEs) on the sphere increase quadratically with the number of gridpoints in one dimension (i.e., $\mathcal{O}(N^2)$). Moreover, finite element methods can incorporate adaptive grids and, compared to spectral methods, are more suitable for massively parallel computers.

In this paper and in a companion paper [10], we present finite element-based numerical methods for the SWEs in spherical coordinates. The SWEs, which describe the inviscid flow of a thin layer of fluid in two dimensions [7], have been used for many years by the atmospheric modeling community to test promising numerical methods for solving atmospheric and oceanic problems. Because the earth is approximately spherical, most global atmospheric models in use today are based on spherical coordinates. To define the equations on the sphere, let u and v be the wind velocity components in the λ (longitudinal) and θ (latidudinal) directions, respectively, and ϕ be the geopotential. Let R be the radius of the earth, Ω its rotational speed, and $f = 2\Omega \sin\theta$ the Coriolis parameter, where R and Ω are assumed to be constant. In spherical coordinates, the SWEs are given by

$$\frac{du}{dt} - \left(f + \frac{u\tan\theta}{R}\right) v + \frac{\phi_\lambda}{R\cos\theta} = 0, \tag{1}$$

$$\frac{dv}{dt} + \left(f + \frac{u\tan\theta}{R}\right) u + \frac{\phi_\theta}{R} = 0, \tag{2}$$

$$\frac{d\phi}{dt} + \frac{\phi}{R\cos\theta}[u_\lambda + (v\cos\theta)_\theta] = 0, \tag{3}$$

where the Lagrangian derivative is defined by

$$\frac{d}{dt} \equiv \frac{\partial}{\partial t} + \frac{u}{R\cos\theta}\frac{\partial}{\partial\lambda} + \frac{v}{R}\frac{\partial}{\partial\theta}, \tag{4}$$

and the subscripts λ and θ denote the spatial derivatives in the two directions. Since u and v are multi-valued at the poles, we adopt the approach of Côté and Staniforth [6] and compute the components of the wind images instead: $U \equiv u\cos\theta/R$ and $V \equiv v\cos\theta/R$. Thus, we solve the following equations, obtained by multiplying the motion equations (1)–(2) and the continuity equation (3) by $\cos\theta/R$ and $\cos\theta$, respectively, expressing the resulting equations in terms of U and V, and isolating the nonlinearity in the continuity equation (3) in a logarithmic term

$$\frac{dU}{dt} - fV + \frac{\phi_\lambda}{R^2} = 0, \tag{5}$$

$$\frac{dV}{dt} + fU + \frac{\cos\theta}{R^2}\phi_\theta + \delta = 0, \tag{6}$$

$$\cos\theta \frac{d}{dt}\log\phi + \left[\frac{U_\lambda}{\cos\theta} + V_\theta\right] = 0, \tag{7}$$

where $\delta \equiv (U^2 + V^2)\sin\theta / \cos^2\theta$.

Along the longitude, functions are assumed periodic, whereas at the poles ($\theta = \pm\pi/2$), homogeneous Dirichlet boundary conditions are imposed on the wind image components: $U(\lambda, \pm\pi/2) = V(\lambda, \pm\pi/2) = 0$. The latitudinal boundary conditions on ϕ are designed to mimic the behaviour of its spherical harmonic expansions [6], and are set to be homogeneous Neumann: $\phi_\theta(\lambda, \pm\pi/2) = 0$.

In this paper, the SWEs are discretized in time using the two-level semi-Lagrangian semi-implicit (SLSI) method, and in space using the quadratic spline Galerkin (QSG) method. In the companion paper [10], the optimal quadratic spline collocation (OQSC) methods are used for the spatial discretization. In Section 2, we outline the SLSI time discretization method. In Section 3, we present the QSG method for the time-discretized SWEs. We describe how the quadratic spline basis functions should be adjusted to satisfy the boundary conditions; we emphasize that special care must be taken for handling the latitudinal boundary conditions associated with the staggered partition, for which the poles (i.e., the latitudinal boundary points) are not gridpoints. (In a spatial discretization method, the boundary points are usually gridpoints of the spatial partition.) At the end of Section 3, we discuss how the integrals arising in the Galerkin method should be evaluated when spatial discretization is done on staggered grids.

In Section 4, we extend the study of Staniforth and Mitchell [15], carried out for Cartesian coordinates, to spherical coordinates. By applying linear stability analysis to a simplified version of the SWEs, we show that, if biperiodic boundary conditions are applied, the Helmholtz equation arising after eliminating the divergence terms from the continuity equation should be derived algebraically rather than analytically in order for the standard spatial discretization methods to be neutrally stable for the meteorologically significant Rossby waves. The analysis applies to the QSG method as well as to all spatial discretization methods that give rise to circulant matrices for biperiodic boundary conditions. In Section 5, we verify, using Boyd's equatorial soliton test case [2], that the QSG method, when applied in conjunction with the SLSI method, is indeed stable and non-dispersive for the Rossby waves. Finally, we present numerical results which indicate that the QSG method has a locally fourth-order spatial convergence.

2 Time Discretization

Discretization schemes based on a semi-Lagrangian treatment of advection have generated considerable interest in the past decade for the efficient integration of atmospheric models, since they offer the promise of larger timestep size, with no loss in accuracy when compared to the Eulerian-based advection schemes, in which the timestep size is limited by more severe stability restrictions [13, 14].

A semi-Lagrangian time discretization scheme in spherical coordinates approximates the Lagrangian derivative along particle trajectories defined by the velocity vector with components $d\lambda/dt = u/(R\cos\theta)$ and $d\theta/dt = v/R$. Let $(\delta\lambda^{n+1}, \delta\theta^{n+1})$ be the displacement of a fluid particle in the time interval t_n to t_{n+1}, ending at the downstream point (λ, θ) at t_{n+1}. We adopt the two-level scheme which approximates a function on a trajectory originating at the upstream departure point $(\lambda - \delta\lambda^{n+1}, \theta - \delta\theta^{n+1})$ and terminating at (λ, θ). For an arbitrary function $\psi(\lambda, \theta, t)$, let $\tilde{\psi}^n$ denote the corresponding upstream function for the time interval t_n to t_{n+1}. That is, for the downstream gridpoint (λ, θ) associated with the displacement ($\delta\lambda^{n+1}$, $\delta\theta^{n+1}$)

$$\tilde{\psi}^n(\lambda, \theta) \equiv \psi(\lambda - \delta\lambda^{n+1}, \theta - \delta\theta^{n+1}, t_n). \tag{8}$$

The Lagrangian derivative in the interval t_n to t_{n+1} is approximated by

$$\frac{d\tilde{\psi}^{n+\frac{1}{2}}}{dt} \approx \frac{\psi^{n+1} - \tilde{\psi}^n}{\Delta t}. \tag{9}$$

The semi-Lagrangian approach provides accurate approximations for advection with virtually no time-step restriction. However, if the gravity terms were treated explicitly, they would severely restrict the timestep even with semi-Lagrangian advection approximations. Therefore, to obtain maximum benefit from the semi-Lagrangian approach, one needs to combine the semi-Lagrangian approximations with semi-implicit approximations. A semi-implicit treatment of the gravity and Coriolis terms implies that a function is averaged in time along particle trajectories:

$$\tilde{\psi}^{n+\frac{1}{2}} \approx \frac{\psi^{n+1} + \tilde{\psi}^{n}}{2}. \tag{10}$$

When discretized in time using the two-level SLSI scheme, the SWEs take the following form:

$$\frac{U^{n+1} - \tilde{U}^{n}}{\Delta t} - \frac{\tilde{f}^{n+\frac{1}{2}}}{2}(V^{n+1} + \tilde{V}^{n}) + \frac{1}{2R^2}(\phi_{\lambda}^{n+1} + \tilde{\phi}_{\lambda}^{n}) = 0, \tag{11}$$

$$\frac{V^{n+1} - \tilde{V}^{n}}{\Delta t} + \frac{\tilde{f}^{n+\frac{1}{2}}}{2}(U^{n+1} + \tilde{U}^{n}) + \frac{\cos\tilde{\theta}^{n+\frac{1}{2}}}{2R^2}(\phi_{\theta}^{n+1} + \tilde{\phi}_{\theta}^{n}) + \tilde{\delta}^{n+\frac{1}{2}} = 0, \tag{12}$$

$$\cos\tilde{\theta}^{n+\frac{1}{2}}\left(\frac{\log\phi^{n+1} - \log\tilde{\phi}^{n}}{\Delta t}\right) + \frac{1}{2}\left[\left(\frac{U_{\lambda}^{n+1} + \tilde{U}_{\lambda}^{n}}{\cos\tilde{\theta}^{n+\frac{1}{2}}}\right) + (V_{\theta}^{n+1} + \tilde{V}_{\theta}^{n})\right] = 0. \tag{13}$$

Functions at time-level t_{n+1} are evaluated at gridpoints (λ_l, θ_m); those at time-level $t_{n+\frac{1}{2}}$ are evaluated at approximate trajectory midpoints $(\lambda_l - \delta\lambda_l^{n+1}/2, \theta_m - \delta\theta_m^{n+1}/2)$; and those at time-level t_n, called the "upstream functions," are evaluated at approximate departure points $(\lambda_l - \delta\lambda_l^{n+1}, \theta_m - \delta\theta_m^{n+1})$. For notational simplicity, let $\tilde{f} \equiv \tilde{f}^{n+\frac{1}{2}}$ and $\tilde{\theta} \equiv \tilde{\theta}^{n+\frac{1}{2}}$. The trajectories are computed according to the algorithm described by Côté and Staniforth [5]. The departure points usually fall between gridpoints and spatial interpolation is required to obtain function values at departure points. Cubic polynomial interpolation gives fourth-order spatial truncation errors with very little damping [1], and is thus the method of choice in our implementation.

