Improved double Fourier series on a sphere and its application to a semi-implicit semi-Lagrangian shallow water model and an Eulerian advection model

Hiromasa Yoshimura

Meteorological Research Institute, Japan Meteorological Agency, 1-1 Nagamine, Tsukuba, Ibaraki 305-0052, Japan

Correspondence to: Hiromasa Yoshimura (hyoshimu@mri-jma.go.jp)

Abstract. The computational cost of a spectral model using spherical harmonics (SH) increases significantly at high resolution because the transform method with SH requires $O(N^3)$ operations, where $N$ is the truncation wavenumber. One way to solve this problem is to use double Fourier series (DFS) instead of SH, which requires $O(N^2 \log N)$ operations. One way to reduce the computational cost of a spectral model using spherical harmonics (SH) is to use double Fourier series (DFS) instead of SH. The transform method using SH usually requires $O(N^3)$ operations, where $N$ is the truncation wavenumber, and the computational cost significantly increases at high resolution. On the other hand, the method using DFS requires only $O(N^2 \log N)$ operations. This paper proposes a new DFS method that improves the numerical stability of the model compared with the conventional DFS methods by adopting the following two improvements: a new expansion method that employs the least-squares method (or the Galerkin method) to calculate the expansion coefficients in order to minimize the error caused by wavenumber truncation, and new basis functions that satisfy the continuity of both scalar and vector variables at the poles. Partial differential equations such as the Poisson equation and the Helmholtz equation are solved by using the Galerkin method. In the semi-implicit semi-Lagrangian shallow water model using the new DFS method, the Williamson test cases 2 and 5 and the Galewsky test case give stable results without the appearance of high-wavenumber noise near the poles, even without using horizontal diffusion and without a zonal Fourier filter. In the Eulerian advection model using the new DFS method, the Williamson test cases 1, which simulates a cosine-bell advection, also gives stable results without horizontal diffusion but with a zonal Fourier filter. The shallow water model using the new DFS method is faster than the SH model, especially at high resolutions, and gives almost the same results, except that small oscillations near the truncation wavenumber in the kinetic energy spectrum appear only in the shallow water model using SH. This small oscillation in the SH model can probably be eliminated by using the vector harmonic transform which is similar to the vector transform using the least-squares method (or the Galerkin method) in the model using the new DFS method. The new DFS model is faster than the SH model, especially at high resolutions, and gives almost the same results.
1 Introduction

Global spectral atmospheric models using the spectral transform method with spherical harmonics (SH) as basis functions are widely used. They are used in the Japan Meteorological Agency (JMA, 2019) and the Meteorological Research Institute (MRI; Yukimoto et al., 2011, 2019) for a range of applications, including operational weather prediction, operational seasonal prediction, and global warming projection. The spectral model has the advantage that the accuracy in horizontal derivatives is good, and the semi-implicit scheme, which improves numerical stability, can be easily applied because the Helmholtz equation and the Poisson equation are easily solved in spectral space. The application of the semi-implicit semi-Lagrangian scheme allows for timesteps longer than the Courant–Friedrichs–Lewy (CFL) condition, which makes the model computationally efficient. In the spectral model using SH, the Legendre transform used in the latitudinal direction significantly increases the computational cost at high resolutions since the Legendre transform usually requires $O(N^3)$ operations and $O(N^3)$ memory usage (unless using the fast Legendre transform or on-the-fly computation of the associated Legendre functions shown below), where $N$ is the truncation wavenumber. One way to reduce the operation count and the memory usage at high resolutions with large $N$ is to use the fast Legendre transform (Suda, 2005; Tygert, 2008; Wedi et al., 2013, Wedi 2014), which requires only $O(N^2 \log N)$ operations and also effectively reduces the memory usage. In the fast Legendre transform, the threshold parameter affecting the accuracy-cost balance is chosen so that a loss of accuracy is sufficiently small, although the accuracy is compromised to reduce the operation count. Dueben et al. (2020) presented global simulations of the atmosphere at 1.45 km grid-spacing in the SH model using the fast Legendre transform. Another approach used to improve the Legendre transform is on-the-fly computation of the associated Legendre functions (Schaeffer, 2013; Ishioka, 2018), which still requires $O(N^3)$ operations but requires only $O(N^2)$ memory usage. This small memory usage also contributes to speeding up calculations by taking advantage of the cache memory.

Alternatively, we can use double Fourier series (DFS) as basis functions to reduce the operation count and the memory usage in the global spectral model. In DFS models (and also in SH models), the scalar variable $F(\lambda, \theta)$ is zonally expanded as

$$F(\lambda, \theta) \approx \sum_{m=-M}^{M} F_m(\theta) e^{i m \lambda}, \quad (1)$$

where $\lambda$ is longitude, $\theta$ is colatitude, and $M$ is the zonal truncation wavenumber. Several methods have been proposed for meridional expansion with DFS: Merilees (1973b), Boer and Steinberg (1975), and Spotz et al. (1998) performed the Fourier transform meridionally along a great circle. Spotz et al. (1998) showed that by using the spherical harmonic filter, the explicit DFS shallow water model using the pseudo-spectral method can produce results comparable with the SH model in terms of accuracy and stability. However, the spherical harmonic filter consists of the forward SH
transform (from grid space to spectral space) followed by the inverse SH transform (from spectral space to grid space), which increases the computational cost.

Orszag (1974) and Boyd (1978) expanded \( F_{\theta}(\theta) \) meridionally as

\[
F_{\theta}(\theta) \approx \begin{cases} 
    f_{\theta}(\theta) & \text{for even } m, \\
    \sin \theta f_{\theta}(\theta) & \text{for odd } m,
\end{cases}
\]

(2a)

\[
f_{\theta}(\theta) \equiv \sum_{n=0}^{N} f_{m,n} \cos n \theta,
\]

(2b)

where \( N \) is the meridional truncation wavenumber. The coefficients \( f_{m,n} \) for odd \( m \) are calculated by the forward Fourier cosine transform of \( F_{\theta}(\theta) / \sin \theta \). Orszag (1974) imposed the following conditions at the poles:

\[
f_{\theta}(0) = 0 \text{ and } f_{\theta}(\pi) = 0 \text{ for } |m| \geq 2,
\]

(3)

which can be expressed in terms of the expansion coefficients \( f_{m,n} \) as

\[
\sum_{n \text{ even}}^{N} f_{m,n} = 0 \text{ and } \sum_{n \text{ odd}}^{N} f_{m,n} = 0 \text{ for } |m| \geq 2.
\]

(4)

Satisfying the above conditions ensures that the scalar variable \( F(\lambda, \theta) \) and its gradient \( \nabla F \) are continuous at the poles. In Orszag (1974), only \( f_{-1,n} \) and \( f_{1,n} \) were modified to satisfy Eq. (4), but this is not the best way to satisfy the same conditions as Eq. (3) or Eq. (4), as will be shown in Sect. 4.3.

Yee (1981) and Layton and Spotz (2003) expanded \( F_{\theta}(\theta) \) as

\[
F_{\theta}(\theta) = \begin{cases} 
    \sum_{n=0}^{N} f_{m,n} \cos n \theta & \text{for even } m, \\
    \sum_{n=1}^{N} f_{m,n} \sin n \theta & \text{for odd } m.
\end{cases}
\]

(5)

In the semi-implicit semi-Lagrangian shallow water model in Layton and Spotz (2003), the spherical harmonic filter was applied to the prognostic variables for stability and accuracy. Layton and Spotz (2003) explained that the expansion with Eq. (5) permits discontinuity at the poles and nonisotropic waves, which may lead to a prohibitive timestep restriction and numerical instability, and these problems can be avoided by applying the spherical harmonic filter.

Cheong (2000a, 2000b) proposed expanding \( F_{\theta}(\theta) \) as

\[
F_{\theta}(\theta) \approx \begin{cases} 
    \sum_{n=0}^{N} f_{m,n} \cos n \theta & \text{for } m = 0, \\
    \sum_{n=1}^{N} f_{m,n} \sin n \theta & \text{for odd } m, \\
    \sum_{n=1}^{N} f_{m,n} \sin \theta \sin n \theta & \text{for even } m \neq 0.
\end{cases}
\]

(6)
The meridional basis functions $\sin \theta \sin n\theta$ for even $m (\neq 0)$ are different from Eq. (5). The coefficients $F_{m,n}$ for even $m (\neq 0)$ are calculated by forward Fourier sine transform of $F_m(\theta) \sin \theta$. The basis functions in Eq. (6) automatically satisfy the same conditions at the poles as Eq. (3) for even $m$, and guarantee the continuity of the scalar variable $F$ at the poles, which is an advantage compared with the basis functions in Eq. (5). However, on the other hand, Eq. (6) does not automatically satisfy the conditions in Eq. (3) for odd $m$, and does not guarantee the continuity of $\nabla F$ at the poles. The shallow water model and the vorticity equation model using a semi-implicit Eulerian scheme ran stably without the spherical harmonic filter by using high-order horizontal diffusion with $O(N^2)$ operations to smooth out the high-wavenumber components (Cheong, 2000b; Cheong et al., 2002; Kwon et al., 2004). The semi-implicit Eulerian hydrostatic atmospheric model also ran stably with high-order horizontal diffusion (Cheong, 2006; Koo and Hong, 2013; Park et al., 2013). However, the computational results of these models appear to be a little different from (slightly worse than) the models using SH. One reason for this seems to be the appearance of high-wavenumber oscillation resulting from the meridional wavenumber truncation with $N = N' \approx 2J/3$ or $J/2$ for even $m (\neq 0)$ (See Sect. 43), and the use of strong high-order horizontal diffusion to smooth out the oscillation, where $J$ is the number of grid points in the latitudinal direction.

Yoshimura and Matsumura (2005) and Yoshimura (2012) stably ran the two-time-level semi-implicit semi-Lagrangian hydrostatic and nonhydrostatic atmospheric models using the DFS basis functions of Cheong in Eq. (6). These models used meridional truncation with $N = J - 1$ and $U = u$ and $V = v$ (instead of $u/\sin \theta$ and $v/\sin \theta$) were transformed from grid space to spectral space, where $u$ is the zonal wind and $v$ is the meridional wind. These models used the same fourth-order horizontal diffusion as the SH models, and did not require the spherical harmonics filter or the strong high-order horizontal diffusion for stability. The numerical stability of the models is improved due to the following reasons:
1. The semi-Lagrangian scheme is used, which avoids the numerical instability due to the nonlinear advection term.
2. The meridional truncation with $N = J - 1$ and $N' = J$ is used, which enables to reconstruct accurately the given grid-data with the expansion coefficients (Cheong et al., 2004) and avoid the error due to the meridional truncation.

$\mathbf{U} = u \sin \theta$ and $\mathbf{V} = v \sin \theta$, instead of $u/\sin \theta$ and $v/\sin \theta$ are transformed from grid space to spectral space, where $u$ is the zonal wind and $v$ is the meridional wind. The results of these models were very similar to those of the SH models. However, we found the following two problems in these models:
1. High wavenumber noise appears near the poles.
2. The meridional truncation needs to be equal to $J$ for even $m (\neq 0)$ because $\mathbf{N}' < J$ (e.g., $N' \mathbf{K} \not\geq 2J/3$), for even $m (\neq 0)$, causes the high-wavenumber oscillation (See Sect. 4) and the numerical instability. To solve these problems, we propose a new DFS method that adopts the following two improvements:
1. A new expansion method to calculate DFS expansion coefficients of scalar and vector variables, which adopts the least-squares method (or the Galerkin method) to minimize the error due to the meridional wavenumber truncation.

2. New DFS basis functions that automatically satisfy the pole conditions in Eq. (3), which guarantee continuity of not only scalar variables but also vector variables at the poles.

We also use the Galerkin method to solve partial differential equations such as the Poisson equation, the Helmholtz equation, and the shallow water equations. Section 2 describes the arrangement of equally spaced latitudinal grid points used in the new DFS method. Section 3 describes the details of the new DFS method using the new DFS expansion method and the new DFS basis functions. Section 4 examines the error due to the wavenumber truncation in the new DFS method, the Orszag’s DFS method of Orszag (1974), and the Cheong’s old DFS method (Cheong, 2000a, 2000b; Yoshimura and Matsumura, 2005), and the new DFS method. Section 5 examines the accuracy of the old and new DFS methods and the SH method for the Laplacian and Helmholtz operators. Section 6 describes how to integrate the semi-implicit semi-Lagrangian shallow water model using the new DFS method. Section 7 presents conclusions and perspectives.

2 Arrangement of equally spaced latitudinal grid points

2.0 Arrangement of equally spaced latitudinal grid points

In DFS models, equally spaced latitudinal grid points are used. We use the following three ways of arranging equally spaced latitudinal grid points in the model using the new DFS method:

\[ j = j^0, \quad \theta_j = \pi (j + 0.5) / j^0, \quad j = 0, \ldots, j^0 - 1, \quad (7a) \]

\[ j = j^0 + 1, \quad \theta_j = \pi j / j^0, \quad j = 0, \ldots, j^0, \quad (7b) \]

\[ j = j^0 - 1, \quad \theta_j = \pi j / j^0, \quad j = 1, \ldots, j^0 - 1, \quad (7c) \]

where \( \theta_j \) is the latitude at each grid point, and \( j^0 \) is the number of latitudinal grid points in Grid \([0]\). When the grid intervals in Grids \([0]\), \([1]\), and \([-1]\) are set equal, the number of grid points \( j_j \) in Grid \([1]\) is \( j^0 + 1 \), and the number of grid points \( j_j \) in Grid \([-1]\) is \( j^0 - 1 \). Figure 1 shows Grids \([0]\), \([1]\), and \([-1]\) when \( j^0 = 4 \) and the grid interval \( \Delta \theta = \pi / 4 \). Grid \([0]\) has been widely used in DFS models (e.g., for example, in Merilees, 1973; Orszag, 1974; Cheong, 2000a, 2000b; and Yoshimura and Matsumura, 2005), and in DFS expansion (e.g., Cheong et al. 2004). Grid \([-1]\) was used, for example, in the SH model using Clenshaw–Curtis quadrature (Hotta and Ujiie, 2013). All of Grids \([0]\), \([1]\), and \([-1]\) were used in the SH expansion (Schwartztrauber and Specz, 2000). We have confirmed that stable integration is possible in the model using the new DFS method with any of Grids \([0]\), \([1]\), and \([-1]\), as shown in Sect. 5 below.
In the new DFS method, the wind vector components $u$ and $v$ (instead of $u \sin \theta$ and $v \sin \theta$ or $u \sin \theta$ and $v \sin \theta$) are transformed from grid space to spectral space and vice versa, as shown in Sects. 3.5 and 3.6 below. This makes it possible to use Grid [1] that has grid points at the poles.

3 Improved double Fourier series on the sphere

In Sect. 3, we describe the new basis functions for scalar and vector variables, and the new method to calculate expansion coefficients which minimizes the error due to wavenumber truncation. We compare the new DFS method with the SH method to see the difference between them. We also describe how to calculate the Laplacian operator, the Poisson equation, the Helmholtz equation, and horizontal diffusion in the new DFS method.

3.1 New basis functions for a scalar variable

We propose the following new DFS basis functions that automatically satisfy the continuity conditions at the poles in Eq. (3). The scalar variable $T(\lambda, \theta)$ is expanded zonally as

$$T(\lambda, \theta) \equiv \sum_{m=0}^{M} T^c_m(\theta) \cos m\lambda + \sum_{m=1}^{M} T^s_m(\theta) \sin m\lambda,$$

where $T^c_m(\theta)$ and $T^s_m(\theta)$ are calculated from $T(\lambda, \theta)$ by the forward Fourier transform as

$$T^c_m(\theta) = \frac{1}{2\pi} \int_0^{2\pi} \cos m\lambda T(\lambda, \theta) d\lambda, \quad a \equiv \begin{cases} 1 & \text{for } m = 0, \\ \frac{1}{2} & \text{for } m \geq 1. \end{cases}$$

$$T^s_m(\theta) = \frac{1}{\pi} \int_0^{2\pi} \sin m\lambda T(\lambda, \theta) d\lambda.$$  

The and the variables $T^c_m(\theta)$ and $T^s_m(\theta)$ are meridionally expanded as

$$T^{c(s)}_m(\theta) \equiv T^{c(s)}_{m,n}(\theta) \equiv \sum_{n=-N_{\max}}^{N_{\max}} T^{c(s)}_{m,n}(\theta)$$

where

$$S_{n,m}(\theta) \equiv \begin{cases} \cos \theta & \text{for } m = 0, \\
\sin \theta \cos n\theta & \text{for } m = 1, \\
\sin \theta \sin n\theta & \text{for even } m \geq 2, \\
\sin^2 \theta \sin n\theta & \text{for odd } m \geq 3. \end{cases}$$

$$N_{\min} = \begin{cases} 0 & \text{for } m = 0, \\
1 & \text{for } m = 1, \\
2 & \text{for even } m \geq 2, \\
3 & \text{for odd } m \geq 3. \end{cases} \quad N_{\max} = \begin{cases} N & \text{for } m = 0, \\
N - 1 & \text{for } m = 1, \\
N - 1 & \text{for even } m \geq 2, \\
N - 2 & \text{for odd } m \geq 3. \end{cases}$$

Here, the superscript c(s) means c or s, and for example $T^{c(s)}_m(\theta)$ means $T^{c(s)}_m(\theta)$ or $T^{c(s)}_m(\theta)$. 

6
In Eq. (87), \(\cos m\lambda\) and \(\sin m\lambda\) are used instead of \(e^{im\lambda}\) as zonal basis functions for convenience in calculating the expansion coefficients using the least-squares method described below in Sects. 3.3 and 3.6. In Eq. (1129), the meridional basis functions \(\sin^6 \theta \sin n \theta\) for odd \(m \geq 3\) are especially different from Cheong's basis functions in Eq. (6). Either \(\sin n \theta\) or \(\sin \theta \cos n \theta\) can be used as the basis functions for \(m = 1\) because it can be shown using Eq. (A2) from Appendix A that \(\sin n \theta\) \((n = 0, \ldots, N - 1)\) are the linear combination of \(\sin n \theta\) \((n = 1, \ldots, N)\), and vice versa. Here we use \(\sin \theta \cos n \theta\) for \(m = 1\) because it can be more easily divided by \(\sin \theta\), which is convenient for calculating \(\nabla T\).

Using Eq. (A2–c), Eq. (108) can be transformed as follows:

\[
P_m^{(n)}(\theta) = \begin{cases} \sum_{n=0}^{N} p_{n,m}^{(n)} \cos n \theta & \text{for even } m, \\ \sum_{n=1}^{N} p_{n,m}^{(n)} \sin n \theta & \text{for odd } m, \end{cases}
\]

where

\[
P_m^{(n)} = n_{n,m}^{(n)}(\theta) \quad (n = 0, \ldots, N) \quad \text{for } m = 0, 
\]

\[
P_m^{(n)} = \frac{n_{n,m}^{(n)} - n_{n+1,m}^{(n)}}{2} \quad (n = 1, \ldots, N) \quad \text{for } m = 1, 
\]

except for \(n_{1,m}^{(n)} = \frac{2n_{1,m}^{(n)} - n_{2,m}^{(n)}}{2} \quad (n = 1), \)

\[
P_m^{(n)} = \frac{-n_{n-1,m}^{(n)} + n_{n+1,m}^{(n)}}{2} \quad (n = 0, \ldots, N) \quad \text{for even } m \geq 2, 
\]

\[
P_m^{(n)} = \frac{-n_{n-2,m}^{(n)} + 2n_{n,m}^{(n)} - n_{n+2,m}^{(n)}}{4} \quad (n = 1, \ldots, N) \quad \text{for odd } m \geq 3, 
\]

except for \(n_{1,m}^{(n)} = \frac{3n_{1,m}^{(n)} - n_{2,m}^{(n)}}{4} \quad (n = 1). \)

The value of \(N_{\max, m}\) in Eq. (12) is determined so that the maximum value of \(n\) for each \(m\) in Eq. (13) becomes \(N\). In Grid [0] and Grid [1] (See Sect. 2), the upper limit of \(N\) is \(f^0 - 1\) for each \(m\). In Grid [−1], the upper limit of \(N\) is \(f^{-1} - 1\) for \(m \geq 2\), but \(f^{-2} - 2 = f - 1\) for \(m = 0\) or 1. This reason is shown in Appendix C.

When calculating the values of \(P_m^{(n)}(\theta)\) in grid space from \(P_m^{(n)}(\theta)\) in spectral space, the coefficients \(n_{n,m}^{(n)}(\theta)\) are calculated from \(P_m^{(n)}(\theta)\) using Eq. (14a) and inverse discrete cosine and sine transforms (See Appendix B) are performed using Eq. (139) (See Appendix B). The calculation of \(P_m^{(n)}(\theta)\) in spectral space from \(P_m^{(n)}(\theta)\) in grid space is described in Sect. 3.3 below.

The truncated variable \(T^{N,M}(\lambda, \theta)\) is defined as
\[ T^{\Omega M}(\lambda, \theta) \equiv \sum_{m=0}^{M} T_{\Omega, m}^{\Lambda}(\theta) \cos m\lambda + \sum_{m=1}^{M} T_{\Omega, m}^{\Lambda}(\theta) \sin m\lambda. \]  

From Eqs. (108) and (9), the values of \( T_{\Omega, m}^{\Lambda}(\theta) \) at the poles are finite for \( m = 0 \), and the values of \( T_{\Omega, m}^{\Lambda}(\theta) \) and \( T_{m}^{\Omega M}(\theta) \) at the poles are zero for \( m \neq 0 \). Therefore \( T^{\Omega M}(\lambda, \theta) \) is continuous at the poles.

### 3.2 Gradient of a scalar variable

The gradient \( \nabla T^{\Omega M} = (\nabla_{\lambda} T^{\Omega M}, \nabla_{\theta} T^{\Omega M}) \) is obtained as follows:

\[ \nabla_{\lambda} T^{\Omega M} \equiv \frac{1}{a \sin \theta} \frac{\partial T^{\Omega M}}{\partial \lambda} = \sum_{m=1}^{M} T_{\Omega, m}^{\Lambda}(\theta) \cos m\lambda + \sum_{m=1}^{M} T_{\Omega, m}^{\Lambda}(\theta) \sin m\lambda, \]  

(1612a)

\[ \nabla_{\theta} T^{\Omega M} \equiv \frac{m}{a \sin \theta} T^{\Omega M}(\theta) = \sum_{m=1}^{M} T_{\Omega, m}^{\Lambda}(\theta) \cos m\lambda + \sum_{m=1}^{M} T_{\Omega, m}^{\Lambda}(\theta) \sin m\lambda, \]  

(1612b)

\[ \nabla_{\phi} T^{\Omega M} \equiv \frac{1}{a} \frac{\partial T^{\Omega M}}{\partial \phi} = -\frac{1}{a} \frac{\partial T^{\Omega M}}{\partial \theta} = \sum_{n=1}^{N-1} T_{\Omega, m}^{\Lambda}(\theta) \cos m\lambda + \sum_{n=1}^{N-1} T_{\Omega, m}^{\Lambda}(\theta) \sin m\lambda, \]  

(1612c)

where \( a \) is the radius of the earth, and \( \phi \) is the latitude. From Eqs. (1612b,c) and (A2), we obtain

\[ T_{\Omega, m}^{\Lambda}(\theta) \equiv \begin{cases} 0 & \text{for } m = 0, \\ \frac{N_{\text{max}, m}}{a} \frac{\partial T_{\Omega, m}^{\Lambda}(\theta)}{\partial \theta} & \text{for } m = 1, \\ \sum_{n=1}^{N-1} T_{\Omega, m}^{\Lambda}(\theta) \cos m\lambda & \text{for even } m \geq 2, \\ \sum_{n=1}^{N-1} T_{\Omega, m}^{\Lambda}(\theta) \sin m\lambda & \text{for odd } m \geq 3, 
\end{cases} \]  

(184)

where

\[ T_{\Omega, m}^{\Lambda} = \frac{1}{a} T_{\Omega, m}^{\Lambda} \sum_{n=0}^{N-1} T_{\Omega, m}^{\Lambda} \cos n\theta, \]  

(195a)

\[ T_{\Omega, m}^{\Lambda} = \frac{1}{a} m T_{\Omega, m}^{\Lambda} \sum_{n=0}^{N-1} T_{\Omega, m}^{\Lambda} \sin n\theta, \]  

(195b)

\[ T_{\Omega, m}^{\Lambda} = \frac{1}{a} \frac{m}{2} (-T_{\Omega, m-1}^{\Lambda} + T_{\Omega, m+1}^{\Lambda}), \]  

(195c)
The equation for $\mathcal{T}_{\phi,m}^\alpha(\theta)$ is the same as Eq. (14), except that the subscript $c$ is replaced with $s$. The equations for $\mathcal{T}_{\phi,m}^\alpha, \mathcal{T}_{\phi,m}^\beta, \mathcal{T}_{\phi,m}^\gamma(\theta)$ are the same as Eq. (195), except that $\mathcal{T}_{\phi,m}^\alpha, \mathcal{T}_{\phi,m}^\beta, \mathcal{T}_{\phi,m}^\gamma(\theta)$ are replaced with $\mathcal{T}_{\phi,m}^\alpha, \mathcal{T}_{\phi,m}^\beta, \mathcal{T}_{\phi,m}^\gamma(\theta)$, respectively. From Eqs. (173b), (139) and (140), we obtain

\[
\mathcal{T}_{\phi,m}^\alpha(\theta) = \begin{cases} 
\sum_{n=1}^{N} \mathcal{T}_{\phi,n,m}^\alpha \sin n\theta & \text{for } m = 0, \\
\sum_{n=1}^{N} \mathcal{T}_{\phi,n,m}^\alpha \cos n\theta & \text{for } m = 1, \\
\sum_{n=1}^{N} \mathcal{T}_{\phi,n,m}^\alpha \sin n\theta & \text{for } m \geq 2,
\end{cases}
\]

for even $m \geq 2$, and

\[
\mathcal{T}_{\phi,m}^\alpha(\theta) = \sum_{n=1}^{N-1} \mathcal{T}_{\phi,n,m}^\alpha \sin n\theta
\]

for odd $m \geq 3$, where

\[
\mathcal{T}_{\phi,m}^\alpha = -\frac{1}{a} \left(-n\mathcal{T}_{\phi,m,n}^\alpha \right) \quad (n = 1, \ldots, N) \quad \text{for } m = 0,
\]

\[
\mathcal{T}_{\phi,m}^\alpha = -\frac{1}{a} \left[\frac{n(\mathcal{T}_{\phi,n,m}^\alpha - \mathcal{T}_{\phi,n+1,m}^\alpha)}{2}\right] \quad (n = 0, \ldots, N) \quad \text{for } m = 1,
\]

except for $\mathcal{T}_{\phi,1,m=1}^\alpha = -\frac{1}{a} \left[\frac{n(\mathcal{T}_{\phi,n-1,m}^\alpha - \mathcal{T}_{\phi,n,m}^\alpha)}{2}\right]$, where $n = 1$,

\[
\mathcal{T}_{\phi,m}^\alpha = -\frac{1}{a} \left[\frac{n(\mathcal{T}_{\phi,n-1,m}^\alpha + 2\mathcal{T}_{\phi,n,m}^\alpha - \mathcal{T}_{\phi,n+1,m}^\alpha)}{4}\right] \quad (n = 0, \ldots, N) \quad \text{for } m = 1.
\]

15

The equations for $\mathcal{T}_{\phi,m=1}^\alpha, \mathcal{T}_{\phi,m=1}^\beta, \mathcal{T}_{\phi,m=1}^\gamma(\theta)$ at the poles are finite for $m = 1$ and zero for $m \neq 1$, and moreover the following relations are satisfied for $m = 1$:

\[
\mathcal{T}_{\phi,m=1}^\alpha(\theta) = -\mathcal{T}_{\phi,m=1}^\beta(\theta) = -\frac{1}{a} \sum_{n=1}^{N-1} \mathcal{T}_{\phi,n,m=1}^\alpha \quad \text{at } \theta = 0 \text{ (North Pole)},
\]

\[
\mathcal{T}_{\phi,m=1}^\gamma(\theta) = \mathcal{T}_{\phi,m=1}^\beta(\theta) = -\frac{1}{a} \sum_{n=1}^{N-1} \mathcal{T}_{\phi,n,m=1}^\alpha \quad \text{at } \theta = 0 \text{ (North Pole)}.
\]
Thus, it is guaranteed that \( \nabla T_\theta \equiv \begin{pmatrix} T_{\theta,0} & T_{\theta,1} \end{pmatrix} \) is continuous at the poles.

### 3.2.3 New method to calculate expansion coefficients for a scalar variable

One way to calculate the coefficients \( T_{\theta,m}^{(a)} \) from \( T_{\theta,m}^{(a)}(\theta) \) in Eq. (108) is to perform a forward cosine transform of \( T_{\theta,m}^{(a)}(\theta) \cos \theta \) for \( \theta = \pi \) and a sine transform of \( T_{\theta,m}^{(a)}(\theta) \sin \theta \) for odd \( \theta \). However, this approach with the meridional wavenumber truncation \( N < J \) leads to the large high-wavenumber oscillation as in Cheong's basis functions for even \( m \geq 2 \) (See Sect. 4.3). Dividing \( T_{\theta,m}^{(a)}(\theta) \) by \( \sin \theta \) reduces the numerical stability of the model more significantly than dividing \( T_{\theta,m}^{(a)}(\theta) \) by \( \sin \theta \).

Here we propose a new method to calculate expansion coefficients using the least-squares method to minimize the error due to the meridional wavenumber truncation. This method also avoids dividing \( T_{\theta,m}^{(a)}(\theta) \) by \( \sin \theta \) or \( \sin^2 \theta \) before the forward cosine or sine transforms. The coefficients \( T_{\theta,m}^{(a)} \) and \( T_m^{(s)} \) in Eq. (108) are calculated as follows. First, \( T_{\theta,m}^{(a)}(\theta) \) and \( T_m^{(s)}(\theta) \) in Eq. (108) are expanded like Eq. (5) as

\[
T_{\theta,m}^{(a)}(\theta) \equiv \frac{1}{n} \sum_{n=0}^{N} T_{\theta,n}^{(a)} \theta \cos n \theta \quad \text{for even } m,
\]

\[
T_m^{(s)}(\theta) \equiv \frac{1}{n} \sum_{n=1}^{N} T_{m,n}^{(s)} \theta \sin n \theta \quad \text{for odd } m,
\]

where \( \theta \) is the number of meridional grid points, and the expansion coefficients \( T_{\theta,n}^{(a)} \) and \( T_{m,n}^{(s)} \) are calculated by the forward discrete cosine transform for even \( m \) and the forward discrete sine transform for odd \( m \) from the values of \( T_{\theta,m}^{(a)}(\theta) \) at the grid points (See Sect. 2.10 Appendix B). The equation for \( E \) is the same as Eq. (19), except that the subscript \( c \) is replaced with \( s \).

Next, \( T_{\theta,m}^{(a)} \) and \( T_m^{(s)} \) are calculated using the least-squares method to minimize the following error \( E \) (the squared \( L_2 \) norm of the residual):

\[
E \equiv \frac{1}{2\pi} \int_0^{2\pi} \int_0^\pi R(\lambda, \theta)^2 d\theta d\lambda,
\]

where the residual \( R(\lambda, \theta) \) is

\[
R(\lambda, \theta) \equiv \sum_{m=0}^{N} \sum_{n=0}^{M} T_{\theta,n}^{(a)}(\theta) \cos m \lambda + \sum_{m=1}^{M} T_{m,n}^{(s)}(\theta) \sin m \lambda - T(\lambda, \theta).
\]
From Eqs. (20), (21), and (A3), we derive

\[
E = \frac{1}{2} \int_0^{2\pi} \left[ \tau_m^{\phi}(\theta) - \tau_m^{\phi}(\phi) \right]^2 + \frac{1}{2} \sum_{n=1}^{\infty} \left( \tau_n^{\phi}(\theta) - \tau_n^{\phi}(\phi) \right)^2 + \frac{1}{2} \sum_{n=1}^{\infty} \left( \tau_n^{\phi}(\theta) - \tau_n^{\phi}(\phi) \right)^2 \, d\phi \tag{22}
\]

Eqs. 19 and (10) are substituted into Eq. (22). The equations \( \partial E / \partial \theta \) and \( \partial E / \partial \phi \) are used to calculate \( \theta \) and \( \phi \), respectively, so that \( E \) is minimized. From Eqs. (240) and (251), the equations \( \partial E / \partial \theta \) and \( \partial E / \partial \phi \) are used in the least-squares method to minimize \( E \) used in the least-squares method, we obtain

\[
\frac{1}{2\pi^2} \int_0^{2\pi} \frac{\partial \tau_m^{\phi}(\theta)}{\partial T_m^{\phi}} \cos m\lambda \, R(\lambda, \theta) \, d\lambda d\lambda = 0, \tag{26a}
\]

\[
\frac{1}{2\pi^2} \int_0^{2\pi} \frac{\partial \tau_m^{\phi}(\theta)}{\partial T_m^{\phi}} \sin m\lambda \, R(\lambda, \theta) \, d\lambda d\lambda = 0. \tag{26b}
\]

From Eq. (108), we derive

\[
\frac{\partial T_m^{\phi}(\theta)}{\partial T_m^{\phi}} = \frac{\partial T_m^{\phi}(\theta)}{\partial T_m^{\phi}} = S_{m,0}(\theta), \tag{27}
\]

where the functions of are the new DFS meridional basis functions defined as

Equations (26a) and (27) shows that the residual \( R(\lambda, \theta) \) is orthogonal to each of the new DFS basis functions \( S_{m,0}(\theta) \) and \( S_{m,0}(\theta) \sin m\lambda \), which means that Eq. (26a) is the same as the equation derived using the Galerkin method. Thus, the equations \( \partial E / \partial \theta \) and \( \partial E / \partial \phi \) used in the least-squares method described in Sect. 2.3 are the same as those derived with the Galerkin method.