To solve the time discretized SWEs (11)–(13), the equations are first discretized in space, and then the wind images U and V are eliminated from the system to yield a nonlinear Helmholtz equation for the geopotential ϕ only. To this end, we move the ϕ_{λ}^{n+1} and ϕ_{θ}^{n+1} terms of the motion equations (11) and (12) to the right side and then rewrite the resulting equations in matrix form

$$\begin{bmatrix} U^{n+1} \\ V^{n+1} \end{bmatrix} = \begin{bmatrix} 1 & -\frac{\Delta t\tilde{f}}{2} \\ \frac{\Delta t\tilde{f}}{2} & 1 \end{bmatrix}^{-1} \left\{ \begin{bmatrix} 1 & \frac{\Delta t\tilde{f}}{2} \\ -\frac{\Delta t\tilde{f}}{2} & 1 \end{bmatrix} \begin{bmatrix} \tilde{U}^{n} \\ \tilde{V}^{n} \end{bmatrix} \right.$$

$$- \left(\frac{\Delta t}{2R^2}\right) \left[\begin{array}{c} \phi_\lambda^{n+1} + \tilde{\phi}_\lambda^n \\ \cos\tilde{\theta}(\phi_\theta^{n+1} + \tilde{\phi}_\theta^n) + 2R^2 \tilde{\delta}^{n+\frac{1}{2}} \end{array} \right] \Bigg\},$$

which can then be rewritten as

$$\begin{aligned} \begin{bmatrix} U^{n+1} \\ V^{n+1} \end{bmatrix} + \left(\frac{\Delta t}{2R^2}\right) \begin{bmatrix} c & d \\ -d & c \end{bmatrix} \begin{bmatrix} \phi_\lambda^{n+1} \\ \cos\tilde{\theta}\phi_\theta^{n+1} \end{bmatrix} \\ = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} \tilde{U}^n \\ \tilde{V}^n \end{bmatrix} - \left(\frac{\Delta t}{2R^2}\right) \begin{bmatrix} c & d \\ -d & c \end{bmatrix} \begin{bmatrix} \tilde{\phi}_\lambda^n \\ \cos\tilde{\theta}\tilde{\phi}_\theta^n + 2R^2\tilde{\delta}^{n+\frac{1}{2}} \end{bmatrix} \equiv \begin{bmatrix} \tilde{R}_u^n \\ \tilde{R}_v^n \end{bmatrix}, \end{aligned} \tag{14}$$

where a, b, c, and d are functions of θ defined as follows:

$$a = \frac{1 - \left(\frac{\Delta t \tilde{f}}{2}\right)^2}{1 + \left(\frac{\Delta t \tilde{f}}{2}\right)^2}, \quad b = \frac{\Delta t \tilde{f}}{1 + \left(\frac{\Delta t \tilde{f}}{2}\right)^2}, \quad c = \frac{1}{1 + \left(\frac{\Delta t \tilde{f}}{2}\right)^2}, \quad d = \frac{\frac{\Delta t}{2}\tilde{f}}{1 + \left(\frac{\Delta t \tilde{f}}{2}\right)^2}. \tag{15}$$

We also rewrite (13) as

$$\begin{aligned} \cos\tilde{\theta}^{n+\frac{1}{2}} \log\phi^{n+1} + \frac{\Delta t}{2}\left(\frac{U_\lambda^{n+1}}{\cos\tilde{\theta}^{n+\frac{1}{2}}} + V_\theta^{n+1}\right) \\ = \cos\tilde{\theta}^{n+\frac{1}{2}} \log\tilde{\phi}^n - \frac{\Delta t}{2}\left(\frac{\tilde{U}_\lambda^n}{\cos\tilde{\theta}^{n+\frac{1}{2}}} + \tilde{V}_\theta^n\right) \equiv \tilde{R}_\phi^n. \end{aligned} \tag{16}$$

To advance the solution by one timestep, (14) and (16) are spatially discretized. Then the divergence terms (i.e., U_λ^{n+1} and V_θ^{n+1}) are eliminated from the spatially discretized equation arising from (16) using the respective equation corresponding to (14), resulting in a Helmholtz equation for ϕ^{n+1}. The details of the above procedure are described in Section 3 and in Appendix A.

3 Space Discretization

In this section, we develop the procedures with which the QSG method can be applied to equations that are discretized in time using the SLSI method. As will be explained in Section 4, the discretized Helmholtz equation is derived from the spatially discretized form of (14) and (16), in order to preserve the phase velocity of the Rossby waves. Furthermore, the correct direction for energy propagation is maintained by discretizing on the Arakawa C-type grid [7].

Along the longitude, we define two uniform partitions

$$\begin{aligned} \Delta_\lambda &\equiv \{0 = \lambda_0 < \lambda_1 < \cdots < \lambda_{N_\lambda} = 2\pi\} \text{ and} \\ \hat{\Delta}_\lambda &\equiv \left\{-\tfrac{\Delta\lambda}{2} = \hat{\lambda}_{-1} < \hat{\lambda}_0 < \cdots < \hat{\lambda}_{N_\lambda} = 2\pi + \tfrac{\Delta\lambda}{2}\right\}, \end{aligned}$$

where $N_\lambda > 0$ is an integer and $\Delta\lambda = 2\pi/N_\lambda$ denotes the meshsize in the λ-direction. The two partitions Δ_λ and $\hat{\Delta}_\lambda$ are staggered with respect to each other. The gridpoints in Δ_λ and $\hat{\Delta}_\lambda$ are chosen so that $\lambda_i = i\Delta\lambda$ for $i = 0, \ldots, N_\lambda$, and $\hat{\lambda}_i = (i+1/2)\Delta\lambda$ for $i = -1, \ldots, N_\lambda$, respectively. Similarly, staggered partitions Δ_θ and $\hat{\Delta}_\theta$ are defined in the θ-direction by

$$\begin{aligned} \Delta_\theta &\equiv \left\{-\tfrac{\pi}{2} = \theta_0 < \theta_1 < \cdots < \theta_{N_\theta} = \tfrac{\pi}{2}\right\} \text{ and} \\ \hat{\Delta}_\theta &\equiv \left\{-\tfrac{\pi}{2} - \tfrac{\Delta\theta}{2} = \hat{\theta}_{-1} < \hat{\theta}_0 < \cdots < \hat{\theta}_{N_\theta} = \tfrac{\pi}{2} + \tfrac{\Delta\theta}{2}\right\}, \end{aligned}$$

where $N_\theta > 2$ is an integer, $\Delta\theta = \pi/N_\theta$, $\theta_j = -\pi/2 + j\Delta\theta$ for $j = 0, \ldots, N_\theta$, and $\hat{\theta}_j = -\pi/2 + (j + 1/2)\Delta\theta$ for $j = -1, \ldots, N_\theta$.

For each partition, consider the associated one-dimensional quadratic spline space, i.e., the space of piecewise quadratic polynomials with C^1 continuity on the gridpoints of the partition. Define the model quadratic spline function ψ by

$$\psi(\lambda) = \begin{cases} \lambda^2 & 0 \le \lambda \le 1, \\ \lambda^2 - 3(\lambda-1)^2 & 1 \le \lambda \le 2, \\ \lambda^2 - 3(\lambda-1)^2 + 3(\lambda-2)^2 & 2 \le \lambda \le 3, \\ 0 & \text{otherwise,} \end{cases}$$

and let $\{\beta_i(\lambda) = \frac{1}{2}\psi(\frac{\lambda}{\Delta\lambda} - i + 2)\}_{i=0}^{N_\lambda+1}$ and $\{\hat{\beta}_i(\lambda) = \frac{1}{2}\psi(\frac{\lambda}{\Delta\lambda} - i + \frac{3}{2})\}_{i=-1}^{N_\lambda+1}$ be the sets of basis functions for the one-dimensional quadratic spline approximation spaces corresponding to partitions Δ_λ and $\hat{\Delta}_\lambda$, respectively. For the partitions Δ_θ and $\hat{\Delta}_\theta$, let $\{\beta_j(\theta) = \frac{1}{2}\psi(\frac{\theta+\pi/2}{\Delta\theta} - j + 2)\}_{j=0}^{N_\theta+1}$ and $\{\hat{\beta}_j(\theta) = \frac{1}{2}\psi(\frac{\theta+\pi/2}{\Delta\theta} - j + \frac{3}{2})\}_{j=-1}^{N_\theta+1}$ be the respective sets of basis functions. Figure 1 shows a diagram of the latitudinal staggered grids together with the associated basis functions. There are $N_\lambda + 2$ basis functions associated with Δ_λ, $N_\lambda + 3$ with $\hat{\Delta}_\lambda$, $N_\theta + 2$ with Δ_θ, and $N_\theta + 3$ with $\hat{\Delta}_\theta$. Note that the latitudinal boundary points (i.e., the poles) are not gridpoints in $\hat{\Delta}_\theta$.

The target functions belong to two-dimensional approximation spaces, which are chosen to be tensor products of the associated one-dimensional approximation spaces.

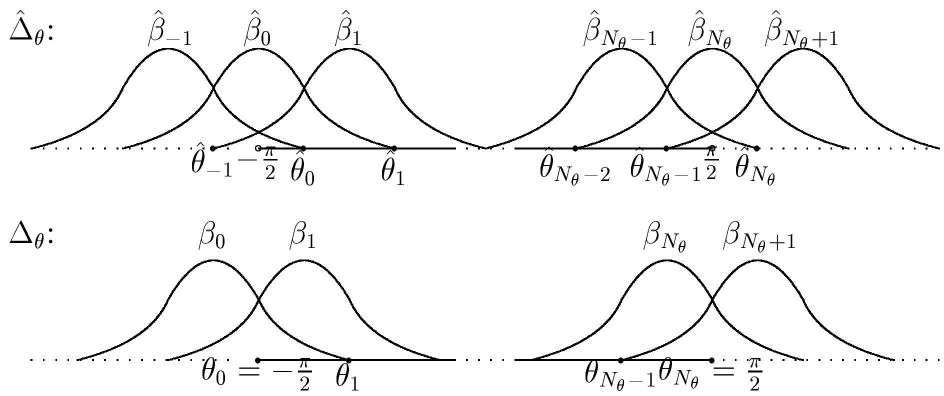

Figure 1: A diagram of the latitudinal staggered grids and associated basis functions.

Discretized on a C-grid, the target functions U^{n+1}, V^{n+1}, and ϕ^{n+1} are approximated in the biquadratic spline spaces defined on the induced grid partitions $\Delta_1 \equiv \hat{\Delta}_\lambda \times \Delta_\theta$, $\Delta_2 \equiv \Delta_\lambda \times \hat{\Delta}_\theta$, and $\Delta_3 \equiv \Delta_\lambda \times \Delta_\theta$, respectively, by linear combinations of the basis functions $\{\hat{\beta}_i(\lambda)\beta_j(\theta)\}_{i=-1,j=0}^{N_\lambda+1,N_\theta+1}$, $\{\beta_i(\lambda)\hat{\beta}_j(\theta)\}_{i=0,j=-1}^{N_\lambda+1,N_\theta+1}$, and $\{\beta_i(\lambda)\beta_j(\theta)\}_{i=0,j=0}^{N_\lambda+1,N_\theta+1}$, respectively. That is,

$$U_\Delta^{n+1}(\lambda,\theta) = \sum_{i=-1,j=0}^{N_\lambda+1,N_\theta+1} \mathsf{U}_{i,j}^{n+1}\hat{\beta}_i(\lambda)\beta_j(\theta), \tag{17}$$

$$V_\Delta^{n+1}(\lambda,\theta) = \sum_{i=0,j=-1}^{N_\lambda+1,N_\theta+1} \mathsf{V}_{i,j}^{n+1}\beta_i(\lambda)\hat{\beta}_j(\theta), \tag{18}$$

$$\phi_\Delta^{n+1}(\lambda,\theta) = \sum_{i=0,j=0}^{N_\lambda+1,N_\theta+1} \Phi_{i,j}^{n+1}\beta_i(\lambda)\beta_j(\theta). \tag{19}$$

Furthermore, the quadratic spline representation of $\log\phi^{n+1}$ is needed because of the presence of $\log\phi^{n+1}$ in (16). Thus, we define $\zeta^{n+1} \equiv \log\phi^{n+1}$ and approximate ζ^{n+1} on the induced grid partition Δ_3 (same as ϕ^{n+1}) by

$$\zeta_\Delta^{n+1}(\lambda,\theta) = \sum_{i=0,j=0}^{N_\lambda+1,N_\theta+1} \Gamma_{i,j}^{n+1}\beta_i(\lambda)\beta_j(\theta). \tag{20}$$

The coefficients $\Gamma_{i,j}^{n+1}$ can be computed from the geopotential coefficients $\Phi_{i,j}^{n+1}$ by first transforming ϕ_Δ^{n+1} from quadratic spline space to grid space, then computing ζ_Δ^{n+1} in grid space and transforming ζ_Δ^{n+1} to quadratic spline space to give $\Gamma_{i,j}^{n+1}$.