From Eqs. (26a), (27), (28), (29), and (A3), we derive

\[
\int_0^{\pi} S_{m,0}(\theta) T_m^{\phi}(\theta) \, d\theta = \int_0^{\pi} S_{m,0}(\theta) T_m^{\phi}(\theta) \, d\theta. \tag{28}
\]

From Eq. (28) and (DM) in Appendix D, we obtain

\[
\int_0^{\pi} S_{m,0}(\theta) T_m^{\phi}(\theta) \, d\theta = \int_0^{\pi} S_{m,0}(\theta) T_m^{\phi}(\theta) \, d\theta. \tag{29}
\]

From Eq. (22) and (29), the following equations for \( T_m^{\phi} \) and \( T_m^{\phi} \) are derived, as shown in Appendix E.

From Eq. (9) \( \theta = \phi \) and Eq. (A5), we derive

\[
\frac{\partial}{\partial T_m^{\theta}} \left[ \sum_{n=1}^{\infty} \left( \tau_n^{\phi}(\theta) - \tau_n^{\phi}(\phi) \right)^2 \right] = 0 \text{ for even } m, \tag{23a}
\]

\[
\frac{\partial}{\partial T_m^{\phi}} \left[ \sum_{n=1}^{\infty} \left( \tau_n^{\phi}(\theta) - \tau_n^{\phi}(\phi) \right)^2 \right] = 0 \text{ for odd } m. \tag{23b}
\]

From Eq. (23) and (10), we derive the following equations for \( T_m^{\phi} \).
For $m = 0$,
\[ T_{m}^{(x)} = T_{m}^{(x)} \quad (0 \leq n \leq N). \tag{3024a} \]

For $m = 1$,
\[ -2T_{0,n}^{(x)} + 2T_{n+2}^{(x)} - T_{n+1}^{(x)} = -2T_{0,n-3}^{(x)} + 2T_{n+2}^{(x)} \quad (0 \leq n \leq N - 1), \tag{3024b} \]
5

with the exception of the following underlined values:
\[ \begin{bmatrix}
T_{1,1}^{(x)} - T_{1,0}^{(x)} = 2T_{2,1}^{(x)} \\
-2T_{0,1}^{(x)} + 2T_{2,1}^{(x)} - T_{1,0}^{(x)} = -2T_{1,2}^{(x)} + 2T_{2,1}^{(x)}
\end{bmatrix} \tag{n = 1, n = 2} \]

For even $m (\geq 2)$,
\[ -T_{n-2}^{(x)} + 2T_{n}^{(x)} - T_{n+2}^{(x)} = 2T_{n+1}^{(x)} - 2T_{n+1}^{(x)} \quad (1 \leq n \leq N - 1), \tag{3024c} \]
10

with the exception of the following underlined values:
\[ \begin{bmatrix}
3T_{1,1}^{(x)} - T_{1,0}^{(x)} = 4T_{1,2}^{(x)} \\
-3T_{0,1}^{(x)} + 2T_{2,1}^{(x)} - T_{1,0}^{(x)} = -3T_{2,2}^{(x)} + 2T_{1,2}^{(x)}
\end{bmatrix} \tag{n = 1} \]

For odd $m (\geq 3)$,
\[ T_{n-2}^{(x)} - 4T_{n-1}^{(x)} + 6T_{n}^{(x)} - 4T_{n+1}^{(x)} + T_{n+2}^{(x)} = -4T_{n-2}^{(x)} + 8T_{n}^{(x)} - 6T_{n+2}^{(x)} \quad (1 \leq n \leq N - 2), \tag{3024d} \]
15

with the exception of the following underlined values:
\[ \begin{bmatrix}
10T_{1,1}^{(x)} - 5T_{1,0}^{(x)} + T_{1,2}^{(x)} = 12T_{1,2}^{(x)} - 4T_{2,1}^{(x)} \\
-5T_{0,1}^{(x)} + 2T_{2,1}^{(x)} + T_{1,0}^{(x)} = -5T_{2,2}^{(x)} + 2T_{1,2}^{(x)}
\end{bmatrix} \tag{n = 1, n = 2} \]

\[ -5T_{1,1}^{(x)} + 6T_{1,2}^{(x)} - 4T_{2,1}^{(x)} + T_{1,2}^{(x)} = -5T_{1,1}^{(x)} + 6T_{1,2}^{(x)} - 4T_{2,1}^{(x)} \quad (n = 3). \tag{3024e} \]

From Eq. (30a), $T_{m}^{(x)}$ for $m = 0$ is obtained. From Eqs. (3024d), the two following linear simultaneous equations for $m \geq 3$.

\[ D_{\nu,\tau}^{(x)} T_{\nu,\tau}^{(x)} = D_{\nu,\tau}^{(x)} T_{\nu,\tau}^{(x)} \quad (\nu = 1 \to N, \tau = 1 \to N), \tag{31} \]
20

\[ \begin{bmatrix}
T_{1,1}^{(x)} & T_{1,2}^{(x)} & T_{1,3}^{(x)} & T_{1,4}^{(x)} & \cdots & T_{1,N}^{(x)} \\
T_{2,1}^{(x)} & T_{2,2}^{(x)} & T_{2,3}^{(x)} & T_{2,4}^{(x)} & \cdots & T_{2,N}^{(x)} \\
T_{3,1}^{(x)} & T_{3,2}^{(x)} & T_{3,3}^{(x)} & T_{3,4}^{(x)} & \cdots & T_{3,N}^{(x)} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
T_{N,1}^{(x)} & T_{N,2}^{(x)} & T_{N,3}^{(x)} & T_{N,4}^{(x)} & \cdots & T_{N,N}^{(x)}
\end{bmatrix} \]
where the matrices $C_0$ and $C_2$ are penta-diagonal. From Eqs. (30b,c), the equations similar to Eq. (31) for $m = 1$ and even $m \geq 2$ with tri-diagonal matrices $C_0$ and $C_2$ are derived. By using Eq. (31), the expansion coefficients $T_{2n}$ are calculated from $T_{2n+1}$. A penta-diagonal matrix $C$ can be LU decomposed as

$$C = LU,$$

with $L$ and $U$ defined as

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ * & 1 & 0 & 0 & \cdots & 0 \\ * & * & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ * & * & * & \cdots & 1 & 0 \\ * & * & * & \cdots & 0 & 1 \end{bmatrix},$$

$$(26)$$

$$U = \begin{bmatrix} 1 & * & * & \cdots & 0 \\ 0 & 1 & * & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix},$$

$$(32)$$

To solve $LUx = b$, we solve $Ly = b$ with forward substitution first, and then solve $Ux = y$ with backward substitution.

There are also other methods to solve Eq. (1225). For example, the method using LU decomposition considering penta-diagonal matrices as $2 \times 2$ block tri-diagonal matrices makes SIMD operations more effective. The method using cyclic reduction for block tri-diagonal matrices (e.g., Gander and Golub, 1997) is suitable for vectorization and parallelization. The calculation with these methods for each $m$ requires $O(N)$ operations. The simultaneous equations with tri-diagonal matrices $C$ derived from Eqs. (24b,c) can be solved in a similar way to Eq. (25). Therefore, the calculation of $T_{2m}$ for all $m$ and $n$ with Eq. (3024) requires only $O(N^2)$ operations. The equations for are derived from $\xi = 0$; these are the same equations as Eqs. (24b,c,d), except that the subscript $c$ is replaced with $s$.

### 2.4 Relation between the least-squares method and Galerkin method for a scalar variable

Here we discuss the relation between the least-squares method described above and the Galerkin method when calculating the expansion coefficients of a scalar variable.

From Eqs. (20) and (21) and the equations $\partial E/\partial T_0 = 0$ and $\partial E/\partial T_1 = 0$ used in the least-squares method, we obtain

$$\int_0^{2\pi} \int_0^\pi \sin m \lambda R(\lambda, \theta) \theta d\theta d\lambda = 0,$$

$$20$$

$$(27a)$$

$$\int_0^{2\pi} \int_0^\pi \cos m \lambda R(\lambda, \theta) \theta d\theta d\lambda = 0,$$

$$20$$

$$(27b)$$

From Eq. (23), we obtain
\[ \frac{\partial}{\partial \theta} \left( r \frac{\partial T}{\partial \theta} \right) = \frac{\partial^2 T}{\partial r^2} + \frac{\partial^2 T}{\partial \phi^2} \]

where the functions of \( S_m^\theta (\phi) \) are the new DFS meridional basis functions defined as

\[ S_m^\theta (\phi) \equiv \begin{cases} \cos n\phi & \text{for } m = 0, \\ \sin \theta \cos n\phi & \text{for } m = 1, \\ \sin \theta \sin n\phi & \text{for even } m \geq 2, \\ \sin^2 \theta \sin n\phi & \text{for odd } m \geq 3. \end{cases} \]

Equation (27) shows that the residual \( R(\lambda, \theta) \) is orthogonal to each of the new DFS basis functions \( S_m^\theta (\phi) \cos m\lambda \) and \( S_m^\theta (\phi) \sin m\lambda \), which means that Eq. (27) is the same as the equation derived using the Galerkin method. Thus, the equations \( \frac{\partial E}{\partial \lambda} = 0 \) and \( \frac{\partial E}{\partial \lambda} = 0 \) used in the least-squares method described in Sect. 2.3 are the same as those derived with the Galerkin method.

### 32.45 Comparison of new DFS with SH

Here we compare the new DFS method with the SH method to see the difference between them. In the SH method, \( T_m^\phi (\theta) \) and \( T_m^\theta (\theta) \) in Eq. (87) are expanded with the associated Legendre functions \( P_{m^\phi}^n (\phi) \) as

\[ T_m^\phi (\theta) = \sum_{\ell=0}^{m^\phi} a_{m,\ell} \sin \ell \theta \cos \ell \theta. \]

However, the functions \( \sin n^\theta \cos (n - m) \theta \) for \( n \geq |m| \) can be expressed as the linear combination of \( P_{m^\phi}^{|m|} (\theta) \) for \( m = |m|, \ldots, n \). Substituting Eq. (352) into Eq. (350) gives the following equations.

\[ T_m^{\phi,n} (\theta) = \sum_{n=0}^{N} \frac{n+1}{2} a_{m,n} \sin (m^\theta \theta) \cos \theta, \]

where \( m \geq 0 \). Equation (354) is similar to Eq. (108) in the following sense: the basis functions for \( m = 0 \) and \( m = 1 \) in Eq. (354) are the same as in Eq. (118). The basis functions \( \sin^2 \theta \cos n\theta \) for \( m = 2 \) and \( \sin^3 \theta \cos n\theta \) for \( n = 0, \ldots, N-3 \) for \( m = 3 \) in Eq. (354) are the linear combinations of \( \sin \theta \sin n\theta \) for \( n = 1, \ldots, N-1 \) and
\[
\sin^2 \theta \sin n \theta \quad (n = 1, \ldots, N - 2) \text{ in Eq. (114), respectively (see Eq. (A2a)), and vice versa. The basis functions for } m \geq 4 \text{ in Eq. (362) are different from those in Eq. (118). The number of expansion coefficients in Eq. (332) or Eq. (362) in the SH method is smaller than in Eq. (118) in the new DFS method for each } m \geq 4. \text{ From Eqs. (29) and (360), the number of expansion coefficients } \tau_{n,m}^{\text{SH}} \text{ in the SH model is about } N^2/2 \text{ when } M = N. \text{ This triangular truncation used in the SH method gives a uniform resolution over the sphere. From Eqs. (87) and (108), the number of expansion coefficients } T_{\varphi, \theta} \text{ in the SH model is about } N^2/2 \text{ when } M = N. \text{ This rectangular truncation used in the DFS model gives almost the same resolution as the grid spacing of the regular longitude–latitude grids. Therefore, the zonal Fourier filter (see Appendix F Sect. 2.11) is used in the DFS model to give a more uniform resolution.}
\]

We compare the method used to calculate the expansion coefficients in the new DFS method with that in the SH method.

The SH expansion coefficients \( \tau_{n,m}^{(c)\text{SH}} \) in Eq. (33) are calculated from \( T_m^{(c)}(\theta) \) by the forward Legendre transform as
\[
\tau_{n,m}^{(c)\text{SH}} = \int_0^\pi P_{m,n}(\theta) T_m^{(c)}(\theta) \sin \theta \, d\theta, \tag{37}
\]
where Gaussian quadrature or Clenshaw–Curtis quadrature (e.g., Hotta and Ujiie, 2018) is usually used for integration. They can also be calculated using the same equations as Eq. (37) except that \( T_m^{(c)\text{SH}}(\theta) \) are replaced with \( T_m^{(c)\text{SH}}(\theta) \) (e.g., Sneeuw and Bun, 1996), although the values of \( \tau_{n,m}^{(c)\text{SH}} \) calculated from \( T_m^{(c)\text{SH}}(\theta) \) are different from those calculated from \( T_m^{(c)}(\theta) \).

In the new DFS method, the values of \( \tau_{n,m}^{(c)\text{SH}} \) calculated from \( T_m^{(c)\text{DFS}}(\theta) \) in Eq. (29) are the same as those calculated from \( T_m^{(c)\text{DFS}}(\theta) \) in Eq. (28) (see Eq. (D4) in Appendix D).

Eq. (37) can be derived using
\[
\tau_{n,m}^{(c)\text{SH}} = \int_0^\pi P_{m,n}(\theta) R_{n,m}(\theta, \lambda) \sin \theta \, d\theta, \tag{38a}
\]

where \( R_{n,m}(\theta, \lambda) \) is the latitudinal weight. The coefficients \( R_{n,m}(\theta, \lambda) \) can also be calculated with the least-squares method that minimizes the error \( E_{\text{SH}} \) (the squared \( L_2 \) norm of the residual):
\[
E_{\text{SH}} = \frac{1}{4\pi} \int_0^{2\pi} \int_0^\pi \left( R_{\text{SH}}(\lambda, \theta) \right)^2 \sin \theta \, d\theta \, d\lambda, \tag{38b}
\]
where the residual \( R_{\text{SH}}(\lambda, \theta) \) is
\[
R_{\text{SH}}(\lambda, \theta) = \left( \sum_{n=1}^M \tau_{n,m}^{(c)\text{SH}}(\theta) \cos m \lambda + \sum_{n=1}^M \tau_{n,m}^{(c)\text{SH}}(\theta) \sin m \lambda \right) - T(\lambda, \theta). \tag{39a}
\]

From Eqs. (38a) and (39a), and the equations \( \partial E_{\text{SH}}/\partial \tau_{n,m}^{(c)\text{SH}} = 0 \) and \( \partial E_{\text{SH}}/\partial T_{m,n}^{(c)\text{SH}} = 0 \) used in the least-squares method to minimize \( E_{\text{SH}} \), we derive
\[
\int_0^{2\pi} \int_0^\pi P_{m,n}(\theta) \cos m \lambda \, R_{\text{SH}}(\lambda, \theta) \sin \theta \, d\theta \, d\lambda = 0, \tag{40a}
\]
\[
\int_0^\pi \int_0^{2\pi} P_{n,m}(\theta) \sin m\lambda R^{\text{SH}}(\lambda, \theta) \sin \theta \, d\lambda d\theta = 0.
\tag{40b}
\]

Eqs. (40) is the same as the equation obtained using the Galerkin method. From Eqs. (40), (33), (34), (9) and (A3), we derive Eq. (37).

The SH expansion coefficients \( A_{m,n} \) and \( B_{m,n} \) in Eq. (30) are usually calculated from the grid-point values of \( T^{\text{SH}}(\theta) \) and \( S^{\text{SH}}(\theta) \), respectively, by using Gaussian quadrature or Clenshaw–Curtis quadrature (e.g., Hotta and Ujiie, 2018). They can also be calculated from \( T^{\text{DFS}}(\theta) \) and \( S^{\text{DFS}}(\theta) \) instead of \( T^{\text{SH}}(\theta) \) and \( S^{\text{SH}}(\theta) \) at the grid points as follows (e.g., Sneeuw and Bun, 1996).

From Eqs. (37), (30), (31), and the equations \( \sigma = 0 \) and \( \sigma = 0 \) used in the least-squares method, we can derive the same equations as Eq. (34). In Eqs. (37) and (38), and Eq. (34), the latitudinal weight \( \sin \theta \) appears, unlike in Eqs. (24) and (29), and Eq. (27)), which is another difference between the SH and the new DFS methods. In the DFS method, the constant latitudinal weight is used in Eq. (24), although the latitudinal area weight described below in Appendix G is usually used as the latitudinal weight at the grid points, for example, for the calculation of the global mean.

When calculating the coefficients \( T^{\text{SH}}_{m,n}(\theta) \) and \( S^{\text{SH}}_{m,n}(\theta) \) in Eq. (40b), we can also consider the least-squares method, not using \( E \) in Eq. (24b) but using \( E' \) with latitudinal weight \( \sin \theta \) like Eq. (32). However, minimizing \( E' \) derives the simultaneous equations for calculating \( T^{\text{DFS}}_{m,n} \) with dense matrices, which leads to \( O(N^3) \) operations. When using \( E' \), the simultaneous equations with penta-diagonal or tri-diagonal matrices require only \( O(N^2) \) operations. Therefore, we choose to use \( E' \) instead of \( E' \).

The new DFS meridional basis functions \( S^{\text{DFS}}_{m,n}(\theta) \) for each \( m \) are not orthogonal but independent. Therefore, by using Gram-Schmidt orthogonalization, the basis functions can be converted to orthogonalized basis functions \( S^O_{n,m}(\theta) \), which satisfy

\[
\frac{1}{\pi} \int_0^{\pi} S^O_{n,m}(\theta) S^O_{n',m}(\theta) \, d\theta = \begin{cases} 1 & \text{for } n = n', \\ 0 & \text{for } n \neq n'. \end{cases}
\tag{41}
\]

This is similar to Eq. (34), but the latitudinal weight is constant. \( T^{\text{DFS}}_{m,n}(\theta) \) in Eq. (8) are expanded with \( S^O_{n,m}(\theta) \) as

\[
T^{\text{DFS}}_{m,n}(\theta) \equiv T^{\text{DFS}}_{m,n}(\theta) \equiv \sum_{n,m=0}^{N_{\text{max}}} T^{\text{DFS}}_{m,n} S^O_{n,m}(\theta)
\tag{42}
\]

By using the least-squares method or the Galerkin method with Eq. (42), we obtain the same equations as Eqs. (24)–(29) except that \( T^{\text{DFS}}_{m,n}(\theta) \) and \( S^{\text{DFS}}_{m,n}(\theta) \) are replaced with \( T^{\text{DFS}}_{m,n}(\theta) \) and \( S^{O}_{m,n}(\theta) \) respectively. From Eq. (29) with \( T^{\text{DFS}}_{m,n}(\theta) \) and \( S^{\text{DFS}}_{m,n}(\theta) \) replaced by \( T^{\text{DFS}}_{m,n}(\theta) \) and \( S^{O}_{m,n}(\theta) \), and Eqs. (41) and (42), we derive

\[
T^{\text{DFS}}_{m,n}(\theta) = \frac{1}{\pi} \int_0^{\pi} S^O_{n,m}(\theta) f^{\text{DFS}}_{m,n}(\theta) \, d\theta.
\tag{43}
\]
Thus, $T_{20}$ and $T_{10}$ in Eqs. (43) and (42) are calculated uniquely. This unique solution $T_{m,n}^{(1)}(\theta)$ is the same as $T_{m,n}^{(2)}(\theta)$ in Eq. (29) obtained by the least-squares method with the non-orthogonal basis functions $S_{m,n}(\theta)$, because $S_{m,n}(\theta)$ ($n = N_{\max,\nu} \ldots N_{\max,v}$) are the linear combination of $S_{n,m}(\theta)$ ($n = N_{\max,\nu} \ldots N_{\max,v}$), for each $m$, and vice versa.

3.56 Application of the New basis functions to a wind vector

The velocity potential $\chi$ and the stream function $\psi$ can be converted into the wind vector components $u$ and $v$ using the equations

$$u = \frac{1}{a \cos \phi} \frac{\partial \chi}{\partial \lambda} - \frac{1}{a \sin \theta} \frac{\partial \psi}{\partial \lambda}$$

$$v = \frac{1}{a \cos \phi} \frac{\partial \psi}{\partial \lambda} + \frac{1}{a \sin \theta} \frac{\partial \chi}{\partial \lambda}$$

where $u = a \cos \phi \frac{d \chi}{d t}$ is the zonal wind, and $v = a \frac{d \psi}{d t}$ is the meridional wind. The scalar variables $\chi$ and $\psi$ are expanded like Eqs. (87) and (108) as

$$\chi \approx \chi_0 + \sum \frac{\chi_{m,1}(\theta)}{m} \cos m\lambda$$

$$\psi \approx \psi_0 + \sum \frac{\psi_{m,1}(\theta)}{m} \sin m\lambda$$

The equations for $m \geq 1$ are the same as Eqs. (41) and (42), respectively, except that the subscript $c$ is replaced with $s$. Here, the truncated variables $\psi_{n,m}(\lambda, \theta)$ and $\chi_{n,m}(\lambda, \theta)$ are defined as

$$\psi_{n,m}(\lambda, \theta) \equiv \sum_{n=1}^{N_{\max,\nu}} \frac{\psi_{n,m}(\theta)}{m} S_{m,n}(\theta)$$

$$\chi_{n,m}(\lambda, \theta) = \sum_{n=1}^{N_{\max,\nu}} \frac{\chi_{n,m}(\theta)}{m} S_{m,n}(\theta)$$

From Eqs. (44)–(47), the equations for the wind vector components $u_{n,m}(\lambda, \theta)$ and $v_{n,m}(\lambda, \theta)$ are derived as

$$u_{n,m}(\lambda, \theta) \equiv \frac{1}{a \sin \theta} \frac{\partial \chi_{n,m}(\lambda, \theta)}{\partial \lambda} - \frac{1}{a \cos \phi} \frac{\partial \psi_{n,m}(\lambda, \theta)}{\partial \theta} = \sum_{m=1}^{M} u_{n,m}(\theta) \cos m\lambda + \sum_{m=1}^{M} v_{n,m}(\theta) \sin m\lambda$$
The vector \( (u_{\theta}, v_{\theta}, v_{\phi}) \) in Eqs. (48) and (49) can also be represented as

\[
\begin{align*}
u_{m}^{N}(\theta) & \equiv \frac{m\psi_{m}^{N}(\theta)}{a} \sin \theta + \frac{1}{a} \frac{\partial \psi_{m}^{N}(\theta)}{\partial \theta}, \\
u_{m}^{S}(\theta) & \equiv \frac{m\psi_{m}^{S}(\theta)}{a} \sin \theta + \frac{1}{a} \frac{\partial \psi_{m}^{S}(\theta)}{\partial \theta}
\end{align*}
\]  

where we define the new DFS vector basis functions

\[
\begin{align*}
V_{m}^{1}(\lambda, \theta) & \equiv \frac{mS_{m}(\theta)}{a \sin \theta} \sin m\lambda, \\
V_{m}^{2}(\lambda, \theta) & \equiv \frac{mS_{m}(\theta)}{a \sin \theta} \cos m\lambda, \\
V_{m}^{3}(\lambda, \theta) & \equiv \frac{\partial S_{m}(\theta)}{a \sin \theta} \sin m\lambda, \\
V_{m}^{4}(\lambda, \theta) & \equiv \frac{\partial S_{m}(\theta)}{a \sin \theta} \cos m\lambda
\end{align*}
\]  

From Eqs. (48b,c), (49b,c), and (18)-(21), (41), and (42), we obtain
where

\[ u_{m,n}^e = -\frac{n}{a} \Phi_{m,n} \quad (n = 1, ..., N) \quad \text{for } m = 0, \quad (53a) \]

\[ u_{m,n}^s = \frac{1}{a} \left[ m \chi_{m,n}^s + \frac{n(\psi_{n-1,m}^s - \psi_{n+1,m}^s)}{2} \right] \quad (n = 0, ..., N) \quad \text{for } m = 1, \quad (53b) \]

\[ u_{m,n}^s = \frac{1}{a} \left[ m \chi_{m,n}^s + \frac{n(\psi_{n-1,m}^s - \psi_{n+1,m}^s)}{2} \right] \quad (n = 1, ..., N) \quad \text{for even } m \geq 2, \quad (53c) \]

\[ u_{m,n}^s = \frac{1}{a} \left[ m \chi_{m,n}^s + \frac{4(\psi_{n-1,m}^s - \psi_{n+1,m}^s)}{4} \right] \quad (n = 0, ..., N) \quad \text{for odd } m \geq 3, \quad (53d) \]

except for \( u_{1,n}^s = \frac{1}{a} \left[ m \chi_{1,n}^s + \frac{4(\psi_{n-1,1}^s - \psi_{n+1,1}^s)}{4} \right] \) at \( \theta = 0 \) (North Pole) \quad (54a).
Thus, it is guaranteed that the wind vector $(u, v)$ in Eqs. (545) and (546) is continuous at the poles.

5.2.2 New method to calculate expansion coefficients for a wind vector

We propose a new method that calculates the expansion coefficients $x^k, y^k, f^k, g^k$ in Eqs. (48) and (49) using the least-squares method to minimize the error of $u^M, v^M$ from with respect to $u(\lambda, \theta)$ and $v(\lambda, \theta)$ due to the meridional wavenumber truncation. First, the wind vector components $u$ and $v$ are expanded zonally as

\[
\begin{align*}
  u &\approx \sum_{m=-1}^{N} \sum_{n=1}^{N} C^u_{m,n} \cos n\lambda \\
  v &\approx \sum_{m=-1}^{N} \sum_{n=1}^{N} C^v_{m,n} \sin n\lambda,
\end{align*}
\]

where $C^u_{m,n}$ and $C^v_{m,n}$ are calculated from $u(\lambda, \theta)$ and $v(\lambda, \theta)$ by the forward Fourier transform as

\[
\begin{align*}
  C^u_{m,n} &= \frac{1}{2\pi} \int_0^{2\pi} \frac{d\lambda}{n} \cos n\lambda u(\lambda, \theta) \\
  C^v_{m,n} &= \frac{1}{2\pi} \int_0^{2\pi} \frac{d\lambda}{n} \sin n\lambda v(\lambda, \theta),
\end{align*}
\]

The variables $u^{c0}_m(\theta)$ and $v^{c0}_m(\theta)$ are meridionally expanded as

\[
\begin{align*}
  u^{c0}_m(\theta) &= \sum_{n=0}^{N} C^{u, c0}_m(\theta) \cos n\theta \\
  v^{c0}_m(\theta) &= \sum_{n=0}^{N} C^{v, c0}_m(\theta) \sin n\theta.
\end{align*}
\]

The equations for $u^{c0}_m(\theta)$ and $v^{c0}_m(\theta)$ for $m \geq 1$ are the same as Eqs. (53) and (54), respectively, except that the subscript $c$ is replaced with $s$. Therefore, the expansion coefficients $C^{u, c0}_m$ and $C^{v, c0}_m$ are calculated from $u^{c0}_m(\theta)$ and $v^{c0}_m(\theta)$ by the
forward discrete cosine or sine transform from the grid-point values of , , , and (See Sect. 2.10). The truncated variables and are defined as

\[ a = \text{constant}, \quad b = \text{constant}, \quad c = \text{constant}, \quad d = \text{constant} \]  

Next, \( x_{n,m}, x_{n,m}^*, \psi_{n,m}^* \), and \( \psi_{n,m} \) are calculated to minimize the following error (the squared \( L_2 \) norm of the residual vector):

\[ F = \frac{1}{2\pi^2} \int_{0}^{\pi} \int_{0}^{2\pi} \left( R_{n,m}^*(\lambda, \theta)^2 + R_{n,m}(\lambda, \theta)^2 \right) d\theta d\lambda, \tag{592} \]

where the residual vector \( \left( R_{n,m}^*(\lambda, \theta), R_{n,m}(\lambda, \theta) \right) \) is defined as

\[ R_{n,m}^*(\lambda, \theta) \equiv u^{n,m}(\lambda, \theta) - u(\lambda, \theta), \tag{594a} \]

\[ R_{n,m}(\lambda, \theta) \equiv v^{n,m}(\lambda, \theta) - v(\lambda, \theta). \tag{594b} \]

From Eqs. (55) to (58) and Eqs. (45a), (46a), and (A3), we derive

\[ F = \frac{1}{2\pi^2} \int_{0}^{\pi} \int_{0}^{2\pi} \left( \frac{1}{2} \sum_{d=1}^{D} \left( u^{d,m}_n(\theta) - u^{d,m}_m(\theta) \right)^2 + \frac{1}{2} \sum_{d=1}^{D} \left( v^{d,m}_n(\theta) - v^{d,m}_m(\theta) \right)^2 \right) d\theta d\lambda, \tag{603a} \]

\[ + \left( \frac{1}{2} \sum_{d=1}^{D} \left( u^{d,m}_n(\theta) - u^{d,m}_m(\theta) \right)^2 + \frac{1}{2} \sum_{d=1}^{D} \left( v^{d,m}_n(\theta) - v^{d,m}_m(\theta) \right)^2 \right) d\theta d\lambda. \tag{603b} \]

From Eqs. (50) to (53), we derive

\[ \frac{\partial F}{\partial x_{n,m}} = 0, \quad \frac{\partial F}{\partial x_{n,m}^*} = 0, \quad \frac{\partial F}{\partial \psi_{n,m}} = 0, \quad \frac{\partial F}{\partial \psi_{n,m}^*} = 0 \]

used in the least-squares method. From Eqs. (50a) and (50b), and the equations \( \partial F / \partial x_{n,m} = 0, \partial F / \partial x_{n,m}^* = 0, \partial F / \partial \psi_{n,m} = 0 \), and \( \partial F / \partial \psi_{n,m}^* = 0 \), we derive

\[ \frac{\partial u^{n,m}(\lambda, \theta)}{\partial x_{n,m}} + \frac{\partial u^{n,m}(\lambda, \theta)}{\partial x_{n,m}^*} = V_{n,m}(\lambda, \theta), \tag{61a} \]

\[ + \frac{\partial v^{n,m}(\lambda, \theta)}{\partial \psi_{n,m}} + \frac{\partial v^{n,m}(\lambda, \theta)}{\partial \psi_{n,m}^*} = V_{n,m}^*(\lambda, \theta). \tag{64a} \]
The (64) right-hand sides of Eqs. (64a, b, c, d) are considered as the new DFS vector basis functions (45x46). Equations (603) and (61) show that the residual vector \( R_{2,0}(\lambda, \theta) \) and \( R_{2,3}(\lambda, \theta) \) is orthogonal to each of the vector basis functions in Eq. (64), which means that Eq. (603) is the same as the equation obtained by the Galerkin method. From Eqs. (603), (614), (51), (485a), (496a), (56), (A3) and (D6), we derive

\[
\pi \sum_{m=-\infty}^{N} \left[ \frac{\partial u_{m}(\lambda, \theta)}{\partial x_m} \right] = 0 \quad \text{for } m = 0, \quad \text{even } m = 2, \quad \text{and } m = 0, \quad \text{odd } m > 1.
\]

From Eqs. (62a) and (64), we derive

\[
\sum_{n=1}^{N} \left[ \frac{\partial}{\partial x_n} \left( u_{m,n} - u_{m,n} \right) \right] = 0 \quad \text{for } m = 0, \quad \text{even } m = 2, \quad \text{and } m = 0, \quad \text{odd } m > 1.
\]

From Eqs. (63) and (65) and from the same equations as Eqs. (69b, c, d), except that \( u_{n,m} \) and \( \psi_{n,m} \) are replaced with \( u_{n,m} \) and \( \psi_{n,m} \), respectively, and the same equations as Eqs. (69a, b, d), except that \( \chi_{n,m} \) and \( \alpha_{n,m} \) are replaced with \( \chi_{n,m} \) and \( \alpha_{n,m} \), respectively, we derive the following equations for \( x_{n,m} \) and \( \psi_{n,m} \) are derived as shown in Appendix H.