3.1 Incorporation of Boundary Conditions

In the analysis and the implementation, the approximate spaces and the respective basis functions are adjusted so that they satisfy the boundary conditions by construction. Thus, the approximations U_Δ^{n+1}, V_Δ^{n+1}, ϕ_Δ^{n+1} and ζ_Δ^{n+1} also satisfy the boundary conditions. In this subsection, we explain how the basis functions are adjusted to incorporate the boundary conditions.

For a one-dimensional grid partition with $N+1$ gridpoints, there are $N+2$ degrees of freedom in representing a function in terms of the quadratic spline basis functions. For the staggered $\hat{\Delta}_\theta$, for example, there are $N_\theta + 3$ degrees of freedom. By incorporating the boundary conditions, two or three degrees of freedom can be eliminated. In our case, the "outermost" basis functions are removed and the neighbouring ones are modified. How this is done depends on the boundary conditions.

We first discuss how to incorporate periodic boundary conditions in the longitudinal dimension. Since the longitude "wraps around" the sphere, all the functions in (17)–(20) are periodic in the longitudinal dimension. In order to form a set of basis functions so that all target functions satisfy periodicity along the longitude, we drop $\beta_0(\lambda)$, $\hat{\beta}_{-1}(\lambda)$, $\hat{\beta}_0(\lambda)$, $\beta_{N_\lambda+1}(\lambda)$ and $\hat{\beta}_{N_\lambda+1}(\lambda)$, and modify the definitions of $\beta_j(\lambda)$ and $\hat{\beta}_j(\lambda)$ for $j = 1, \ldots, N_\lambda$ to be

$$\beta_1^P(\lambda) \equiv \begin{cases} \beta_1(\lambda) & \lambda \in [\lambda_0, \lambda_2], \\ \beta_{N_\lambda+1}(\lambda) & \lambda \in [\lambda_{N_\lambda-1}, \lambda_{N_\lambda}], \\ 0 & \text{otherwise}, \end{cases} \qquad \hat{\beta}_1^P(\lambda) \equiv \begin{cases} \hat{\beta}_1(\lambda) & \lambda \in [0, \hat{\lambda}_2], \\ \hat{\beta}_{N_\lambda+1}(\lambda) & \lambda \in [\hat{\lambda}_{N_\lambda-1}, 2\pi], \\ 0 & \text{otherwise}, \end{cases}$$

$$\beta_j^P(\lambda) \equiv \beta_j(\lambda), \qquad \hat{\beta}_j^P(\lambda) \equiv \hat{\beta}_j(\lambda), \qquad j = 2, \ldots, N_\lambda - 2,$$

$$\beta_{N_\lambda-1}^P(\lambda) \equiv \beta_{N_\lambda-1}(\lambda), \qquad \hat{\beta}_{N_\lambda-1}^P(\lambda) \equiv \begin{cases} \hat{\beta}_{-1}(\lambda) & \lambda \in [0, \hat{\lambda}_0], \\ \hat{\beta}_{N_\lambda-1}(\lambda) & \lambda \in [\hat{\lambda}_{N_\lambda-3}, 2\pi], \\ 0 & \text{otherwise}, \end{cases}$$

$$\beta_{N_\lambda}^P(\lambda) \equiv \begin{cases} \beta_0(\lambda) & \lambda \in [\lambda_0, \lambda_1], \\ \beta_{N_\lambda}(\lambda) & \lambda \in [\lambda_{N_\lambda-2}, \lambda_{N_\lambda}], \\ 0 & \text{otherwise}, \end{cases} \qquad \hat{\beta}_{N_\lambda}^P(\lambda) \equiv \begin{cases} \hat{\beta}_0(\lambda) & \lambda \in [0, \hat{\lambda}_1], \\ \hat{\beta}_{N_\lambda}(\lambda) & \lambda \in [\hat{\lambda}_{N_\lambda-2}, 2\pi], \\ 0 & \text{otherwise}. \end{cases}$$

Note that there are N_λ basis functions for either Δ_λ or $\hat{\Delta}_\lambda$. In the analysis in Section 4, we assume biperiodic boundary conditions. When periodic boundary conditions are applied in the latitudinal dimension, the basis functions $\beta_j^P(\theta)$ and $\hat{\beta}_j^P(\theta)$ for $j = 1, \ldots, N_\theta$ are defined similarly as in the longitudinal dimension.

The details of incorporating Dirichlet or Neumann latitudinal boundary conditions are discussed next. The general idea is that at each end of the latitudinal domain we drop one basis function and modify the neighbouring nonzero basis functions so that they satisfy the boundary conditions. The modification is based on an appropriate linear combination of the (original unmodified) basis functions.

Consider the case of Dirichlet conditions. At the poles, U_Δ^{n+1} vanishes: $U_\Delta^{n+1}(\lambda, \pm\pi/2) = 0$. To impose the homogeneous Dirichlet boundary conditions for U_Δ^{n+1}, the modified basis functions $\beta_i^D(\theta), i = 1, \ldots, N_\theta$, are defined to be

$$
\begin{aligned}
&\beta_1^D(\theta) \equiv \beta_1(\theta) - \beta_0(\theta), \\
&\beta_j^D(\theta) \equiv \beta_j(\theta), && j = 2, \ldots, N_\theta - 1, \\
&\beta_{N_\theta}^D(\theta) \equiv \beta_{N_\theta}(\theta) - \beta_{N_\theta+1}(\theta).
\end{aligned}
$$

Homogeneous Neumann boundary conditions are imposed on ϕ_Δ^{n+1} and ζ_Δ^{n+1} at the poles, where their θ-derivatives vanish (see Section 2). The associated basis functions are adjusted to be

$$
\begin{aligned}
&\beta_1^N(\theta) \equiv \beta_1(\theta) + \beta_0(\theta), \\
&\beta_j^N(\theta) \equiv \beta_j(\theta), && j = 2, \ldots, N_\theta - 1, \\
&\beta_{N_\theta}^N(\theta) \equiv \beta_{N_\theta}(\theta) + \beta_{N_\theta+1}(\theta).
\end{aligned}
$$

The latitudinal homogeneous Dirichlet boundary conditions on V_Δ^{n+1} are imposed at the poles ($V_\Delta^{n+1}(\lambda, \pm\pi/2) = 0$), but the poles are *not* gridpoints of $\hat{\Delta}_\theta$, since $\hat{\theta}_i = -\pi/2 + (i + 1/2)\Delta\theta$ for $i = -1, \ldots, N_\theta$ owing to staggering. Therefore, we cannot incorporate homogeneous Dirichlet boundary conditions for V_Δ^{n+1} in the same way we did for U_Δ^{n+1}. Instead, the latitudinal components of the basis functions for V_Δ^{n+1} are given by

$$
\hat{\beta}_0^D(\theta) = \hat{\beta}_0(\theta) - 6\hat{\beta}_{-1}(\theta), \qquad \hat{\beta}_1^D(\theta) = \hat{\beta}_1(\theta) - \hat{\beta}_{-1}(\theta),
$$

$$\hat{\beta}_j^D(\theta) = \hat{\beta}_j(\theta), \qquad j = 2, \ldots, N_\theta - 2,$$

$$\hat{\beta}_{N_\theta - 1}^D(\theta) = \hat{\beta}_{N_\theta - 1}(\theta) - \hat{\beta}_{N_\theta + 1}(\theta), \qquad \hat{\beta}_{N_\theta}^D(\theta) = \hat{\beta}_{N_\theta}(\theta) - 6\hat{\beta}_{N_\theta + 1}(\theta).$$

Notice that, in this case, two basis functions at each end must be modified, and there are $N_\theta + 1$ modified basis functions.

3.2 Quadratic Spline Galerkin Formulation

In the QSG method, the discrete equations are obtained by incorporating boundary conditions as described in Section 3.1 and imposing conditions on the residuals, using the adjusted basis functions as test functions. More specifically, the approximate solutions (17)–(20) are rewritten in terms of the adjusted basis functions defined in Section 3.1, and then substituted into (14) and (16). The inner products of the residuals of the resulting equations with each of the two-dimensional test functions are then set to zero. Thus the residuals are forced to be orthogonal to the approximation space. Imposing this condition on the residuals yields

$$\begin{aligned}
\int_{\theta_0}^{\theta_{N_\theta}} \int_{\lambda_0}^{\lambda_{N_\lambda}} \sum_{i=1}^{N_\lambda} \sum_{j=1}^{N_\theta} \Big\{ & \mathsf{U}_{i,j}^{n+1} \hat{\beta}_i^P(\lambda) \beta_j^D(\theta) \\
& + \left(\frac{\Delta t}{2R^2}\right) \Big[c(\theta) \Phi_{i,j}^{n+1} \beta_i^{P'}(\lambda) \beta_j^N(\theta) + d(\theta) \cos\tilde{\theta} \Phi_{i,j}^{n+1} \beta_i^P(\lambda) \beta_j^{N'}(\theta) \Big] \Big\} \\
& \hat{\beta}_k^P(\lambda) \beta_l^D(\theta) d\lambda d\theta \\
& = \int_{\theta_0}^{\theta_{N_\theta}} \int_{\lambda_0}^{\lambda_{N_\lambda}} \tilde{R}_u^n(\lambda, \theta) \hat{\beta}_k^P(\lambda) \beta_l^D(\theta) d\lambda d\theta,
\end{aligned} \tag{21}$$

for $k = 1, \ldots, N_\lambda$ and $l = 1, \ldots, N_\theta$,

$$\begin{aligned}
\int_{\theta_0}^{\theta_{N_\theta}} \int_{\lambda_0}^{\lambda_{N_\lambda}} \Bigg\{ & \sum_{i=1}^{N_\lambda} \sum_{\hat{j}=0}^{N_\theta} \mathsf{V}_{i,\hat{j}}^{n+1} \beta_i^P(\lambda) \hat{\beta}_{\hat{j}}^D(\theta) \\
& + \left(\frac{\Delta t}{2R^2}\right) \sum_{i=1}^{N_\lambda} \sum_{j=1}^{N_\theta} \Big[-d(\theta) \Phi_{i,j}^{n+1} \beta_i^{P'}(\lambda) \beta_j^N(\theta) + c(\theta) \cos\tilde{\theta} \Phi_{i,j}^{n+1} \beta_i^P(\lambda) \beta_j^{N'}(\theta) \Big] \Bigg\} \\
& \beta_k^P(\lambda) \hat{\beta}_{\hat{l}}^D(\theta) d\lambda d\theta \\
& = \int_{\theta_0}^{\theta_{N_\theta}} \int_{\lambda_0}^{\lambda_{N_\lambda}} \tilde{R}_v^n(\lambda, \theta) \beta_k^P(\lambda) \hat{\beta}_{\hat{l}}^D(\theta) d\lambda d\theta,
\end{aligned} \tag{22}$$