For \( m = 0 \),

\[
\frac{1}{a} [n x_{n,m}] = \psi_{n,m} \quad (1 \leq n \leq N).
\]
The coefficient $\chi_{m,n,p}$ is determined so that the global means of $\chi$ are zero. See Eq. (G1) Appendix B—shows the calculation of the global mean.

For $m = 1$,

$$
\frac{1}{a} \left[ (n-1)^2 \chi_{n-2,m} - 2m \psi_{n-1,m} + (4m^2 + 2n^2 + 2) \chi_{n,m} - 2m \psi_{n+1,m} - (n+1)^2 \chi_{n+2,m} \right] = 2(n-1)\psi_{n-1,m} - 4m \psi_{n,m} - 2(n+1)\psi_{n+1,m} \quad (0 \leq n \leq N-1),
$$

(63ab)

with the exception of the following underlined values:

$$
\frac{1}{a} \left[ (8m^2 + 2) \chi_{n,m} - 4m \psi_{n,m} - 2\chi_{n,m} \right] = -8m \psi_{n,m} - 4\psi_{n,m} \quad (n = 0),
$$

$$
\frac{1}{a} \left[ -4m \psi_{n,m} + (4m^2 + 4) \chi_{n,m} + \ldots \right] = \ldots \quad (n = 1),
$$

$$
\frac{1}{a} \left[ -2\chi_{n,m} - 2m \psi_{n,m} + \ldots \right] = \ldots \quad (n = 2).
$$

10 For even $m \geq 2$,

$$
\frac{1}{a} \left[ -(n-1)^2 \chi_{n-2,m} - 2m \psi_{n-1,m} + (4m^2 + 2n^2 + 2) \chi_{n,m} - 2m \psi_{n+1,m} - (n+1)^2 \chi_{n+2,m} \right] = 2(n-1)\psi_{n-1,m} - 4m \psi_{n,m} - 2(n+1)\psi_{n+1,m} \quad (1 \leq n \leq N-1),
$$

(63bc)

with no exception.

For odd $m \geq 3$,

$$
\frac{1}{a} \left[ (n-2)^2 \chi_{n-4,m} + 2m \psi_{n-3,m} + (-4m^2 - 4n^2 - 8n - 8) \chi_{n-2,m} - 2m \psi_{n-1,m} + (8m^2 - 6n^2 + 8) \chi_{n,m}
- 2m \psi_{n+1,m} + (-4m^2 - 4n^2 - 8n - 8) \chi_{n+2,m} + 2m \psi_{n+3,m} + (n + 1)^2 \chi_{n+4,m} \right] = 4(n-2)\psi_{n-2,m} - 8m \psi_{n-1,m} - 8n \psi_{n,m} + 8m \psi_{n+1,m} + 4(n+1)\psi_{n+2,m} \quad (1 \leq n \leq N-2),
$$

(63bd)

with the exception of the following underlined values:

$$
\frac{1}{a} \left[ (12m^2 + 10) \chi_{n,m} - 4m \psi_{n,m} + (-4m^2 - 21) \chi_{n+2,m} + \ldots \right] = -16m \psi_{n,m} - 12\psi_{n,m} + \ldots \quad (n = 1),
$$

$$
\frac{1}{a} \left[ -4m \psi_{n,m} + (8m^2 + 32) \chi_{n,m} + \ldots \right] = \ldots \quad (n = 2),
$$

$$
\frac{1}{a} \left[ (-4m^2 - 21) \chi_{n+2,m} - 2m \psi_{n,m} + \ldots \right] = \ldots \quad (n = 3).
$$

Similarly, from Eq. (6259b) $\psi_{n,m}$, $\psi_{n,m}$, and $\psi_{n,m}$, we derive the same equations as Eqs. (621b, c, d), except that $\chi^0$, $\psi^0$, $\psi^0$, and $\psi^0$ are replaced with $\chi^0$, $\psi^0$, $\psi^0$, and $\psi^0$, respectively. From Eq. (625ac), we derive the same equations as Eqs. (621a, b, c, d), except that $\chi^0$, $\psi^0$, $\psi^0$, and $\psi^0$ are replaced with $-\psi^0$, $\psi^0$, $\psi^0$, and $-\psi^0$, respectively. From Eq. (625d), we derive the same equations as Eqs. (621b, c, d), except that $\chi^0$, $\psi^0$, $\psi^0$, and $\psi^0$ are replaced with $\psi^0$, $\chi^0$, $-\psi^0$, and $-\psi^0$, respectively.
Eq. (63a) is easily solved. From Eqs. (64,b,c,d), and from the same equations as Eqs. (63a,b,c,d), except that \( \chi, \psi, \theta, u \), and \( \bar{u} \) are replaced with \( \psi, \chi, \theta, -u \), and \( -\bar{u} \), respectively, we derive the following linear simultaneous equations with nine-diagonal matrices for each odd \( m \geq 3 \):

\[
\begin{bmatrix}
X_{m} \\
\psi_{m} \\
\omega_{m} \\
\phi_{m} \\
\phi_{m} \\
\end{bmatrix} = \begin{bmatrix}
E_{m} \\
\psi_{m} \\
\omega_{m} \\
\phi_{m} \\
\end{bmatrix},
\]

(62)

where the matrices \( E_{m}, D_{1m} \), and \( D_{2m} \) are nine-diagonal. From Eqs. (63b,c), we derive the equations similar to Eq. (64) for \( m = 1 \) and even \( m \geq 2 \) with penta-diagonal matrices \( E_{m} \), \( \omega_{m} \), and \( \phi_{m} \). We also derive two similar linear simultaneous equations with penta-diagonal matrices for \( m = 1 \) and even \( m \geq 2 \). The simultaneous equations with nine-diagonal or penta-diagonal matrices \( E_{m} \) can be solved in a similar way to Eq. (64), and the expansion coefficients \( \chi_{m} \) and \( \psi_{m} \) in Eq. (63a) can be solved efficiently. From the same equations as Eqs. (64,b,c,d), except that \( \chi, \psi, \theta, u \), and \( \bar{u} \) are replaced with \( \chi, -\psi, \theta, -u \), and \( -\bar{u} \), respectively, and the same equations as Eqs. (64,b,c,d), except that \( \chi, \psi, \theta, u \), and \( \bar{u} \) are replaced with \( -\psi, \chi, -\theta, -u \), and \( -\bar{u} \), respectively, two similar linear simultaneous equations with nine-diagonal matrices for each \( m = 2 \) and two linear simultaneous equations with penta-diagonal matrices for \( m = 1 \) and each even \( m \geq 2 \). The simultaneous equations with nine-diagonal or penta-diagonal matrices \( E_{m} \), \( \psi_{m} \), and \( \phi_{m} \) are obtained calculated from \( \bar{u}_{m} \), \( \psi_{m} \), and \( \phi_{m} \). The expansion coefficients \( \chi_{m} \), \( \psi_{m} \), and \( \phi_{m} \) using Eqs. (64,b,c,d), and the similar equations.

The expansion coefficients \( u_{m} \), \( \psi_{m} \), and \( \phi_{m} \) are calculated obtained from \( \chi_{m} \), \( \psi_{m} \), \( \phi_{m} \), and \( \phi_{m} \) using Eq. (64,b,c,d) for \( u_{m} \) and the similar equations for \( u_{m} \), \( \psi_{m} \), and \( \phi_{m} \).

2.3 Relation between the least-squares method and the Galerkin method for the wind vector

Here we discuss the relation between the least-squares method described above and the Galerkin method when calculating the expansion coefficients related to the wind vector.

From Eqs. (57), (58), and the equations \( \partial E/\partial \psi = 0, \partial E/\partial \phi = 0, \partial E/\partial \phi = 0, \partial E/\partial \psi = 0 \), and \( \partial E/\partial \phi = 0 \) used in the least-squares method, we obtain

\[
\frac{1}{2} \int_{0}^{1} \left[ \frac{\partial E}{\partial \psi} (u, \bar{u}) + \frac{\partial E}{\partial \phi} (u, \bar{u}) \right] dA = 0.
\]

(63a)
The right-hand sides of Eqs. (60a, b, c, d) are considered as the new DFS vector basis functions. Equation (63) shows that the residual vector \( \Phi_0 \) is orthogonal to each of the vector basis functions in Eq. (64), which means that Eq. (64) is the same as the equation obtained by the Galerkin method.

This method to calculate the DFS expansion coefficients of \( \chi \) and \( \psi \) from \( u \) and \( v \) using the least-squares method (or the Galerkin method with the DFS vector basis functions) is similar to the vector harmonic transform method (Browning et al., 1989; Temperton, 1991; Swarztrauber, 1993), where the SH expansion coefficients of the divergence \( D = \nabla \cdot \chi \) and the vorticity \( \zeta = \nabla \times \psi \) are calculated from the grid-point values of \( u \) and \( v \) using the Galerkin spectral method with the orthogonal vector SH basis functions.

### 3.9 Arrangement of equally spaced latitudinal grid points

In DFS models, equally spaced latitudinal grid points are used. We use the following three ways of arranging equally spaced latitudinal grid points:

- **Grid [0]**: \( f = j \Delta f, \quad \theta_j = \pi (j + 0.5)/j, \quad j = 0, \ldots, j^* - 1 
- **Grid [1]**: \( f = j^* + 1, \quad \theta_j = \pi j/j^*, \quad j = 0, \ldots, j^* 
- **Grid [11]**: \( f = j^* - 1, \quad \theta_j = \pi j/j^*, \quad j = 1, \ldots, j^* - 1

10 From Eqs. (45), (46), and (78) we derive

\[
\begin{align*}
\frac{1}{2\pi} \int_0^{2\pi} \frac{d\psi}{d\lambda} \left[ \frac{d}{d\lambda} \left( \frac{\sin m\lambda}{\sin m\psi} \right) \right] d\lambda &= 0, \\
\frac{1}{2\pi} \int_0^{2\pi} \frac{d\psi}{d\lambda} \left[ \frac{d}{d\lambda} \left( \frac{\sin m\lambda}{\sin m\psi} \right) \right] d\lambda &= 0,
\end{align*}
\]

\[
\frac{1}{2\pi} \int_0^{2\pi} \frac{d\psi}{d\lambda} \left[ \frac{d}{d\lambda} \left( \frac{\sin m\lambda}{\sin m\psi} \right) \right] d\lambda = 0.
\]

5 The right-hand sides of Eqs. (60a, b, c, d) are considered as the new DFS vector basis functions. Equation (63) shows that the residual vector \( \Phi_0 \) is orthogonal to each of the vector basis functions in Eq. (64), which means that Eq. (64) is the same as the equation obtained by the Galerkin method.

This method to calculate the DFS expansion coefficients of \( \chi \) and \( \psi \) from \( u \) and \( v \) using the least-squares method (or the Galerkin method with the DFS vector basis functions) is similar to the vector harmonic transform method (Browning et al., 1989; Temperton, 1991; Swarztrauber, 1993), where the SH expansion coefficients of the divergence \( D = \nabla \cdot \chi \) and the vorticity \( \zeta = \nabla \times \psi \) are calculated from the grid-point values of \( u \) and \( v \) using the Galerkin spectral method with the orthogonal vector SH basis functions.

### 3.9 Arrangement of equally spaced latitudinal grid points

In DFS models, equally spaced latitudinal grid points are used. We use the following three ways of arranging equally spaced latitudinal grid points:

- **Grid [0]**: \( f = j \Delta f, \quad \theta_j = \pi (j + 0.5)/j, \quad j = 0, \ldots, j^* - 1 
- **Grid [1]**: \( f = j^* + 1, \quad \theta_j = \pi j/j^*, \quad j = 0, \ldots, j^* 
- **Grid [11]**: \( f = j^* - 1, \quad \theta_j = \pi j/j^*, \quad j = 1, \ldots, j^* - 1

20 From Eqs. (45), (46), and (78) we derive

\[
\begin{align*}
\frac{1}{2\pi} \int_0^{2\pi} \frac{d\psi}{d\lambda} \left[ \frac{d}{d\lambda} \left( \frac{\sin m\lambda}{\sin m\psi} \right) \right] d\lambda &= 0, \\
\frac{1}{2\pi} \int_0^{2\pi} \frac{d\psi}{d\lambda} \left[ \frac{d}{d\lambda} \left( \frac{\sin m\lambda}{\sin m\psi} \right) \right] d\lambda &= 0,
\end{align*}
\]

\[
\frac{1}{2\pi} \int_0^{2\pi} \frac{d\psi}{d\lambda} \left[ \frac{d}{d\lambda} \left( \frac{\sin m\lambda}{\sin m\psi} \right) \right] d\lambda = 0.
\]

From Eqs. (45), (46), and (78) we derive
where $\theta$ is the latitude at each grid point, and $J_0$ is the number of latitudinal grid points in Grid [0]. When the grid intervals in Grids [0], [1], and [−1] are not equal, the number of grid points $J_1$ in Grid [1] is $J_0 - 1$ and the number of grid points in Grid [−1] is $J_0 + 1$. Figure 1 shows Grids [0], [1], and [−1] when $\Delta = \pi$ and the grid interval $\Delta \theta = \pi / 4$. Grid [0] has been widely used in DFS models, for example, in Merilees (1973b), Orszag (1974), Cheong (2000a, 2000b), and Yoshimura and Matsumura (2005). Grid [1] was used, for example, in the DFS expansion in Yee (1981). Grid [−1] was used, for example, in the SH model using Clenshaw–Curtis quadrature in Hotta and Ujihara (2018). All of Grids [0], [1], and [−1] were used in the SH expansion in Swarztrauber and Spotz (2000). We have confirmed that stable integration is possible in the model using the new DFS method with any of Grids [0], [1], and [−1], as shown in Sect. 5 below.

### 2.10 Discrete Fourier cosine and sine transforms in latitude

Forward discrete Fourier cosine and sine transforms are performed in Eqs. (19), (53), and (54), and inverse discrete Fourier cosine and sine transforms are performed in Eqs. (9), (47), and (48), in the latitudinal direction. The calculation of the discrete cosine and sine transforms in Grids [0], [1], and [−1] is shown below. Here, $g_0(\theta)$ and $h_0(\theta)$ are grid-point values, and $g_n$ and $h_n$ are expansion coefficients.

When using Grid [0], forward and inverse discrete cosine transforms are performed as

$$ g_n = \frac{1}{J_0} \sum_{j=0}^{J_0-1} g_0(\theta) \cos n\Delta \theta, \quad h_n = \begin{cases} 1 & \text{for } n = 0, \\ \frac{1}{2} & \text{for } 0 < n < J_0. \end{cases} $$

When using Grid [0], forward and inverse discrete sine transforms are performed as

$$ h_n = \frac{1}{J_0} \sum_{j=0}^{J_0-1} h_0(\theta) \sin n\Delta \theta, \quad g_n = \begin{cases} 1 & \text{for } n = 0, \\ \frac{1}{2} & \text{for } 0 < n < J_0. \end{cases} $$

When using Grid [1], forward and inverse discrete cosine transforms are performed as

$$ g_n = \frac{1}{J_0} \sum_{j=0}^{J_0-1} g_0(\theta) \cos n\Delta \theta, \quad h_n = \begin{cases} 1 & \text{for } 0 < n = J_0, \\ \frac{1}{2} & \text{for } 0 < n < J_0, \end{cases} $$

$$ h_n = \frac{1}{J_0} \sum_{j=0}^{J_0-1} h_0(\theta) \sin n\Delta \theta, \quad g_n = \begin{cases} 1 & \text{for } n = 0, \\ \frac{1}{2} & \text{for } 0 < n < J_0. \end{cases} $$
When using Grid [1], forward and inverse discrete sine transforms are performed as

\[ h_\theta = \frac{2}{J} \sum_{k=1}^{J} h(k) \sin k \theta \]  

\[ h(k) = \sum_{\theta} h_\theta \sin k \theta, h(0) = 0 \]  

Grid [−1] is the same as Grid [1], except that there are no grid points at the North and South poles. The zonal wavenumber components of scalar variables at the poles are zero except for \( m = 0 \) (See Eq. (69a)), and those of vector variables at the poles are zero except for \( m = 1 \) (See Eqs. (47) and (48)). When we use Grid [−1] and the values at the poles are known to be zero, forward and inverse discrete cosine transforms can be performed using Eq. (68) and forward and inverse discrete sine transforms can be performed using Eq. (69) in the same way as for Grid [1]. When we use Grid [−1] and the values at the poles are unknown (i.e., the zonal wavenumber components of scalar variables for \( m = 0 \), and those of vector variables for \( m = 1 \)), the inverse discrete cosine transform can be performed using Eq. (68b), but the forward discrete cosine transform cannot be performed using Eq. (68a). We can calculate the expansion coefficients \( g_m \) from \( g(\theta) \) in the following way. Eq. (68b) is multiplied by \( \sin \theta \), and we define \( g_m(\theta) \) as

\[ g_m(\theta) = g(\theta) \sin \theta = \sum_{k=1}^{J} g_\theta \sin k \theta \cos k \theta \]  

Since the values \( g_m(\theta) \) at the pole \( \theta = 0 \) are zero, we can expand \( g_m(\theta) \) as

\[ g_m(\theta) = \sum_{k=1}^{J} g_\theta \sin k \theta \]  

The expansion coefficients \( g_m \) can be obtained from \( g_m(\theta) \) in the same way as in Eq. (69a) by forward discrete sine transforms

\[ f_m = \sum_{k=1}^{J} g_\theta \sin k \theta \]  

From Eqs. (70) and (71), we obtain

\[ \sum_{k=1}^{J} g_\theta \sin k \theta = \sum_{k=1}^{J} g_\theta \sin k \theta \]  

\[ g_m = \sum_{k=1}^{J} g_\theta \sin k \theta \]  

By using Eq. (72), we obtain

\[ \sum_{k=1}^{J} \sum_{k=1}^{J} g_\theta \sin k \theta \cos k \theta = \sum_{k=1}^{J} \sum_{k=1}^{J} g_\theta \sin k \theta \sin k \theta \sin (J-k) \theta \]  

\[ \sum_{k=1}^{J} \sum_{k=1}^{J} g_\theta \sin k \theta \cos k \theta = 0 \]  

\[ f_m = g_m \]  

\[ g_m = (J-m) \sin \theta + \sum_{k=1}^{J} g_\theta \sin k \theta \]  

Equation (74)
By substituting Eq. (74) into Eq. (73), and comparing the left and right sides of the equation, we obtain

\[
\begin{align*}
\gamma_n &= \frac{\Phi_n - \Phi_{n+1}}{2} \\
\Phi_n &= \frac{\gamma_n + \gamma_{n+1}}{2} \\
\Phi_n &= \gamma_n \\
\Phi_n &= \gamma_{n+1} \\
\Phi_n &= \frac{\gamma_n + \gamma_{n+1}}{2} \\
\end{align*}
\]

(75)

We can calculate \(g'\) from \(g\) using Eq. (70), calculate \(g\) from \(g'\) using Eq. (71), and calculate \(\gamma\) from \(g\) using Eq. (75).

2.11 Zonal Fourier filter

In regular longitude–latitude grids, the longitudinal grid spacing becomes narrow at high latitudes. In DFS methods, the zonal Fourier filter (Merilees 1974; Boer and Steinberg 1975; Cheong 2000a), which filters out the high zonal wavenumber components at high latitudes, is usually used to obtain a more uniform resolution. In this study, we set the largest zonal wavenumber \(M_{zonal}\) as

\[
M_{zonal}(\theta) = \min\{M, M_{\theta} + M \sin(\theta)\}
\]

(76)

where we use the value \(M_{\theta} = 20\) to make the resolution similar to that in the reduced grid of Miyamoto (2006). The values of \(T_{\theta}(\theta)\) and \(T_{\theta}(\theta)\) in Eq. (7) are set to zero for \(m > M_{zonal}(\theta)\) during the spectral transform. The use of a reduced grid (Hortal and Simmons, 1991; Juang, 2004; Miyamoto, 2006) has a similar effect to the zonal Fourier filter.

2.12.3.7 Laplacian operator and Poisson equation

The calculation of the Laplacian operator and the Poisson equation in the new DFS method is described in this section here. In the equation

\[
g(\lambda, \theta) = \nabla^2 f(\lambda, \theta) = \frac{1}{\alpha^2} \sin^2 \theta \frac{\partial^2 f}{\partial \lambda^2} + \frac{1}{\sin \theta} \frac{\partial f}{\partial \theta} \left( \sin \theta \frac{\partial f}{\partial \theta} \right)
\]

(6.522)

where \(\nabla^2\) is the Laplacian operator, the variables \(f\) and \(g\) are expanded zonally using like Eq. (27) as

\[
f = a_0 + \sum_{m=1}^{M} a_m \cos m\lambda
\]

(79)

\[
g = b_0 + \sum_{m=1}^{M} b_m \sin m\lambda
\]

(80)

\[
\begin{align*}
\frac{f(\lambda, \theta)}{g(\lambda, \theta)} &= \frac{\sum_{m=1}^{M} f_m(\theta) g_m^*(\theta)}{\sum_{m=1}^{M} g_m(\theta) g_m^*(\theta)} \cos m\lambda + \frac{\sum_{m=1}^{M} f_m^*(\theta) g_m(\theta)}{\sum_{m=1}^{M} g_m(\theta) g_m^*(\theta)} \sin m\lambda
\end{align*}
\]

(66)

The variables \(f_m(\theta), f_m^*(\theta), g_m(\theta), \) and \(g_m^*(\theta)\) are expanded meridionally using like Eq. (108) as

\[
\begin{align*}
f_m(\theta) &= a_m \cos m\lambda
\end{align*}
\]

(82)

\[
\begin{align*}
g_m(\theta) &= b_m \sin m\lambda
\end{align*}
\]

(83)
The equations for and for \( m \geq 1 \) are the same as Eqs. (80) and (81), respectively, except that the subscript \( c \) is replaced with \( s \). We define the truncated variables \( f_{m}^{N}(\theta) \) and \( g_{m}^{N}(\theta) \) as

\[
\begin{align*}
    &f_{m}^{N}(\theta) = \sum_{n=0}^{N_{\text{MAX}}} f_{m,n}^{N}(\theta) S_{n,m}(\theta) \quad (67) \\
    &g_{m}^{N}(\theta) = \sum_{n=0}^{N_{\text{MAX}}} g_{m,n}^{N}(\theta) S_{n,m}(\theta) 
\end{align*}
\]

\( \equiv \frac{1}{a^2} \sum_{m=0}^{M} \frac{1}{\sin m \lambda} \int_{0}^{\pi} [ \sin m \lambda \frac{\partial f_{m,n}^{N}(\theta)}{\partial \theta} - \frac{m^2}{\sin^2 \theta} f_{m,n}^{N}(\theta) ] \sin m \lambda d \theta \]  

From Eqs. (65a) and (68), we obtain

\[
\begin{align*}
    \nabla \cdot f_{m}^{N}(\lambda, \theta) &= \sum_{n=0}^{N_{\text{MAX}}} \frac{1}{a^2} \int_{0}^{\pi} S_{n,m}(\theta) \left[ \cos m \lambda \frac{\partial f_{m,n}^{N}(\theta)}{\partial \theta} - \frac{m^2}{\sin^2 \theta} f_{m,n}^{N}(\theta) \right] d \theta \\
    &= \sum_{n=0}^{N_{\text{MAX}}} \frac{1}{a^2} \int_{0}^{\pi} S_{n,m}(\theta) \left[ \cos m \lambda \frac{\partial f_{m,n}^{N}(\theta)}{\partial \theta} - \frac{m^2}{\sin^2 \theta} f_{m,n}^{N}(\theta) \right] d \theta 
\end{align*}
\]

Here we use the Galerkin method to calculate the Laplacian operator and the Poisson equation, and obtain

\[
\begin{align*}
    \delta \int_{0}^{\pi} S_{n,m}(\theta) \left[ \cos m \lambda \frac{\partial f_{m,n}^{N}(\theta)}{\partial \theta} - \frac{m^2}{\sin^2 \theta} f_{m,n}^{N}(\theta) \right] d \theta &= 0 \\
    \delta \int_{0}^{\pi} S_{n,m}(\theta) \left[ \cos m \lambda \frac{\partial g_{m,n}^{N}(\theta)}{\partial \theta} - \frac{m^2}{\sin^2 \theta} g_{m,n}^{N}(\theta) \right] d \theta &= 0 
\end{align*}
\]

where the residual

\[
R_{m}(\lambda, \theta) = \frac{1}{2 \pi^2} \int_{0}^{\pi} \frac{S_{n,m}(\theta)}{\sin m \lambda} R_{m}(\lambda, \theta) d \theta d \lambda = 0, 
\]

is orthogonal to each of the new DFS basis functions \( S_{n,m}(\theta) \) can \( m \lambda \) and \( S_{n,m}(\theta) \sin m \lambda \) (see Sect. 2.4).

We can also use the least-squares method described in Sect. 2.3 instead of the Galerkin method so that the following error \( H \) (the squared \( L_{2} \) norm of the residual) is minimized:

\[
H = \frac{1}{2 \pi^2} \int_{0}^{\pi} \int_{0}^{\pi} R_{m}(\lambda, \theta)^2 d \theta d \lambda 
\]

When calculating \( g \) by applying the Laplacian operator to a given \( f \), \( g_{m,n}^{N} \) and \( g_{m,n}^{N} \) can also be calculated from \( \delta H / \delta g_{m,n}^{N} \) and \( \delta H / \delta g_{m,n}^{N} \) using the least-squares method. The equations \( \delta H / \delta g_{m,n}^{N} \) and \( \delta H / \delta g_{m,n}^{N} \) give the equivalent equations to Eq. (70). When calculating \( f \) from a given \( g \) in the Poisson equation, \( f_{m,n}^{N} \) and \( f_{m,n}^{N} \) can also be calculated from \( \delta H / \delta f_{m,n}^{N} \) and \( \delta H / \delta f_{m,n}^{N} \) using the least-squares method. However, the equations derived from \( \delta H / \delta f_{m,n}^{N} \) and \( \delta H / \delta f_{m,n}^{N} \) are different from Eq. (70). If we use different equations for calculating \( g \) from \( f \) and \( f \) from \( g \), the original values are changed when calculating \( g \) from \( f \) followed by calculating \( f \) from \( g \), which may be not good for numerical stability. Therefore, we use Eq. (70) obtained with the Galerkin method for calculating both \( g \) from \( f \) and \( f \) from \( g \). Generally it cannot be said that
the least-squares method is superior to the Galerkin method or vice versa, and here we choose to use the Galerkin method because of the reason described above.

From Eqs. (6788) and Eq. (A3) we derive

$$
\int_0^\pi S_m(\theta) \left( g_m^{(\ell)}(\theta) - \frac{1}{a^2} - \frac{m^2}{\sin^2 \theta} \frac{d \ell}{m} \right) \left( \sin \theta \frac{d f_m^{(\ell)}(\theta)}{d \theta} \right) d\theta = 0,
$$

(7188a)

5

Instead of Eq. (7188b) following Yee (1981) and Cheong (2000a) for ease of calculation. For $0 \leq m \leq 3$, the exact solutions of $g_m^{(\ell)}$ can be obtained from Eq. (7188) because the new DFS meridional basis functions for $0 \leq m \leq 3$ are linear combinations of the associated Legendre functions for $0 \leq m \leq 3$ and vice versa as described in Sect. 3.42.5.

For $m = 0$, we calculate $g_m^{(\ell)}$ by using

$$
g_m^{(\ell)}(\theta) = \frac{1}{a^2} - \frac{m^2}{\sin^2 \theta} \frac{d \ell}{m} \frac{d f_m^{(\ell)}(\theta)}{d \theta} \sin \theta \frac{d f_m^{(\ell)}(\theta)}{d \theta},
$$

(7188b)

instead of Eq. (7188b) following Yee (1981) and Cheong (2000a) for ease of calculation. For $0 \leq m \leq 3$, the exact solutions of $g_m^{(\ell)}$ can be obtained from Eq. (7188) because the new DFS meridional basis functions for $0 \leq m \leq 3$ are linear combinations of the associated Legendre functions for $0 \leq m \leq 3$ and vice versa as described in Sect. 3.42.5.

For $m = 0$, by substituting Eqs. (6788) and (A3) into Eq. (7188) multiplied by $\sin^2 \theta$, transforming using Eqs. (A2d) and (A5b), and comparing both sides of the equation, we obtain

$$
- \frac{1}{a^2} + 2 g_0^{(\ell)}(\theta) - 2 g_0^{(\ell)}(\theta) = \frac{1}{a^2} \left( n \pi^2 - 2 m^2 \right) f_0^{(\ell)}(\theta) + \frac{1}{a^2} \left( n \pi^2 - 2 m^2 \right) f_0^{(\ell)}(\theta),
$$

(7189a)

except for the following underlined values:

1. $g_0^{(\ell)}(\theta) - g_0^{(\ell)}(\theta) = \cdots (n = 1)$,

2. $g_0^{(\ell)}(\theta) + g_0^{(\ell)}(\theta) = \cdots (n = 2)$.

For $m = 1$, by substituting Eqs. (6788) and (A3) into Eq. (7188a) and using Eqs. (A2d), (A4a) and (A5b), we obtain

$$
- \frac{1}{a^2} + 2 g_1^{(\ell)}(\theta) - 2 g_1^{(\ell)}(\theta) = \frac{1}{a^2} \left[ (n - 1) \pi^2 - 2 m^2 \right] f_1^{(\ell)}(\theta) + \frac{1}{a^2} \left[ (n - 1) \pi^2 - 2 m^2 \right] f_1^{(\ell)}(\theta),
$$

(7189b)

except for the following underlined values:

1. $g_1^{(\ell)}(\theta) - g_1^{(\ell)}(\theta) = \cdots (n = 1)$,

2. $g_1^{(\ell)}(\theta) + g_1^{(\ell)}(\theta) = \cdots (n = 2)$.

For even $m \geq 2$, by substituting Eqs. (6788) and (A3) into Eq. (7188a) and using Eqs. (A2c), (A4b) and (A5d), we obtain

$$
- \frac{1}{a^2} + 2 g_m^{(\ell)}(\theta) - 2 g_m^{(\ell)}(\theta) = \frac{1}{a^2} \left[ (n - 1) \pi^2 - 2 m^2 \right] f_m^{(\ell)}(\theta) + \frac{1}{a^2} \left[ (n - 1) \pi^2 - 2 m^2 \right] f_m^{(\ell)}(\theta),
$$

(7189c)

except for the following underlined values:

1. $g_1^{(\ell)}(\theta) - g_1^{(\ell)}(\theta) = \cdots (n = 1)$,

2. $g_1^{(\ell)}(\theta) + g_1^{(\ell)}(\theta) = \cdots (n = 2)$.

For odd $m \geq 3$, by substituting Eqs. (6788) and (A3) into Eq. (7188a) and using Eqs. (A2c,e), (A4b) and (A5d), we obtain
\[
\begin{align*}
\phi_{n-m}^{(c)} - 4\phi_{n+2m}^{(c)} + 6\phi_{2m}^{(c)} - 4\phi_{n+4m}^{(c)} &= \frac{1}{a^2} \left[ -(n-2)(n-1)\phi_{n-4m}^{(c)} + 4(n^2 - 6n + 4 + 4m^2)\phi_{n-2m}^{(c)} - (6n^2 + 4 + 8m^2)\phi_{n,m}^{(c)} \\
+ (4n^2 + 6n + 4 + 4m^2)\phi_{n+2m}^{(c)} - (n + 2)(n + 1)\phi_{n+4m}^{(c)} \right] \quad (1 \leq n \leq N - 2)
\end{align*}
\] (75b)

except for the following underlined values:

\[
\begin{align*}
\phi_{n-4m}^{(c)} &= 1 \\
\phi_{n-2m}^{(c)} &= a^2 - (n - 2)(n - 1) \phi_{n-4m}^{(c)} \\
\phi_{n-4m}^{(c)} &= (n + 2)(n + 1) \phi_{n+4m}^{(c)} \\
\phi_{n+2m}^{(c)} &= (2n + 4 + 4m^2) \phi_{n+2m}^{(c)} \\
\phi_{n+4m}^{(c)} &= (6n^2 + 4 + 8m^2) \phi_{n,m}^{(c)}
\end{align*}
\]

From Eq. (75c), we obtain the following two linear simultaneous equations with tri-diagonal or penta-diagonal matrices:

\[
\begin{align*}
A_{n \text{, even}} g_{n \text{, even}}^{(c)} + B_{n \text{, even}} f_{n \text{, even}}^{(c)} &= A_{n \text{, odd}} g_{n \text{, odd}}^{(c)} + B_{n \text{, odd}} f_{n \text{, odd}}^{(c)} \\
A_{n \text{, odd}} g_{n \text{, odd}}^{(c)} + B_{n \text{, even}} f_{n \text{, even}}^{(c)} &= A_{n \text{, even}} g_{n \text{, even}}^{(c)} + B_{n \text{, odd}} f_{n \text{, odd}}^{(c)}
\end{align*}
\] (76a)

where \( g_{n \text{, even}}^{(c)} \) and \( g_{n \text{, odd}}^{(c)} \) are the vectors whose components are \( g_{n,m}^{(c)} \) (\( n \) is even) and \( g_{n,m}^{(c)} \) (\( n \) is odd), respectively, and \( f_{n \text{, even}}^{(c)} \) and \( f_{n \text{, odd}}^{(c)} \) are the vectors whose components are \( f_{n,m}^{(c)} \) (\( n \) is even) and \( f_{n,m}^{(c)} \) (\( n \) is odd), respectively; \( A_{n \text{, even}}, B_{n \text{, even}}, A_{n \text{, odd}}, B_{n \text{, odd}} \) are tri-diagonal or penta-diagonal matrices, \( g_{n \text{, even}}^{(c)} \) and \( g_{n \text{, odd}}^{(c)} \) are calculated by

\[
\begin{align*}
\phi_{n,m}^{(c)} &= A_{n \text{, even}} g_{n,m}^{(c)} + B_{n \text{, even}} f_{n,m}^{(c)} \\
\phi_{n,m}^{(c)} &= A_{n \text{, odd}} g_{n,m}^{(c)} + B_{n \text{, odd}} f_{n,m}^{(c)}
\end{align*}
\] (77)

which can be solved efficiently as in Eq. (3125).