for $k = 1, \ldots, N_\lambda$ and $\hat{l} = 0, \ldots, N_\theta$, and

$$\int_{\theta_0}^{\theta_{N_\theta}} \int_{\lambda_0}^{\lambda_{N_\lambda}} \left\{ \sum_{i=1}^{N_\lambda} \sum_{j=1}^{N_\theta} \cos\tilde{\theta} \Gamma_{i,j}^{n+1} \beta_i^P(\lambda) \beta_j^N(\theta) \right.$$
$$\left. + \left(\frac{\Delta t}{2}\right) \left[\sum_{i=1}^{N_\lambda} \sum_{j=1}^{N_\theta} \frac{1}{\cos\tilde{\theta}} \mathsf{U}_{i,j}^{n+1} \hat{\beta}_i^{P'}(\lambda) \beta_j^D(\theta) + \sum_{i=1}^{N_\lambda} \sum_{\hat{j}=0}^{N_\theta} \mathsf{V}_{i,\hat{j}}^{n+1} \beta_i^P(\lambda) \hat{\beta}_{\hat{j}}^{D'}(\theta) \right] \right\}$$
$$\beta_k^P(\lambda) \beta_l^N(\theta) d\lambda d\theta$$
$$= \int_{\theta_0}^{\theta_{N_\theta}} \int_{\lambda_0}^{\lambda_{N_\lambda}} \tilde{R}_\phi^n(\lambda, \theta) \beta_k^P(\lambda) \beta_l^N(\theta) d\lambda d\theta, \tag{23}$$

for $k = 1, \ldots, N_\lambda$ and $l = 1, \ldots, N_\theta$.

Equations (21)–(23) are solved by using (21) and (22) to eliminate the U^{n+1} and V^{n+1} dependence in (23). The result is a discretized Helmholtz equation for ϕ_Δ^{n+1} (65), which is nonlinear owing to the presence of a logarithmic term. The discretized Helmholtz equation is linearized and solved using fixed-point iteration. Once ϕ_Δ^{n+1} is computed, U_Δ^{n+1} and V_Δ^{n+1} are updated using the discretized motion equations (21) and (22). The details of this procedure are described in Appendix A.

Some of the latitudinal integrals in (21)–(23) involve functions that may be time-dependent (e.g., $a(\theta)$) and are numerically computed at every timestep. The two-point Gauss quadrature rule, which gives fourth-order approximations, on a uniform spatial grid, is used. Care must be taken in defining the grid partition to which the quadrature rule is applied, since the quadratic spline basis functions, being piecewise polynomials, have discontinuous second derivatives at gridpoints. Therefore, unless the integration grid partition is chosen properly, the assumption associated with the two-point Gauss rule that the integrand has a continuous fourth-derivative within each interval of the integration grid partition may be violated, and, consequently, fourth-order convergence cannot be guaranteed. In order to satisfy the smoothness requirement of the two-point Gauss rule, the integration grid is defined to be twice as refined as the spline grid in each space dimension. That is, the integration partitions are $\Delta_\lambda^I \equiv (\Delta_\lambda \cup \hat{\Delta}_\lambda) \cap [0, 2\pi]$ and $\Delta_\theta^I \equiv (\Delta_\theta \cup \hat{\Delta}_\theta) \cap [-\pi/2, \pi/2]$ in the λ- and θ-directions, respectively. Consequently, the points of discontinuity of the spline basis and test functions do not fall within the integration grid intervals. This complication would not arise if an unstaggered grid were used because in that case all points of discontinuity of the basis and test functions would coincide with the spline grid, and consequently, the spline grid could be used as the integration grid for computing the integral.

4 Rossby Wave Stability

In this section, we perform a stability analysis for the QSG method applied to a simplified version of the SWEs and examine the conditions under which the discretized solutions are stable. We assume, for simplicity, an unstaggered grid and biperiodic boundary conditions. In [15] Staniforth and Mitchell showed that, if Cartesian coordinates are used, if the discretization grid is unstaggered, and if biperiodic boundary conditions are assumed, then the Helmholtz equation should be derived algebraically rather than analytically in order for the standard spatial discretization methods to be neutrally stable for the Rossby waves. We extend their study and demonstrate similar results in spherical coordinates.

The mathematical analysis is done on the simplified SWEs, obtained by linearizing (1)–(3) and assuming constant values for the coefficient associated with the Coriolis terms $f + u\tan\theta/R (= f^*)$, for $u (= u^*)$ and $v (= v^*)$ in the Lagrangian derivative, and for the coefficients $\phi (= \Phi^*)$ and $\theta (= \Theta^*)$ associated with the gradient terms. The simplified SWEs are

$$\frac{du}{dt} - f^* v + \frac{\phi_\lambda}{R\cos\Theta^*} = 0, \tag{24}$$

$$\frac{dv}{dt} + f^* u + \frac{\phi_\theta}{R} = 0, \tag{25}$$

$$\frac{d\phi}{dt} + \frac{\Phi^*}{R}\left(\frac{u_\lambda}{\cos\Theta^*} + v_\theta\right) = 0, \tag{26}$$

where

$$\frac{d}{dt} \equiv \frac{\partial}{\partial t} + \frac{u^*}{R\cos\Theta^*}\frac{\partial}{\partial\lambda} + \frac{v^*}{R}\frac{\partial}{\partial\theta}. \tag{27}$$

Equations (24)–(26) can also be obtained by linearizing (5)–(7), since, with $\cos\Theta^*$ assumed to be constant, U and V are simply constant multiples of u and v.

4.1 The Continuous Case

We first examine the continuous problem. We assume that the solution to (24)–(26) is of the form

$$\begin{pmatrix} u(\lambda,\theta,t) \\ v(\lambda,\theta,t) \\ \phi(\lambda,\theta,t) \end{pmatrix} = \begin{pmatrix} u_0 \\ v_0 \\ \phi_0 \end{pmatrix} e^{\imath(k\lambda+l\theta+\omega t)}, \tag{28}$$

where k and l are the longitudinal and latitudinal wave numbers, respectively, and ω denotes the wave frequency. The Lagrangian frequency ν is defined by $\nu \equiv \frac{u^*}{R\cos\Theta^*} + \frac{v^*}{R} + \omega$.

To compute the Lagrangian frequency of the true solution, (28) is substituted into the simplified SWEs (24)–(26). This results in a 3×3 system of equations for u_0, v_0, and ϕ_0. Then, by setting the determinant of the system to zero, an expression for the Lagrangian frequency ν can be obtained. This expression can then be solved to yield the solutions

$$\nu = 0, \pm\sqrt{\Phi^*\left[\left(\frac{k}{R\cos\Theta^*}\right)^2 + \left(\frac{l}{R}\right)^2\right] + f^{*2}}. \tag{29}$$

The first solution, $\nu = 0$ or $\omega = -u^*/(R\cos\Theta^*) - v^*/R$, corresponds to the solution of interest, i.e., the Rossby waves. The other two frequencies are associated with the gravity waves.

4.2 Algebraic Derivation of the Helmholtz Equation

We now solve the simplified SWEs (24)–(26) using the two-level SLSI time integration method with a finite element spatial discretization scheme. We analyze the stability of the numerical solution when the Helmholtz equation is derived algebraically; in other words, we compare (29) to the Lagrangian frequency of the discretized solution when the divergence terms are eliminated *after* spatial discretization.

Note that the longitudinal mass matrix P and the first derivative matrix P_λ associated with the finite element scheme have common eigenvectors $\vartheta_k = (e^{\imath kj\Delta\lambda})_{j=0}^{N_\lambda-1}$ for $k = 0, \ldots, N_\lambda - 1$. (The definitions for P and P_λ for the QSG method can be found in Appendix A.) Let the associated eigenvalues be p_k and $p_{\lambda k}$, respectively. Thus, we have

$$P\vartheta_k = p_k\vartheta_k, \quad P_\lambda\vartheta_k = p_{\lambda k}\vartheta_k. \tag{30}$$

Similarly, the latitudinal mass matrix Q and the first-derivative matrix Q_θ have common eigenvectors denoted by $\vartheta'_l = (e^{\imath lj\Delta\theta})_{j=0}^{N_\theta-1}$ for $l = 0, \ldots, N_\theta - 1$, with the associated eigenvalues denoted by q_l and $q_{\theta l}$, respectively. Thus, we have

$$Q\vartheta'_l = q_l\vartheta'_l, \quad Q_\theta\vartheta'_l = q_{\theta l}\vartheta'_l. \tag{31}$$

The one-dimensional mass and first derivative matrices arising from the QSG method and many other discretization methods with biperiodic boundary conditions satisfy relations (30) and (31), with the eigenvalues p_k, $p_{\lambda k}$, q_l, and $q_{\theta l}$ set appropriately, since these matrices have the common property that they are circulant and therefore diagonalizable by the discrete Fourier transform matrix.