By using Eq. (87b) instead of Eq. (87a), we obtain the equations to calculate \( f \) from \( g \), which are the same as Eqs. (88) to (91), except that the superscript \( c \) is replaced with the superscript \( s \).

We have verified that all the eigenvalues of the matrices \( A_{n \text{, even}}^{-1} B_{n \text{, even}} \) and \( A_{n \text{, odd}}^{-1} B_{n \text{, odd}} \) are negative real numbers for several truncation wavenumbers \( M \) and \( N \), but we have not yet proved that this is true for all truncation wavenumbers.

In the Poisson equation, \( f \) is calculated from given \( g \) in Eq. (65c). We calculate \( f \) from \( g \) by the reverse calculation of \( g \) from \( f \) in Eq. (72a). That is, we calculate \( f \) from \( g \) by

\[
\begin{align*}
f_{n \text{, even}}^{(s)} &= B_{n \text{, even}}^{-1} A_{n \text{, even}} g_{n \text{, even}}^{(s)} \\
f_{n \text{, odd}}^{(s)} &= B_{n \text{, odd}}^{-1} A_{n \text{, odd}} g_{n \text{, odd}}^{(s)}
\end{align*}
\] (79)

except when \( m = 0 \) and \( n \) is even. For \( m = 0 \), \( f_{n \text{, even}}^{(s)} \) disappears in Eq. (75a). The coefficients \( f_{n \text{, even}}^{(s)} \) (even \( n \geq 2 \)) are calculated from \( g_{n \text{, even}}^{(s)} \) (even \( n \geq 2 \)) by Eq. (75a). The value \( f_{0,0}^{(s)} \) is calculated from \( f_{0,0}^{(s)} \) (even \( n \geq 2 \)) so that the
global mean of $f$ is zero using Eq. (GK1). and are also calculated from the equations where the superscript $c$ in Eq. (62) are replaced with the superscript $s$.

In Eq. (65), the global mean of $g$ must be zero because the global mean of the right-hand side of Eq. (65) is zero. Before calculating $f$ from a given $g$ in the Poisson equation, we should subtract the global mean from $g$ (Cheong 2000b).

See Eq. (GK1) about the calculation of the global mean.

2.14.3.3 The Helmholtz equation

From Eq. (91), Eq. (77) is represented as

$$g = Bf,$$

(92)

where the subscripts $m$, $n_{even}$ and $n_{odd}$, and the superscripts $c$ and $s$ are omitted. The matrix $B$ represents the Laplacian operator in spectral space.

The Helmholtz equation is

$$f - \epsilon \nabla^2 f = \left(1 - \frac{\epsilon}{a}\right) \frac{1}{\sin^2 \theta} \frac{\partial^2}{\partial \lambda^2} + \frac{\partial}{\sin \theta \partial \theta} \left(\sin \theta \frac{\partial}{\partial \theta}\right) f = g,$$

(7944)

where $f$ is calculated from $g$. From Eq. (76), the Poisson equation in Eq. (65) is represented as

$$A g = B f,$$

(80)

where the subscripts $n_{even}$, $n_{odd}$ and $m$, and the superscripts $c$ and $s$ are omitted. Similarly, by using the Galerkin method, Eq. (7944) is represented as

$$Af - \epsilon Bf = Ag.$$

(81)

From Eq. (81), $f$ is calculated from $g$ by

$$f = (A - \epsilon B)^{-1} Ag.$$

(82)

Since $A - \epsilon B$ is a penta-diagonal or tri-diagonal matrix, Eq. (82) can be efficiently solved as in Eq. (312).

Similarly, the Helmholtz-like equation

$$f - \epsilon \nabla^2 f = \nabla^2 g,$$

(83)

is represented as

$$Af - \epsilon Bf = Bg.$$

(84)

From Eq. (84), $f$ is calculated from $g$ by

$$f = (A - \epsilon B)^{-1} Bg.$$

(85)

2.14.3.9 Horizontal diffusion

The horizontal diffusion is calculated in the same similar way as in Cheong et al. (2004). Here we describe how to calculate fourth-order diffusion. Higher-order diffusion can be calculated similarly.

The equation for fourth-order hyperdiffusion is
\[ f + i \sqrt{\epsilon} \nabla f = g, \quad (100) \]

where \( f \) is calculated from \( g \). Equation (86) can be converted into

\[ (1 + i \sqrt{\epsilon} \nabla^2)(1 - i \sqrt{\epsilon} \nabla^2)f = g, \quad (87) \]

where \( i = \sqrt{-1} \). The calculation of Eq. (86) is accomplished by successive calculations of the following Helmholtz equations:

\[ (1 + i \sqrt{\epsilon} \nabla^2)f' = g, \quad (88a) \]

\[ (1 - i \sqrt{\epsilon} \nabla^2)f = f', \quad (88b) \]

which are represented as

\[ (A + i \sqrt{\epsilon} B)f' = Ag, \quad (89a) \]

\[ (A - i \sqrt{\epsilon} B)f = Af'. \quad (89b) \]

Equation (101) is represented as

\[ f = g. \quad (102) \]

From Eqs. (89), we obtain the equation to calculate \( f \) from \( g \) as

\[ f = (A - i \sqrt{\epsilon} B)^{-1} A (A + i \sqrt{\epsilon} B)^{-1} Ag. \quad (90) \]

Here, \( A - i \sqrt{\epsilon} B \) and \( A + i \sqrt{\epsilon} B \) are complex matrices and \( f \) and \( g \) are real column vectors. For efficient computation, two real column vectors can be combined into one complex column vector (Cheong et al., 2004); for example, \( f = f^c + i f^s \) and \( g = g^c + i g^s \), where the superscript \( c \) indicates the zonal cosine component, and the superscript \( s \) indicates the zonal sine component.

The error due to meridional wavenumber truncation in DFS expansion methods

Here we examine the error due to the meridional wavenumber truncation when the same continuity conditions at the poles as Eq. (3) are satisfied. In the DFS method of Orszag (1974) using Eq. (2), only \( f_{-1,n} \) and \( f_{0,n} \) are modified to satisfy Eq. (4) equivalent to Eq. (3). In the old DFS method using Eq. (6), which is proposed in Cheong (2000a, 2000b) and used in Yoshimura and Matsumura (2005), the DFS meridional basis functions of Cheong in Eq. (6) automatically satisfy the pole conditions in Eq. (3) for even \( m \), but not for odd \( m \). In the new DFS method using Eq. (10), the DFS meridional basis functions in Eq. (8) automatically satisfy the condition in Eq. (3) for both even and odd \( m \). We examine the error due to the wavenumber truncation among these DFS methods while comparing it with the SH method.

Figure 2 shows the error due to the wavenumber truncation, when we use Grid [0] (see Sect. 2.9) with the number of latitudinal grid points is \( J = 64 \). The initial conditions of \( F_\theta(\theta) \) are set to one at the grid points north of 30°N (except for the North pole), and zero at the grid points south of 30°N. Grid [0] is used in the DFS methods, and the Gaussian grid is
used in the SH methods. There are no grid points at the poles. Since the values at the poles are zero due to the pole conditions in Eq. (3), the initial values abruptly change around the North pole. The initial values are meridionally transformed from grid space to spectral space (forward transform), truncated with $N = 42$, and then transformed back from spectral space to grid space (inverse transform) to obtain the truncated reconstruction of $f_m(\theta)$. In the DFS method of Orszag using Eq. (2), a very large error occurs, especially for odd $|m| \geq 3$ (Fig. 2c), when $f_{-1,m}$ and $f_{0,m}$ are modified to satisfy the pole conditions in Eq. (4). Dividing $f_m(\theta)$ by $\sin \theta$ before the forward Fourier cosine transform for odd $m$ also contributes to the large error.

In the old DFS method of Cheong using Eq. (6), large high-wavenumber oscillations appear for even $m \neq 0$ in Fig. 2a. Although the basis functions in the method of Cheong for even $m \neq 0$ in the old DFS method are the same as those in the new method, the expansion coefficients are calculated differently in the two methods. In the old DFS method of Cheong, the simple meridional truncation with $N < f$ after the forward Fourier sine transform of a variable divided by $\sin \theta$ causes the large high-wavenumber oscillations. The large oscillations appear especially when the initial values abruptly change around the poles. In the case shown in Fig. 2, the initial values near the North Pole are one, but the value at the North Pole abruptly becomes zero due to the pole conditions of Eq. (3). When $N = f$ for even $m \neq 0$, the forward transform followed by the inverse transform does not change the original values, and the oscillations do not appear. For this reason, Yoshimura and Matsumura (2005) and Yoshimura (2012) set $N = f$ for even $m$ to improve stability. The result in the old DFS method of Cheong for odd $|m| \geq 3$ is not shown in Fig. 2c because the method does not satisfy the condition of Eq. (3) for odd $m$.

In the new DFS method described in Sect. 32, the usual small oscillations from the Gibbs phenomenon appear in Fig. 2, but the error is small because the expansion coefficients are calculated using the least-squares method (or the Galerkin method) to minimize the error. Because of this, the truncation with arbitrary $N < f$ does not cause large oscillations in the new DFS method. The values for even $m \geq 2$ and odd $m \geq 3$ in the new DFS method are similar to those for $m = 2$ and $m = 3$ in the SH method, respectively. In the SH method, when $m$ is large, the values become close to zero at high latitudes.

Even when using the basis functions of Orszag in Eq. (2), we can also obtain results equivalent to the new DFS method by calculating the expansion coefficients using the least-squares method with Lagrange multipliers to minimize the error while satisfying the pole conditions in Eq. (4). Figure 3a shows the same figure as Fig. 2a except for $N = 63$. In the old DFS method using Eq. (6), we set $N = 63$ for $m = 0$, and $N' = 64$ for $m \neq 0$. Because $N = f$ for even $m \geq 2$, the forward transform followed by the inverse transform does not change the original values at the grid points, and the oscillations do not appear in the old DFS method. For this reason, Yoshimura and Matsumura (2005) and Yoshimura (2012) set $N = f$ for even $m \geq 2$ to improve stability. However, there is a problem with the latitudinal derivative in the old DFS method even when $N = f$ for even $m \geq 2$. Fig. 3b is the same as Fig. 3a except that it also shows the values between grid points calculated from the expansion coefficients by using...
Eq. (6) or Eq. (10). The large oscillations appear in the old DFS method with Grid [0], and it makes the latitudinal derivative at the grid points unrealistically large. In the new DFS method with the least-squares method, the large oscillations do not appear.

5 Tests of the DFS methods with the Laplacian operator and the Helmholtz equation

We examine the accuracy of the old and new DFS methods for the Laplacian operator in Eq. (6577) and the Helmholtz equation

\[(1 - \epsilon \nabla^2)f = h.\]  \hspace{1cm} (91)

Here, we give the function \(f\) as

\[f = \begin{cases} \frac{H}{R} \left( 1 + \cos \frac{\pi r}{R} \right)^2 & \text{if } r < R \\ 0 & \text{if } r \geq R \end{cases} \]  \hspace{1cm} (92)

\[r = a \cos^{-1}(\sin \phi \sin \phi + \cos \phi \cos \phi \cos(\lambda - \lambda_0)).\]  \hspace{1cm} (93)

where \(H = 1000, R = a/3, \phi\) is latitude, \(a\) is longitude, \(\sigma\) is the radius of the earth and \(r\) is the distance between \((\lambda, \phi)\) and the center \((\lambda_0, \phi_0) = (3\pi/2, \pi/2 - 0.05)\). The function \(f\) is similar to the cosine bell in the Williamson test case 1, but \((1 + \cos \frac{\pi r}{R})^2\) is squared so that the second derivative of \(f\) is continuous. To easily calculate the exact values of \(\nabla^2 f\), the center is temporarily set to the North Pole, that is, \((\lambda_0, \phi_0) = (0, \pi/2)\) and \(r = a \cos^{-1}(\sin \phi) = a \theta\), where \(\theta\) is colatitude.

At this time, \(g\) is calculated as follows:

\[g = \nabla^2 f = 1 - \epsilon \nabla^2 f \]  \hspace{1cm} (94)

\[= \frac{1}{a^2} \left( \frac{1}{\sin^4 \theta} \frac{\partial^2 f}{\partial \theta^2} + \frac{1}{\sin^2 \theta} \frac{\partial f}{\partial \theta} \cos \theta \frac{\partial f}{\partial \theta} \right) \]  \hspace{1cm} (95)

Equation (R1694) is satisfied at any position of the center. The function \(h\) in Eq. (R1391) is calculated by

\[h = (1 - \epsilon \nabla^2 f) = f - \epsilon g.\]  \hspace{1cm} (96)

where \(\epsilon = 0.01 a^2\), and \(f\) and \(g\) are given by Eqs. (92) and (94).

To examine the accuracy for the Laplacian operator, \(f\) is given by (92), and \(\nabla^2 f\) is calculated from \(f\) with the old DFS method (Cheong 2000a), the new DFS method (See Sect. 3.7) and the SH method. The calculated values are compared with the exact values of \(\nabla^2 f\) in Eq. (94). Here, the exact values of \(\nabla^2 f\) are truncated by the forward transform followed by the inverse transform in order to see the error that does not include the error due to the wavenumber truncation. Table 1 shows the normalized \(L_2\) error between the calculated values and the exact values, which is normalized by the \(L_2\) norm of the exact values. The differences in error between the methods are small, but the results of the SH method are a little better than the old and new DFS methods. Table 2 shows the global mean of calculated \(\nabla^2 f\). The exact value of the global mean of \(\nabla^2 f\) is zero. In Table 2, the global means calculated with each method are very close to zero. The global means of \(\nabla^2 f\) in the DFS methods using Grid [1] and Grid [-1] are not as close to zero as those in the DFS methods using Grid [0] and the SH method.
This seems to be because the accuracy of the meridional discrete cosine and sine transforms in the DFS methods using Grid [1] and Grid [-1] is not as good as that in the DFS methods using Grid [0].

To examine the accuracy of the solution of the Helmholtz equation, \( h \) is given in Eq. (95) and the Helmholtz equation in Eq. (91) is solved with the old DFS method (Cheong 2000a), the new DFS method (See Sect. 3.8) and the SH method. The calculated values are compared with the exact solution \( f \) in Eq. (92). The exact values of \( f \) are also truncated as described above. Table 3 shows the normalized \( L_2 \) error between the calculated values and the exact values. The differences in error between the methods are small, and which is better depends on the resolution and the arrangement of the grid points.

4 Shallow water model on a sphere

4.1 Equations

The prognostic equations of the shallow water model on a sphere are

\[
\frac{dv}{dt} = -\gamma (\Omega \times v) + gh
\]

\( \Omega \times v \) the horizontal wind vector, \( h \) is the height, \( h_0 \) is the surface height, \( g \) is the acceleration due to gravity, \( \Omega \) is the 3-dimensional angular velocity of the earth’s rotation, and the subscript \( H \) indicates the horizontal component.