The discretized solution at time-level t_{n+1} is assumed to be of the following wave form:

$$\begin{pmatrix} u_\Delta^{n+1}(\lambda,\theta) \\ v_\Delta^{n+1}(\lambda,\theta) \\ \phi_\Delta^{n+1}(\lambda,\theta) \end{pmatrix} = \begin{pmatrix} u_0 \\ v_0 \\ \phi_0 \end{pmatrix} e^{\imath(k\lambda+l\theta+\omega(n+1)\Delta t)}. \tag{32}$$

Under the assumptions of this section for constant f^*, u^*, v^*, Φ^*, and Θ^*, the upstream function $\tilde{\psi}^n(\lambda,\theta)$ of a function $\psi(\lambda,\theta,t)$ is given by $\tilde{\psi}^n(\lambda,\theta) \equiv \psi(\lambda - u^*\Delta t/(R\cos\Theta^*), \theta - v^*\Delta t/R, t_n)$. Discretizing the simplified SWEs (24)–(26) in time using the SLSI method yields

$$\frac{u^{n+1}-\tilde{u}^n}{\Delta t} - f^*\left(\frac{v^{n+1}+\tilde{v}^n}{2}\right) + \frac{1}{R\cos\Theta^*}\left(\frac{\phi_\lambda^{n+1}+\tilde{\phi}_\lambda^n}{2}\right) = 0, \tag{33}$$

$$\frac{v^{n+1}-\tilde{v}^n}{\Delta t} + f^*\left(\frac{u^{n+1}+\tilde{u}^n}{2}\right) + \frac{1}{R}\left(\frac{\phi_\theta^{n+1}+\tilde{\phi}_\theta^n}{2}\right) = 0, \tag{34}$$

$$\frac{\phi^{n+1}-\tilde{\phi}^n}{\Delta t} + \frac{\Phi^*}{R\cos\Theta^*}\left(\frac{u_\lambda^{n+1}+\tilde{u}_\lambda^n}{2}\right) + \frac{\Phi^*}{R}\left(\frac{v_\theta^{n+1}+\tilde{v}_\theta^n}{2}\right) = 0. \tag{35}$$

Let E_λ and E_θ be the quadratic spline interpolation matrices that map the vector of spline coefficients (e.g., $\boldsymbol{U}^{n+1}$) to the vector of function values at data points (e.g., $\boldsymbol{u}_\Delta^{n+1}$), where we have used the boldface type to denote vectors. Thus, we have

$(E_\lambda)_{i,j} = \beta_j^P(\lambda_{i-\frac{1}{2}})$ and $(E_\theta)_{i,j} = \beta_j^P(\theta_{i-\frac{1}{2}})$, taking into account that standard quadratic spline interpolation uses the midpoints as data points. The spatially discretized form of (33) can be written as

$$(P \otimes Q)(E_\lambda \otimes E_\theta)^{-1} \left(\frac{\boldsymbol{u}_\Delta^{n+1} - \tilde{\boldsymbol{u}}_\Delta^n}{\Delta t} \right) - f^*(P \otimes Q)(E_\lambda \otimes E_\theta)^{-1} \left(\frac{\boldsymbol{v}_\Delta^{n+1} + \tilde{\boldsymbol{v}}_\Delta^n}{2} \right) + \frac{1}{R\cos\Theta^*}(P_\lambda \otimes Q)(E_\lambda \otimes E_\theta)^{-1} \left(\frac{\boldsymbol{\phi}_\Delta^{n+1} + \tilde{\boldsymbol{\phi}}_\Delta^n}{2} \right) = 0. \tag{36}$$

With biperiodic boundary conditions, all matrices in parentheses in (36) commute. By pre-multiplying (36) through by $(E_\lambda \otimes E_\theta)$ and simplifying we obtain

$$(P \otimes Q) \left(\frac{\boldsymbol{u}_\Delta^{n+1} - \tilde{\boldsymbol{u}}_\Delta^n}{\Delta t} \right) - f^*(P \otimes Q) \left(\frac{\boldsymbol{v}_\Delta^{n+1} + \tilde{\boldsymbol{v}}_\Delta^n}{2} \right) + \frac{1}{R\cos\Theta^*}(P_\lambda \otimes Q) \left(\frac{\boldsymbol{\phi}_\Delta^{n+1} + \tilde{\boldsymbol{\phi}}_\Delta^n}{2} \right) = 0. \tag{37}$$

Following a similar procedure, we obtain the following discretized form of (34) and (35):

$$(P \otimes Q) \left(\frac{\boldsymbol{v}_\Delta^{n+1} - \tilde{\boldsymbol{v}}_\Delta^n}{\Delta t} \right) + f^*(P \otimes Q) \left(\frac{\boldsymbol{u}_\Delta^{n+1} + \tilde{\boldsymbol{u}}_\Delta^n}{2} \right) + \frac{1}{R}(P \otimes Q_\theta) \left(\frac{\boldsymbol{\phi}_\Delta^{n+1} + \tilde{\boldsymbol{\phi}}_\Delta^n}{2} \right) = 0, \tag{38}$$

$$(P \otimes Q) \left(\frac{\boldsymbol{\phi}_\Delta^{n+1} - \tilde{\boldsymbol{\phi}}_\Delta^n}{\Delta t} \right) + \frac{\Phi^*}{R\cos\Theta^*}(P_\lambda \otimes Q) \left(\frac{\boldsymbol{u}_\Delta^{n+1} + \tilde{\boldsymbol{u}}_\Delta^n}{2} \right) + \frac{\Phi^*}{R}(P \otimes Q_\theta) \left(\frac{\boldsymbol{v}_\Delta^{n+1} + \tilde{\boldsymbol{v}}_\Delta^n}{2} \right) = 0. \tag{39}$$

Substituting the solution (32) into (37)–(39) and using relations (30) and (31), we obtain

$$pq \left(\frac{e^{\imath\nu\Delta t} - 1}{\Delta t} \right) u_0 - f^* pq \left(\frac{e^{\imath\nu\Delta t} + 1}{2} \right) v_0 + \frac{p_\lambda q}{R\cos\Theta^*} \left(\frac{e^{\imath\nu\Delta t} + 1}{2} \right) \phi_0 = 0, \tag{40}$$

$$pq \left(\frac{e^{\imath\nu\Delta t} - 1}{\Delta t} \right) v_0 + f^* pq \left(\frac{e^{\imath\nu\Delta t} + 1}{2} \right) u_0 + \frac{pq_\theta}{R} \left(\frac{e^{\imath\nu\Delta t} + 1}{2} \right) \phi_0 = 0, \tag{41}$$

$$pq \left(\frac{e^{\imath\nu\Delta t} - 1}{\Delta t} \right) \phi_0 + \frac{\Phi^*}{R\cos\Theta^*} p_\lambda q \left(\frac{e^{\imath\nu\Delta t} + 1}{2} \right) u_0 + \frac{\Phi^*}{R} pq_\theta \left(\frac{e^{\imath\nu\Delta t} + 1}{2} \right) v_0 = 0, \tag{42}$$

where, in order to simplify the notation, we have omitted the subscripts k and l from the eigenvalues p_k, $p_{\lambda k}$, q_l and $q_{\theta l}$. Equations (40)–(42) hold for $k = 0, \ldots, N_\lambda - 1$ and $l = 0, \ldots, N_\theta - 1$.

Equations (40)–(42) form a 3×3 system for u_0, v_0, and ϕ_0. By setting the determinant of the resulting system to zero to find a nontrivial solution to (40)–(42), an equation for $e^{\imath\nu\Delta t}$ can be obtained. The roots are found to be

$$e^{\imath\nu\Delta t} = 1, \ \frac{4A - B \pm 4\imath\sqrt{AB}}{4A + B}, \tag{43}$$

where $A = (R\cos\Theta^* pq)^2$ and $B = (\Delta t f^* R \cos\Theta^* pq)^2 - \Delta t^2 \Phi^* (p_\lambda^2 q^2 + \cos^2\Theta^* p^2 q_\theta^2)$. Since $e^{\imath\nu\Delta t} = 1$, or $\nu = 0$, is one of the solutions, the numerical scheme is neutrally stable for the Rossby waves. The solutions for the gravity waves are also bounded since

$$|e^{\imath\nu\Delta t}| = \left| \frac{4A - B \pm 4\imath\sqrt{AB}}{4A + B} \right| = 1. \tag{44}$$

Therefore, the numerical scheme is also stable for the gravitational modes. Note that the results of this analysis hold for the QSG method, the OQSC methods presented in the companion paper [10], as well as any discretization method of which the operators satisfy (30) and (31) for some p_k, $p_{\lambda k}$, q_l, and $q_{\theta l}$.

4.3 Analytic Derivation of the Helmholtz Equation

As noted in Appendix A, (65) is only an approximation to a Helmholtz equation. A genuine Helmholtz equation may be derived by eliminating the wind velocity dependence from the continuity equation (26) *before* spatial discretization. We solve (33) and (34) for u^{n+1} and v^{n+1} in terms of ϕ_λ^{n+1} and ϕ_θ^{n+1} as we did in Section 2. Then, taking the λ- and θ-derivatives of the resulting equations, we get

$$u_\lambda^{n+1} = \quad a\tilde{u}_\lambda^n + b\tilde{v}_\lambda^n - \frac{\Delta t}{R}\left[\quad c\left(\frac{\phi_{\lambda\lambda}^{n+1} + \tilde{\phi}_{\lambda\lambda}^n}{2\cos\Theta^*}\right) + d\left(\frac{\phi_{\theta\lambda}^{n+1} + \tilde{\phi}_{\theta\lambda}^n}{2}\right)\right], \tag{45}$$

$$v_\theta^{n+1} = -b\tilde{u}_\theta^n + a\tilde{v}_\theta^n - \frac{\Delta t}{R}\left[-d\left(\frac{\phi_{\lambda\theta}^{n+1} + \tilde{\phi}_{\lambda\theta}^n}{2\cos\Theta^*}\right) + c\left(\frac{\phi_{\theta\theta}^{n+1} + \tilde{\phi}_{\theta\theta}^n}{2}\right)\right], \tag{46}$$

where a, b, c, and d are defined in (15) with $\tilde{f} = f^*$. Equations (45) and (46) can then be used to eliminate the unknown divergence terms from the continuity equation (35) to yield

$$(\phi^{n+1} - \tilde{\phi}^n) - \Phi^* \left(\frac{\Delta t}{2R}\right)^2 c \left[\frac{\phi_{\lambda\lambda}^{n+1} + \tilde{\phi}_{\lambda\lambda}^n}{\cos^2\Theta^*} + \phi_{\theta\theta}^{n+1} + \tilde{\phi}_{\theta\theta}^n\right] + \Phi^* \left(\frac{\Delta t}{2R}\right) \left[\frac{1+a}{\cos\Theta^*}\tilde{u}_\lambda^n - b\tilde{u}_\theta^n + \frac{b}{\cos\Theta^*}\tilde{v}_\lambda^n + (1+a)\tilde{v}_\theta^n\right] = 0. \quad (47)$$

Let $P_{\lambda\lambda}$ and $Q_{\theta\theta}$ denote the stiffness matrices in the λ- and θ-directions, respectively; note that $P_{\lambda\lambda}$ and $Q_{\theta\theta}$ are circulant. Let $p_{\lambda\lambda_k}$ and $q_{\theta\theta_l}$ be the eigenvalues of $P_{\lambda\lambda}$ and $Q_{\theta\theta}$ associated with the eigenvectors ϑ_k and ϑ'_l, respectively, as defined in Section 4.2. The spatially discretized form of (47) is

$$(P\otimes Q)(\boldsymbol{\phi}_\Delta^{n+1} - \tilde{\boldsymbol{\phi}}_\Delta^n) - \Phi^*\left(\frac{\Delta t}{2R}\right)^2 c\left[\frac{P_{\lambda\lambda}\otimes Q}{\cos^2\Theta^*} + (P\otimes Q_{\theta\theta})\right](\boldsymbol{\phi}_\Delta^{n+1} + \tilde{\boldsymbol{\phi}}_\Delta^n) + \Phi^*\left(\frac{\Delta t}{2R}\right)\left\{\left[\frac{1+a}{\cos\Theta^*}(P_\lambda\otimes Q) - b(P\otimes Q_\theta)\right]\tilde{\boldsymbol{u}}_\Delta^n + \left[\frac{b}{\cos\Theta^*}(P_\lambda\otimes Q) + (1+a)(P\otimes Q_\theta)\right]\tilde{\boldsymbol{v}}_\Delta^n\right\} = 0. \quad (48)$$

Substituting the solution (32) into the above equation, rewriting the resulting equation in terms of the eigenvalues, and simplifying, we obtain

$$pq(e^{\imath\nu\Delta t} - 1)\phi_0 - \Phi^*\left(\frac{\Delta t}{2R}\right)^2 c\left[\frac{p_{\lambda\lambda}q}{\cos^2\Theta^*} + pq_{\theta\theta}\right](e^{\imath\nu\Delta t}+1)\phi_0 + \Phi^*\left(\frac{\Delta t}{2R}\right)\left\{\left[\frac{1+a}{\cos\Theta^*}p_\lambda q - bpq_\theta\right]u_0 + \left[\frac{b}{\cos\Theta^*}p_\lambda q + (1+a)pq_\theta\right]v_0\right\} = 0. \quad (49)$$

By setting the determinant of equations (40), (41), and (49) to zero, an equation for $e^{\imath\nu\Delta t}$ is obtained. We found that $e^{\imath\nu\Delta t} = 1$ is no longer a solution to this frequency equation. This result, which is consistent with [15], implies that an undesirable phase shift has been introduced into the Rossby mode. Therefore, in our formulation, the Helmholtz equation is derived algebraically to preserve the neutral stability of the Rossby waves.

5 Numerical Results

We now demonstrate numerically the stability of our method with respect to the Rossby waves and study its convergence behaviour. In the companion paper [10], we compare the computational cost of the QSG method to those of the linear spline Galerkin method and three quadratic spline collocation methods; we also show that when the spatial discretization is done on the C-grid, gravity waves propagate in the proper directions.

5.1 The Equatorial Rossby Wave

The equatorial solitary wave described by Boyd [2] is frequently used to test the stability of a numerical method for the SWEs with respect to Rossby waves [9, 11]. To use this test problem on our model, we transform Boyd's formulation, originally written in Cartesian coordinates, to spherical coordinates. We adopt Boyd's notation and let ϕ' be the geopotential perturbation; that is, if we denote the reference geopotential by the constant ϕ^*, then $\phi = \phi^* + \phi'$. The resulting nondimensional nonlinear SWEs on the equatorial β-plane are

$$\frac{du}{dt} - R\theta v + \frac{\phi'_\lambda}{R\cos\theta} = 0, \tag{50}$$

$$\frac{dv}{dt} + R\theta u + \frac{\phi'_\theta}{R} = 0, \tag{51}$$

$$\frac{d}{dt}\log(\phi^* + \phi') + \frac{1}{R\cos\theta}[u_\lambda + (v\cos\theta)_\theta] = 0. \tag{52}$$

The above equations are integrated with the following initial conditions

$$u(\lambda,\theta,0) = \eta(R\lambda)\frac{-9+6(R\theta)^2}{4}e^{-(R\theta)^2/2}, \tag{53}$$

$$v(\lambda,\theta,0) = 2\frac{\partial\eta(R\lambda)}{\partial\lambda}R\theta e^{-(R\theta)^2/2}, \tag{54}$$

$$\phi'(\lambda,\theta,0) = \eta(R\lambda)\frac{3+6(R\theta)^2}{4}e^{-(R\theta)^2/2}, \tag{55}$$

where $\eta(\lambda) = A\mathrm{sech}^2(B\lambda)$, $A = 0.12$, and $B = 0.394$. In this study, a timestep of $\Delta t = 0.05$ non-dimensional units (approximately two hours in simulated time) is used. Experiments are conducted with grids of sizes $N \times N$, with $N = 32, 64$, and 128. Variables are staggered using the C-grid.

The equations describe an equatorial soliton which slowly propagates westward, with no change in shape. Figure 2 shows the result of a long simulation on a 128×128 grid for 24 non-dimensional time units, which corresponds to approximately 41 days in simulated time. In our experiments, the height fields lost about 5% of their initial amplitude which traveled eastward as equatorial Kelvin waves. We believe this is because the initial conditions (53)–(55) are inexact, as suggested by Iskandarani et al. [9] and Ma [11]. The soliton propagated westward, as predicted, with little change in amplitude or phase, thereby confirming our analysis that spatial discretization with the QSG method is neutrally stable for the Rossby modes. The small amount of dispersion present can be explained as follows. With semi-Lagrangian integration, variables are needed at departures points, which are usually off-mesh points. Cubic Lagrange interpolation is used to estimate function values at departure points. Spatial interpolation causes damping and phase shift. Compared to linear or quadratic interpolation, the effects introduced by cubic interpolation are less severe. The finer the mesh, the less prominent the damping and dispersive effects should be [1, 12]. This has been confirmed in the numerical tests described in [16].

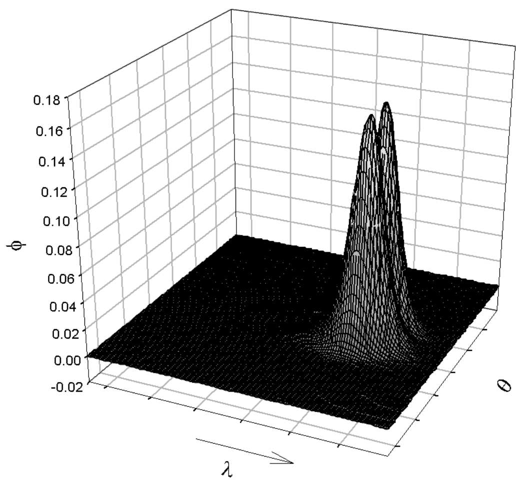

Figure 2: Simulation by the QSG method of a Rossby soliton traveling westward (in the direction indicated by the arrow, which is the direction of decreasing λ values) on a 128×128 grid for 24 time units (approximately 41 days).

5.2 Convergence Tests

Since no analytical solution is known for the general form of the two-dimensional SWEs (5)–(7), we introduce into the equations forcing terms constructed in such a way that an analytical solution is known a priori. The modified SWEs are

$$\frac{dU}{dt} - fV + \frac{\phi_\lambda}{R^2} = F_u, \tag{56}$$

$$\frac{dV}{dt} + fU + \frac{\cos\theta}{R^2}\phi_\theta + \frac{\sin\theta}{\cos^2\theta}(U^2+V^2) = F_v, \tag{57}$$

$$\cos\theta\frac{d}{dt}\log\phi + \left[\frac{U_\lambda}{\cos\theta} + V_\theta\right] = F_\phi, \tag{58}$$

where F_u, F_v, and F_ϕ are known functions of λ, θ, and t, defined to yield the following solution:

$$\begin{aligned} U(\lambda,\theta,t) = {} & \frac{u_0}{R}\left[1-\exp\left(-\frac{t}{T_0}\right)\right]\cos\theta\times \\ & \left\{\exp\left[-a_3(\lambda-\lambda_0-d_0)^2 - a_3(\theta-\theta_0)^2\right]\right. \\ & \left. -\exp\left[-a_3(\lambda-\lambda_0+d_0)^2 - a_3(\theta-\theta_0)^2\right]\right\}, \end{aligned} \tag{59}$$

$$\begin{aligned} V(\lambda,\theta,t) = {} & \frac{v_0}{R}\left[1-\exp\left(-\frac{t}{T_0}\right)\right]\cos\theta\times \\ & \left\{\exp\left[-a_3(\lambda-\lambda_0)^2 - a_3(\theta-\theta_0-d_0)^2\right]\right. \\ & \left. -\exp\left[-a_3(\lambda-\lambda_0)^2 - a_3(\theta-\theta_0+d_0)^2\right]\right\}. \end{aligned} \tag{60}$$

$$\begin{aligned} \phi(\lambda,\theta,t) = {} & \phi_0\left\{\exp\left[-a_1(\lambda-\lambda_0)^2 - a_1(\theta-\theta_0)^2\right]\right. \\ & \left. -\log_e\left(1+\frac{t}{T_0}\right)\exp\left[-a_2(\lambda-\lambda_0)^2 - a_2(\theta-\theta_0)^2\right]\right\}, \end{aligned} \tag{61}$$

The above functions are designed to resemble the solutions of the original SWEs (5)–(7) when these functions are given as initial conditions. In our experiments, we chose the following values for the constants

$$\begin{array}{lll} \phi_0 = 50\,g, & u_0 = v_0 = 3/2, & T_0 = 12000\text{ s}, \\ a_1 = 10, & a_2 = 60, & a_3 = 25, \\ \lambda_0 = \pi, & \theta_0 = 0, & d_0 = \pi/10, \end{array}$$

where $g = 9.80616\,\text{ms}^{-2}$ is the gravitational constant. The system was integrated on the sphere $\{(\lambda,\theta) \in [0,2\pi]\times[-\pi/2,\pi/2]\}$, using timesteps of 60 seconds (i.e.,

$\Delta t = 60$ s). The solution for ϕ was computed at $T = 16\frac{2}{3}$ hours and compared to the reference solution (61).

We solved the system using the linear spline Galerkin (LSG) method and the QSG method with grid sizes 32×32, 64×64, 128×128, and 256×256. The Arakawa C-grid was used in our discretization. The convergence results are shown in Table 1, where $N = N_\lambda = N_\theta$ is the number of subintervals in one dimension. Two error norms are computed for QSG: the local error norms at gridpoints and the global error norms, labelled "$|e|$" and "$||e||$," respectively, in Table 1. Only the local error norms at gridpoints are shown for the LSG method, because, unlike the QSG method, the LSG method is not expected to exhibit superconvergence at special points (i.e., the local and global errors for the LSG method are of the same order). The error norms are computed as ratios of the 2-norm of the difference between the numerical solution and the reference solution, to the 2-norm of the reference solution. The errors reflect the local truncation error of each method, which, in these experiments, is found to be roughly second order for LSG, and roughly fourth order for QSG locally at gridpoints and third order globally. We estimate the convergence order of the errors by $p \equiv -\log\left(\frac{e_1}{e_2}\right) / \log\left(\frac{N_1}{N_2}\right)$ where e_1 and e_2 are error norms corresponding to grid sizes N_1 and N_2, respectively. The computed values for p are listed in the columns labelled "p" in Table 1.

Table 1: Observed errors and respective orders of convergence for the QSG and LSG methods with various spatial grid sizes.

	QSG				LSG	
N	$\|e\|$	p	$\|\|e\|\|$	p	$\|e\|$	p
32	1.756e-1	—	1.832e-1	—	2.486e-1	—
64	1.154e-2	3.93	2.322e-2	2.98	6.906e-2	1.85
128	8.779e-4	3.72	3.498e-3	2.86	1.887e-2	1.87
256	7.175e-5	3.61	4.375e-4	2.87	4.917e-3	1.94

6 Remarks

Spatial discretization schemes commonly used in meteorological applications are currently limited to spectral methods or low-order finite difference/finite element methods [3, 4, 8, 17, 18]. Spectral methods, which yield high-order solutions, give rise to dense matrices, therefore their parallel implementation is not scalable. In contrast, finite element methods have more potential for parallelism and give rise to reasonably scalable parallel implementations. Therefore, high-order finite element methods seem to be a viable alternative to spectral methods. In this paper, we present a method that combines the two-level SLSI time integration method and the QSG method for solving the SWEs on the sphere. We regard this method as a step towards high-order finite element methods that may be more efficient than spectral methods for solving partial differential equations on the sphere. A worthwhile challenge is the extension of the techniques presented in this paper and the companion paper [10] for quadratic spline collocation methods to higher-order spline methods, which may yield solutions with accuracy comparable to spectral methods.

Appendix A

In this appendix, we define the discretization matrices and present details of the procedure with which (21)–(23) are solved. The mass and first derivative matrices in the longitudinal dimension, where functions are assumed to be periodic, may be written in terms of the adjusted basis functions as

$$\begin{array}{ll}
P_{i,j} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \beta_j^P(\lambda)\beta_i^P(\lambda)d\lambda, & P_{\lambda i,j} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \beta_j^{P'}(\lambda)\beta_i^P(\lambda)d\lambda, \\
\hat{P}_{i,j} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \hat{\beta}_j^P(\lambda)\hat{\beta}_i^P(\lambda)d\lambda, & \hat{P}_{\lambda_{i,j}} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \hat{\beta}_j^{P'}(\lambda)\hat{\beta}_i^P(\lambda)d\lambda, \\
P_{i,j}^{\Delta\to\hat{\Delta}} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \beta_j^P(\lambda)\hat{\beta}_i^P(\lambda)d\lambda, & P_{\lambda}{}_{i,j}^{\Delta\to\hat{\Delta}} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \beta_j^{P'}(\lambda)\hat{\beta}_i^P(\lambda)d\lambda, \\
P_{i,j}^{\hat{\Delta}\to\Delta} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \hat{\beta}_j^P(\lambda)\beta_i^P(\lambda)d\lambda, & P_{\lambda}{}_{i,j}^{\hat{\Delta}\to\Delta} = \int_{\lambda_0}^{\lambda_{N_\lambda}} \hat{\beta}_j^{P'}(\lambda)\beta_i^P(\lambda)d\lambda,
\end{array}$$

for $i,\ j = 1, \ldots, N_\lambda$. Since the basis functions are periodic in the longitudinal dimension, $\hat{\beta}_j(\lambda) = \beta_j(\lambda - \Delta\lambda/2)$; it follows that $\hat{P}_{i,j} = P_{i,j}$, and $\hat{P}_{\lambda_{i,j}} = P_{\lambda_{i,j}}$. Therefore, throughout the paper, we drop $\hat{P}_{i,j}$ and $\hat{P}_{\lambda_{i,j}}$ and use $P_{i,j}$ and $P_{\lambda_{i,j}}$ instead.

In the mathematical analysis in Section 4, we use periodic boundary conditions in the latitudinal dimension as well. When periodic boundary conditions are applied in the latitudinal dimension, the mass and first derivative matrices in the latitudinal dimension are defined similarly as the corresponding matrices in the longitudinal dimension, and are written without a superscript.

Recall that, in the formulation and implementation of the method, homogeneous Dirichlet latitudinal boundary conditions are used for U_Δ^{n+1} and V_Δ^{n+1}, and Neumann ones are used for ϕ_Δ^{n+1}. Integrals in the latitudinal dimension may involve functions other than the basis functions, such as $a(\theta)$, $b(\theta)$, $c(\theta)$ or $d(\theta)$ defined in (15). We first define in terms of β_j^D the matrices

$$\begin{array}{ll} Q^D_{i,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^D(\theta)\beta_i^D(\theta)d\theta, & Q^D_{\theta\ i,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^{D'}(\theta)\beta_i^D(\theta)d\theta, \\ Q^{D,\Delta\to\hat\Delta}_{k,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^D(\theta)\hat\beta_k^D(\theta)d\theta, & Q_{\theta}{}^{D,\Delta\to\hat\Delta}_{k,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^{D'}(\theta)\hat\beta_k^D(\theta)d\theta, \end{array}$$

for $i,\ j = 1,\ldots,N_\theta$ and $k = 0,\ldots,N_\theta$. Note that Q^D and Q^D_θ are $N_\theta \times N_\theta$ matrices, whereas $Q^{D,\Delta\to\hat\Delta}$ and $Q_\theta^{D,\Delta\to\hat\Delta}$ are $(N_\theta+1)\times N_\theta$. We then consider an arbitrary function $g(\theta)$ and use the following notation to define the associated mass and first derivative matrices

$$\begin{array}{l} \{Q^D \circ g\}_{i,j} = \int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^D(\theta)\beta_i^D(\theta)d\theta, \\ \{Q^D_\theta \circ g\}_{i,j} = \int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^{D'}(\theta)\beta_i^D(\theta)d\theta, \\ \{Q^{D,\Delta\to\hat\Delta} \circ g\}_{k,j} = \int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^D(\theta)\hat\beta_k^D(\theta)d\theta, \\ \{Q_\theta^{D,\Delta\to\hat\Delta} \circ g\}_{k,j} = \int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^{D'}(\theta)\hat\beta_k^D(\theta)d\theta, \end{array}$$

for $i,\ j = 1,\ldots,N_\theta$ and $k = 0,\ldots,N_\theta$. Note that in the above definition the symbol $\circ$ does not mean function composition.

The mass and first derivative matrices for ϕ_Δ^{n+1} in the latitudinal dimension are identified by the superscript N, and expressed in terms of β_j^N. We first define the matrices

$$\begin{array}{ll} Q^N_{i,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^N(\theta)\beta_i^N(\theta)d\theta, & Q^N_{\theta\ i,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^{N'}(\theta)\beta_i^N(\theta)d\theta, \\ Q^{N,\Delta\to\hat\Delta}_{k,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^N(\theta)\hat\beta_k^D(\theta)d\theta, & Q_{\theta}{}^{N,\Delta\to\hat\Delta}_{k,j} = \int_{\theta_0}^{\theta_{N_\theta}} \beta_j^{N'}(\theta)\hat\beta_k^D(\theta)d\theta, \end{array}$$

for $i,\ j = 1,\ldots,N_\theta$ and $k = 0,\ldots,N_\theta$. Note that Q^N and Q^N_θ are $N_\theta \times N_\theta$ matrices, whereas $Q^{N,\Delta\to\hat\Delta}$ and $Q_\theta^{N,\Delta\to\hat\Delta}$ are $(N_\theta+1)\times N_\theta$. The associated mass and first derivative matrices for ϕ_Δ^{n+1} in the latitudinal dimension are defined by

$$\begin{aligned}
\{Q^N \circ g\}_{i,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^N(\theta)\beta_i^N(\theta)d\theta, \\
\{Q_\theta^N \circ g\}_{i,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^{N'}(\theta)\beta_i^N(\theta)d\theta, \\
\{Q^{N,\Delta\to\hat\Delta} \circ g\}_{k,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^N(\theta)\hat\beta_k^D(\theta)d\theta, \\
\{Q_\theta^{N,\Delta\to\hat\Delta} \circ g\}_{k,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\beta_j^{N'}(\theta)\hat\beta_k^D(\theta)d\theta,
\end{aligned}$$

for $i,\ j = 1, \ldots, N_\theta$ and $k = 0, \ldots, N_\theta$.

The mass and first derivative matrices for V_Δ^{n+1} in the latitudinal dimension are identified by the superscript D, and the hat symbol or the superscript $\hat\Delta \to \Delta$. We first define the matrices

$$\begin{aligned}
\{\hat Q^D\}_{i,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} \hat\beta_j^D(\theta)\hat\beta_i^D(\theta)d\theta, &\quad \{\hat Q_\theta^D\}_{i,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\hat\beta_j^{D'}(\theta)\hat\beta_i^D(\theta)d\theta, \\
\{Q^{D,\hat\Delta\to\Delta}\}_{k,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} \hat\beta_j^D(\theta)\beta_k^D(\theta)d\theta, &\quad \{Q_\theta^{D,\hat\Delta\to\Delta}\}_{k,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} \hat\beta_j'^D(\theta)\beta_k^N(\theta)d\theta,
\end{aligned}$$

for $i,\ j = 0, \ldots, N_\theta$ and $k = 1, \ldots, N_\theta$. Note that $\hat Q^D$ and $\hat Q_\theta^D$ are $(N_\theta+1)\times(N_\theta+1)$ matrices, whereas $Q^{D,\hat\Delta\to\Delta}$ and $Q_\theta^{D,\hat\Delta\to\Delta}$ are $N_\theta \times (N_\theta+1)$. The associated mass and first derivative matrices are then defined by

$$\begin{aligned}
\{\hat Q^D \circ g\}_{i,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\hat\beta_j^D(\theta)\hat\beta_i^D(\theta)d\theta, \\
\{\hat Q_\theta^D \circ g\}_{i,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\hat\beta_j^{D'}(\theta)\hat\beta_i^D(\theta)d\theta, \\
\{Q^{D,\hat\Delta\to\Delta} \circ g\}_{k,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\hat\beta_j^D(\theta)\beta_k^D(\theta)d\theta, \\
\{Q_\theta^{D,\hat\Delta\to\Delta} \circ g\}_{k,j} &= \textstyle\int_{\theta_0}^{\theta_{N_\theta}} g(\theta)\hat\beta_j'^D(\theta)\beta_k^N(\theta)d\theta,
\end{aligned}$$

for $i,\ j = 0, \ldots, N_\theta$ and $k = 1, \ldots, N_\theta$.

Equations (21)–(23) are expressed in matrix form using the mass and first derivative matrices defined above, and then solved by using (21) and (22) to eliminate the divergence terms (i.e., U^{n+1} and V^{n+1}) in (23). The resulting discretized Helmholtz equation for ϕ_Δ^{n+1}, which is nonlinear due to the presence of a logarithmic term, is linearized and solved using fixed-point iteration. Once ϕ_Δ^{n+1} is computed, U_Δ^{n+1} and V_Δ^{n+1} are updated using the discretized motion equations (21) and (22). The details of this procedure are described below.

Let $\tilde{\boldsymbol{R}}_u^n$, $\tilde{\boldsymbol{R}}_v^n$, and $\tilde{\boldsymbol{R}}_\phi^n$ denote the vectors of values, arranged in natural ordering, of $\int\int \tilde R_u^n(\lambda,\theta)\hat\beta_i^P(\lambda)\beta_j^D(\theta)d\lambda d\theta$, $\int\int \tilde R_v^n(\lambda,\theta)\beta_i^P(\lambda)\hat\beta_j^D(\theta)d\lambda d\theta$, and

$\int\int \tilde{R}_\phi^n(\lambda,\theta)\beta_i^P(\lambda)\beta_j^D(\theta)d\lambda d\theta$, respectively. Equations (21)–(23) can be expressed in tensor product form and rearranged as

$$(P\otimes Q^D)\mathbf{U}^{n+1}+\left(\frac{\Delta t}{2R^2}\right)\left[(P_\lambda^{\Delta\to\hat{\Delta}}\otimes Q^N\circ c)+(P^{\Delta\to\hat{\Delta}}\otimes Q_\theta^N\circ(d\cos\tilde{\theta}))\right]\mathbf{\Phi}^{n+1}$$
$$=\tilde{\boldsymbol{R}}_u^n, \tag{62}$$

$$(P\otimes\hat{Q}^D)\mathbf{V}^{n+1}+\left(\frac{\Delta t}{2R^2}\right)\left[-(P_\lambda\otimes Q^{N,\Delta\to\hat{\Delta}}\circ d)+(P\otimes Q_\theta^{N,\Delta\to\hat{\Delta}}\circ(c\cos\tilde{\theta}))\right]\mathbf{\Phi}^{n+1}$$
$$=\tilde{\boldsymbol{R}}_v^n, \tag{63}$$

$$(P\otimes Q^N\circ\cos\tilde{\theta})\mathbf{\Gamma}^{n+1}+\left(\frac{\Delta t}{2}\right)\left[\left(P_\lambda^{\hat{\Delta}\to\Delta}\otimes Q^D\circ\left(\frac{1}{\cos\tilde{\theta}}\right)\right)\mathbf{U}^{n+1}\right.$$
$$\left.+\left(P\otimes Q_\theta^{D,\hat{\Delta}\to\Delta}\right)\mathbf{V}^{n+1}\right]=\tilde{\boldsymbol{R}}_\phi^n. \tag{64}$$

The next step involves eliminating the $\mathbf{U}^{n+1}$ and $\mathbf{V}^{n+1}$ dependence from the discretized continuity equation (64) using (62) and (63). By pre-multiplying (62) and (63) by the inverses of $(P\otimes Q^D)$ and $(P\otimes\hat{Q}^D)$, respectively, and rearranging, $\mathbf{U}^{n+1}$ and $\mathbf{V}^{n+1}$ can be expressed in terms of $\mathbf{\Phi}^{n+1}$ and other quantities that are known. Then, pre-multiplied by $(P_\lambda^{\hat{\Delta}\to\Delta}\otimes Q^D\circ(\frac{1}{\cos\tilde{\theta}}))$ and $(P\otimes Q_\theta^{D,\hat{\Delta}\to\Delta})$, respectively, the resulting equations can be substituted into (64) to eliminate the divergence terms and yield the equation

$$(P\otimes Q^N\circ\cos\tilde{\theta})\mathbf{\Gamma}^{n+1}+\left(\frac{\Delta t}{2}\right)\left(P_\lambda^{\hat{\Delta}\to\Delta}\otimes Q^D\circ\left(\frac{1}{\cos\tilde{\theta}}\right)\right)(P\otimes Q^D)^{-1}\times$$
$$\left\{\tilde{\boldsymbol{R}}_u^n-\left(\frac{\Delta t}{2R^2}\right)\left[(P_\lambda^{\Delta\to\hat{\Delta}}\otimes Q^N\circ c)+(P^{\Delta\to\hat{\Delta}}\otimes Q_\theta^N\circ(d\cos\tilde{\theta}))\right]\mathbf{\Phi}^{n+1}\right\}$$
$$+\left(\frac{\Delta t}{2}\right)\left(P\otimes Q_\theta^{D,\hat{\Delta}\to\Delta}\right)(P\otimes\hat{Q}^D)^{-1}\times$$
$$\left\{\tilde{\boldsymbol{R}}_v^n-\left(\frac{\Delta t}{2R^2}\right)\left[-(P_\lambda\otimes Q^{N,\Delta\to\hat{\Delta}}\circ d)+(P\otimes Q_\theta^{N,\Delta\to\hat{\Delta}}\circ(c\cos\tilde{\theta}))\right]\mathbf{\Phi}^{n+1}\right\}$$
$$=\tilde{\boldsymbol{R}}_\phi^n, \tag{65}$$

which is an approximation to a Helmholtz equation. Equation (65) is nonlinear because of the nonlinear relationship between $\mathbf{\Gamma}^{n+1}$ and $\mathbf{\Phi}^{n+1}$ (see Section 3). Hence, (65) is solved using fixed-point iteration. Following the approach of [6], we first linearize (65) by moving the nonlinear term $(P\otimes Q^N\circ\cos\tilde{\theta})\mathbf{\Gamma}^{n+1}$ and the known terms to the right side. Then a fixed-point relation is obtained by adding $(P\otimes Q^N)\mathbf{\Phi}^{n+1}$

to both sides. Therefore, at the k-th iteration, we solve for $\mathbf{\Phi}^{n+1,[k]}$ in the following equation:

$$
\begin{aligned}
&\Bigg\{(P \otimes Q^N) - \left(\frac{\Delta t}{2R}\right)^2 \left(P_\lambda^{\hat{\Delta}\to\Delta} \otimes Q^D \circ \left(\frac{1}{\cos\tilde{\theta}}\right)\right) (P \otimes Q^D)^{-1} \times \\
&\qquad\qquad \left[\left(P_\lambda^{\Delta\to\hat{\Delta}} \otimes Q^N \circ c\right) + (P^{\Delta\to\hat{\Delta}} \otimes Q_\theta^N \circ (d\cos\tilde{\theta}))\right] \\
&\qquad - \left(\frac{\Delta t}{2R}\right)^2 \left(P \otimes Q_\theta^{D,\hat{\Delta}\to\Delta}\right) (P \otimes \hat{Q}^D)^{-1} \times \\
&\qquad\qquad \left[-(P_\lambda \otimes Q^{N,\Delta\to\hat{\Delta}} \circ d) + (P \otimes Q_\theta^{N,\Delta\to\hat{\Delta}} \circ (c\cos\tilde{\theta}))\right]\Bigg\} \mathbf{\Phi}^{n+1,[k]} \\
&= (P \otimes Q^N)\mathbf{\Phi}^{n+1,[k-1]} - (P \otimes Q^N \circ \cos\tilde{\theta})\mathbf{\Gamma}^{n+1,[k-1]} \\
&\qquad - \left(\frac{\Delta t}{2}\right) \left(P_\lambda^{\hat{\Delta}\to\Delta} \otimes Q^D \circ \left(\frac{1}{\cos\tilde{\theta}}\right)\right) (P \otimes Q^D)^{-1} \tilde{\boldsymbol{R}}_u^n \\
&\qquad - \left(\frac{\Delta t}{2}\right) \left(P \otimes Q_\theta^{D,\hat{\Delta}\to\Delta}\right) (P \otimes \hat{Q}^D)^{-1} \tilde{\boldsymbol{R}}_v^n + \tilde{\boldsymbol{R}}_\phi^n.
\end{aligned}
\tag{66}
$$

When the QSG method is applied to the linearized Helmholtz equation (66), the inner products in the θ-dimension are first computed using the two-point Gauss rule as described in Section 3.2 before the iteration (66) is started. When applying the fixed-point iteration (66) using an initial guess formed by linear extrapolation $\mathbf{\Phi}^{n+1,[0]} = 2\mathbf{\Phi}^n - \mathbf{\Phi}^{n-1}$ two fixed-point iterations usually suffice. Then at each iteration of (66), the linear system is solved with the conjugate gradient iterative method to yield $\mathbf{\Phi}^{n+1,[k]}$. The solution $\mathbf{\Phi}^{n+1}$ is used to compute $\mathbf{U}^{n+1}$ and $\mathbf{V}^{n+1}$ using the discretized motion equations (62) and (63).

References

[1] J. R. Bates and A. McDonald. Multiply-upstream, semi-Lagrangian advection schemes: analysis and application to a multi-level primitive equation model. *Mon. Wea. Rev.*, 110:1831–1842, December 1982.

[2] J. P. Boyd. Equatorial solitary waves. Part I: Rossby solitons. *J. Phys. Oceanogr.*, 10:1699–1717, November 1980.

[3] J. Côté, J.-G. Desmarais, S. Gravel, A. Méthot, A. Patoine, M. Roch, and A. Staniforth. The operational CMC-MRB Global Environment Multiscale (GEM) model. Part II: Results. *Mon. Wea. Rev.*, 126:1397–1416, June 1998.

[4] J. Côté, S. Gravel, A. Méthot, A. Patoine, M. Roch, and A. Staniforth. The operational CMC-MRB Global Environmental Multiscale (GEM) model. Part I: Design considerations and formulation. *Mon. Wea. Rev.*, 126:1373–1395, June 1998.

[5] J. Côté and A. Staniforth. A two-time-level semi-Lagrangian semi-implicit scheme for spectral models. *Mon. Wea. Rev.*, 116:2003–2012, October 1988.

[6] J. Côté and A. Staniforth. An accurate and efficient finite-element global model of the shallow water equations. *Mon. Wea. Rev.*, 118:2707–2717, December 1990.

[7] G. J. Haltiner and R. T. Williams. *Numerical Prediction and Dynamic Meteorology*. John Wiley and Sons, 1980.

[8] M. Hortal. Aspects of the numerics of the ECMWF model. In *Proceedings of a seminar held at ECMWF on recent developments in numerical methods for atmospheric modelings*, pages 127–143, 1999.

[9] M. Iskandarani, D. B. Haidvogel, and J. P. Boyd. A staggered spectral element model with application to the ocean shallow water equations. *Int. J. Numer. Methods Fluids*, 20:393–414, 1995.

[10] A. T. Layton, C. C. Christara, and K. R. Jackson. Optimal quadratic spline collocation methods for the shallow water equations on the sphere. Submitted to *Mathematics and Computers in Simulation*, 2002.

[11] H. Ma. A spectral element basin model for the shallow water equations. *J. Comput. Phys.*, 109:133–149, 1992.

[12] A. McDonald. Accuracy of multiply-upstream, semi-Lagrangian advective schemes. *Mon. Wea. Rev.*, 112:1267–1275, June 1984.

[13] A. Robert. A semi-Lagrangian, semi-implicit numerical integration scheme for the primitive meteorological equations. *Atmos.-Ocean*, 19:35–46, 1981.

[14] A. Staniforth and J. Côté. Semi-Lagrangian integration schemes for atmospheric models – a review. *Mon. Wea. Rev.*, 119:2206–2223, September 1991.

[15] A. Staniforth and H. Mitchell. A semi-implicit finite-element barotropic model. *Mon. Wea. Rev.*, 105:154–169, February 1977.

[16] A. W. Tam. High-order spatial discretization methods for the shallow water equations. http://www.cs.toronto.edu/pub/reports/na/tam-01-phd.ps.Z, Ph.D. Thesis, Department of Computer Science, University of Toronto, Toronto, Ontario, Canada, February 2001.

[17] D. L. Williamson and J. G. Olson. Climate simulations with a semi-Lagrangian version of the NCAR Community Climate Model. *Mon. Wea. Rev.*, 122:1594–1610, July 1994.

[18] D. L. Williamson, J. G. Olson, and B. A. Boville. A comparison of semi-Lagragnain and Eulerian tropical climate simulations. *Mon. Wea. Rev.*, 126:1001–1012, April 1998.