Equation (104) is transformed for the advective treatment of the Coriolis term (Temperton, 1997) into

\[
\frac{dv}{dt} = -gh
\]

where \( r \) is the 3-dimensional position vector from the Earth’s center. Equation (105) is transformed for the spatially averaged Eulerian treatment of mountains (Ritchie and Tanguay, 1996) into

\[
\frac{dh}{dt} = -gh - h_0 \nabla \cdot v + \nabla \cdot \frac{gh}{2}
\]

4.2 Time integration method

A two-time-level semi-implicit semi-Lagrangian scheme (e.g., Temperton et al., 2001) and the Stable Extrapolation Two-Time-Level Scheme (SETTLS; Hortal, 2002) are adopted to discretize Eqs. (106) and (107) in time

\[
\frac{h + h_0}{2} \Delta t = \frac{h(h + h_0) + h_0(h + h_0)}{2}
\]

\[
\frac{h}{2} \Delta t = \frac{h(h + h_0) + h_0(h + h_0) + \gamma (\Omega \times v) + gh}{2}
\]
\[
\begin{align*}
\mathbf{D} & = \nabla \cdot \mathbf{v} = a \frac{\partial u}{\partial \lambda} + \cos \phi \left( \frac{\partial v}{\partial \phi} \right) \\
\Delta t & \text{ is a timestep; the superscripts } -1, 0, \text{ and } 1 \text{ mean past time } (t - \Delta t), \text{ present time } (t), \text{ and future time } (t + \Delta t), \text{ respectively, and the superscript } (\pm) \text{ means future time } (t + \Delta t) \text{ extrapolated in time, for example, } \frac{\partial u}{\partial \lambda} \\
\text{ is horizontal divergence; } \\
\text{ is a constant value of height for semi-implicit linear terms; } \beta_{\mathbf{v}} \text{ and } \beta_{\psi} \text{ are second-order decentering parameters (Yukimoto et al., 2011). Using } \beta_{\mathbf{v}} \text{ and } \beta_{\psi} \text{ larger than 1.0 (e.g., 1.2) increases the effect of the semi-implicit scheme improving computational stability, but } \beta_{\mathbf{v}} = \beta_{\psi} = 1.0 \text{ is used here because } \frac{\partial u}{\partial \lambda} \text{ is enough for stable calculations in the shallow water model.}
\end{align*}
\]

The departure point \( \mathbf{x} \) is the upstream horizontal position from the arrival point along the wind vector between present time \( t \) and future time \( t + \Delta t \). Here, the arrival point \( \mathbf{x} \) is on a grid point, and the departure point \( \mathbf{x} \) is not generally on a grid point. Since the right-hand sides of Eqs. (108) and (109) are the time average between present time \( t \) and future time \( t + \Delta t \) and the spatial average between the departure point and the arrival point, these equations have second-order precision in time and space. In SETTLS, \( \mathbf{x} \) is calculated using

\[
\mathbf{x} = \mathbf{x} - \frac{\mathbf{v}_0 + \mathbf{v}_{\pm}}{2} \Delta t.
\]  

However, when \( \Delta t \) is longer than 30 minutes, using \( \mathbf{x} \) extrapolated in time to calculate \( \mathbf{x} \) causes numerical instability in our experiments. To avoid instability when \( \Delta t \) is 1 hour, here we use

\[
\begin{align*}
\mathbf{x} &= \mathbf{x} - \frac{\mathbf{v}_0 + \mathbf{v}_{\pm}}{2} \Delta t \\
\mathbf{v}_{\pm} &= \mathbf{v}_0 + (7D \times \mathbf{r})_\omega - 7D \times \mathbf{r} - \frac{\partial \mathbf{u}(\mathbf{x}_0 + \Delta \mathbf{x})}{\partial t} \Delta t.
\end{align*}
\]

\[\text{(112a)}\]
\[\text{(112b)}\]

This method using a provisional future value to calculate \( \mathbf{x} \) is similar to the method in Gospodinov et al., (2001). Since the value with the subscript \( \mathbf{D} \) depends on \( \mathbf{x}_0 \), \( \mathbf{x}_0 \) is calculated iteratively from Eq. (113). (e.g., Ritchie, 1985; Temperton et al., 2001). Since \( \mathbf{x} \) is not generally on the grid point, the value at \( \mathbf{x} \) is calculated by spatial interpolation from nearby grid points. In the right-hand side of Eq. (113), the value at \( \mathbf{x} \) with the subscript \( \mathbf{D} \) is calculated by third-order Lagrange interpolation.
Eqs. (108) and (109) are transformed into
\[
\frac{\partial \mathbf{v}}{\partial t} = \frac{1}{2} \nabla \times \left( \mathbf{v} \times \mathbf{v} \right) - \mathbf{R}
\]
(114a)
\[
\frac{\partial t}{\partial t} \frac{\partial \mathbf{v}}{\partial t} - \mathbf{R}
\]
(114b)
\[
\frac{\partial \mathbf{v}}{\partial t} \cdot \mathbf{v} - \mathbf{f} \cdot \mathbf{v} - \mathbf{f} \cdot \mathbf{v}
\]
(115a)
\[
\mathbf{R} = \mathbf{v} \cdot \nabla \mathbf{v} - \frac{\partial \mathbf{v}}{\partial t} \cdot \mathbf{v} - \mathbf{f} \cdot \mathbf{v} - \mathbf{f} \cdot \mathbf{v}
\]
(115b)

In Eqs. (114a) and (114b), the values at \( t_D \) with the subscript D are calculated by fifth-order and third-order Lagrange interpolations, respectively, since high-order interpolation of wind vector components increases the accuracy of the model's results in our experiments. From Eq. (114a), we obtain
\[
\frac{\partial \mathbf{v}}{\partial t} + \nabla \times \left( \mathbf{v} \times \mathbf{v} \right) = \mathbf{R}
\]
(116)
\[
\zeta = \mathbf{R}
\]
(117)
where
\[
\zeta = \mathbf{v} \cdot \nabla \mathbf{v}
\]
(118)

We calculate \( \zeta \) and \( \mathbf{v} \) using the spectral transform method and the Galerkin method with the new DES method as follows:

1. The vector variable \( \mathbf{R} \) is transformed from grid space to spectral space using Eqs. (19) to (25). The components of the vector variable \( \mathbf{R}_D = \left( R_1, R_2 \right) \) in grid space are transformed to \( R_D \) and \( R_p \) in spectral space using Eqs. (51) to (62), where \( R_D \) and \( R_p \) are the velocity potential and the stream function of \( \mathbf{R} \), respectively.

2. \( R_p \) and \( R_d \) are calculated by
\[
\mathbf{R}_p = \nabla \times \mathbf{R}_D
\]
(119)
\[
\mathbf{R}_d = \nabla \times \mathbf{R}_D
\]
(120)

Using Eqs. (98) and (99), \( \mathbf{R} \) is obtained from \( \mathbf{R}_p \) and \( \mathbf{R}_d \) using Eq. (115a).

3. Equations (115a) and (115b) are substituted into Eq. (116) and we obtain
\[
\frac{\partial \mathbf{v}}{\partial t} + \nabla \times \left( \mathbf{v} \times \mathbf{v} \right) - \mathbf{R}
\]
(121)

\( \mathbf{v} \) is calculated by solving the Helmholtz-like equation Eq. (121) using Eqs. (97) and (99).

4. \( \mathbf{R} \) is calculated from \( \zeta \) and \( \mathbf{v} \) using Eq. (115a).
5. \( \psi \) and \( \phi \) are calculated from \( D \) and \( \zeta \) by solving the Poisson equations
\[
\nabla^2 \psi = D, \tag{122}
\]
\[
\nabla^2 \phi = \zeta, \tag{123}
\]
using Eqs. (80) and (82).

6. \( \mathbf{v} = (u, v) \) is calculated from \( \psi \) and \( \phi \) using Eq. (124) for \( \psi \) and the similar equations for \( \phi \). \( D \) and \( \nabla^2 D \) in spectral space are transformed to grid space. \( h \) and \( D \) are transformed meridionally using Eqs. (83) and (106). \( \psi \) and \( \phi \) are transformed meridionally using Eq. (124). \( \psi \) can also be calculated from \( h \) and \( \phi \) at the latitudinal grid points using Eqs. (124) and (125), and additionally using Eq. (116) at the pole when using Grid 1, which is more efficient than using Eqs. (124) and (125) because the meridional inverse discrete cosine and sine transforms of \( \psi \) become unnecessary.

6.5 Evaluation of the DFS methods using shallow water test cases

We ran the Williamson test cases 1, 2, 5 and 6 (Williamson et al., 1992), and the Galewsky test case (Galewsky et al., 2004) in the model using the new DFS method described in Sect. 3, the model using the old DFS method of Yoshimura and Matsumura (2005), and the model using the SH method. By comparing the results of these models, we evaluated the old and new DFS methods.

6.5.1 Shallow water equations on a sphere

The prognostic equations of the shallow water model on a sphere are
\[
\frac{d\mathbf{v}}{dt} = -2\Omega \times \mathbf{v} - g\nabla h, \tag{96104}
\]
\[
\frac{d(h - h_s)}{dt} = -(h - h_s)\mathbf{v} \cdot \mathbf{v}. \tag{97105}
\]

where \( t \) is time, \( \mathbf{v} \) is the horizontal wind vector, \( h \) is the height, \( h_s \) is the surface height, \( g \) is the acceleration due to gravity, \( \Omega \) is the 3-dimensional angular velocity of the earth’s rotation, and the subscript \( H \) indicates the horizontal component.

Equation (96104) is transformed for the advective treatment of the Coriolis term (Temperton, 1997) into
\[
\frac{d(\mathbf{v} + 2\Omega \times \mathbf{r})}{dt} = -g\nabla h, \tag{98106}
\]

where \( \mathbf{r} \) is the 3-dimensional position vector from the Earth’s center. Equation (10597) is transformed for the spatially averaged Eulerian treatment of mountains (Ritchie and Tanguay, 1996) into
\[
\frac{dh}{dt} = -(h - h_s)\nabla \cdot \mathbf{v} + \mathbf{v} \cdot \nabla h_s. \tag{99107}
\]
Equations (98.106) and (99.107) are integrated in time using a two-time-level semi-implicit semi-Lagrangian scheme (See Appendix I).

6.2 Models

We ran the shallow water test cases Williamson test cases 2 and 5 (Williamson et al., 1992) and the Galewsky test case (Galewsky et al., 2004) in the semi-implicit semi-Lagrangian shallow water model or the Eulerian advection model (See Sect. 5) using the new improved DFS method described in Sect. 2 (hereafter the new DFS model). We also ran the same test cases in the semi-implicit semi-Lagrangian shallow water model using the old DFS method of Yoshimura and Matsumura (2005) with the basis functions of Cheong (2000a, 2000b) (hereafter the old DFS model), and in the model using the SH method (hereafter the SH model) for comparison. The new DFS model was run for each of Grid [0], [1], and [−1]. In the old DFS model, Grid [0] was used. In the SH model, the Gaussian grid was used. We use a regular longitude-latitude grid, $\Delta t = 3600$ s at about 300 km resolution with $f^0 = 64$ around $128 \times 64$ grid points, $\Delta t = 1800$ s at about 120 km resolution with $f^0 = 160$, $\Delta t = 1200$ s at about 60 km resolution with $f^0 = 320$, $\Delta t = 600$ s at about 20 km resolution with $f^0 = 960$, $\Delta t = 1200$ s at about 1.3 km resolution with $f^0 = 1920$ grid points, where $f^0$ is the number of latitudinal grid points in Grid [0]. The number of latitudinal grid points $f = f^0$ in Grid [0] (and in the Gaussian grid), $f = f^0 + 1$ in Grid [1], and $f = f^0 − 1$ in Grid [−1] (See Sect. 2). The number of longitudinal grid points is $I = 2f^0$, the number of longitudinal grid points is $f = 128$ and the number of latitudinal grid points is $f = 64$. The meridional truncation wavenumber $N$ and the zonal wavenumber $M$ are set to be equal. In the Eulerian advection model, shorter timesteps are used as shown in Sect. 6.3. Horizontal diffusion is not used in all test cases. The zonal Fourier filter described in Appendix F is used in the DFS models. We have confirmed that numerical instability occurs in some test cases in the old DFS shallow water model without the zonal Fourier filter, but stable integration is possible in all test cases shown here in the new DFS shallow water model, even without the zonal Fourier filter.

The zonal Fourier transforms in all of the models and the meridional Fourier cosine and sine transforms in the DFS models are calculated using the Netlib BIHAR library, which includes a double precision version of the Netlib FFTPACK library (Swartztrauber, 1982). The meridional Legendre transform in the SH model is calculated using the ISPACK library (Ishioka, 2018), which adopts on-the-fly computation of the associated Legendre functions. We use the ISPACK library’s optimization option for Intel AVX512, which is highly optimized by using assembly language together with Fortran.

6.3 Williamson test case 1

The Williamson test case 1 simulates a cosine-bell advection. In the semi-Lagrangian models, the advection is calculated in the semi-Lagrangian scheme and the horizontal derivatives calculated from the expansion coefficients are not used for the...
advection calculation. Therefore, we also use the Eulerian scheme here to simulate the advection in the DFS and SH models to test the expansion methods. The advection equation is

$$\frac{dh}{dt} = \frac{\partial h}{\partial t} + \mathbf{v} \cdot \nabla h.$$  \hspace{1cm} (100)

In the Eulerian models, the advection term $\mathbf{v} \cdot \nabla h$ is evaluated using the spectral transform method. The advection equation is integrated by the leap-frog scheme with the Robert-Asselin time filter (Robert, 1969; Asselin, 1972) with a coefficient of 0.05. The horizontal diffusion is not used, but the zonal Fourier filter is used in the old and new DFS methods. In Eq. (100), the value $M_0 = 20$ is used in the DFS shallow water models. However, the larger the value $M_0$ is, the higher the longitudinal resolution around the pole is. Because of this, when the Eulerian scheme is used and $M_0$ is large, a timestep must be very short due to the CFL condition. Therefore $M_0$ should be as small as possible. We have tested $M_0 = 0$, but this degrades the result of the Williamson test case 1. We have also tested $M_0 = 1$, and this result is good. Therefore, we use $M_0 = 1$ in the Eulerian models.

Figure 4 shows the predicted height after a 12-day integration in the Williamson test case 1 when using the Eulerian advection models at the resolution $J_0 = 64$. The meridional truncation wavenumber $N$ and the zonal truncation wavenumber $M$ are set as $N = M = 42 \approx 2J^3/3$ because the 2/3 rule (Orszag, 1971) is used in order to avoid aliasing in the nonlinear advection term. The timestep is 30 minutes. The results for DFS [0], DFS [1], DFS [−1] and SH are very similar. Instability occurs in the old DFS model without horizontal diffusion. This is probably because of the appearance of high-wavenumber oscillations due to the wavenumber truncation with $N \approx 2J^3/3$ for even $m \neq 0$ in the old DFS method, as shown in Sect. 4. Table 4 shows the normalized $L_2$ errors of the predicted height after a 12-day integration when using the Eulerian advection models. The timesteps are 30, 15, 7.5, and 2.5 minutes at the resolution $J_0 = 64, 160, 320$ and 960 ($N = 42, 106, 213$ and 639), respectively. The errors are very close between the models at each resolution. At the resolution $N = 639$, the new DFS model without horizontal diffusion is unstable when the timestep is 200 seconds. The SH model without horizontal diffusion is stable when the timestep was 240 seconds and unstable when the timestep is 300 seconds. One reason for this difference in timestep is probably that the longitudinal resolution near the poles is higher in the new DFS model with $M_0 = 1$ than in the SH model. When the fourth order horizontal diffusion in Eq. (86) with $\varepsilon = 7.2(107/(N + 1))^2$ is used, the old DFS and SH models are stable when the timestep is 240 seconds and are unstable when the timestep is 300 seconds. The old DFS model is unstable even when the same fourth order horizontal diffusion is used. Higher-order horizontal diffusion, which effectively smooths out the high wavenumber components, is necessary to stabilize the Eulerian old DFS model (Cheong, 2000b; Cheong et al., 2002).

Table 5 shows the same as Table 4 except for using the semi-Lagrangian scheme. In the semi-Lagrangian models, the forward transform followed by the inverse transform are executed at every timestep, but the expansion coefficients are not used for the advection calculation. The timesteps are the same as described in Sect. 6.2. The errors are very close between the models. At the resolution $J_0 = 64$, the errors in the semi-Lagrangian models are larger than those in the Eulerian models.
but at the resolutions $j^0 = 160, 320$ and 960, the errors in the semi-Lagrangian models are smaller than those in the Eulerian models.

The conservation of mass in the Williamson test case 1 was also examined, and the results are shown in Sect. S2 in the supplement.

### Williamson test case 2

The Williamson test case 2 simulates a steady state non-linear zonal geostrophic flow. In this test case, the angle between the solid body rotation and the polar axis $\alpha$ is given, and the zonal and meridional components of $2\Omega \times r$ become

\[
2\Omega \times r = \left( 2\Omega \cos \theta \cos \alpha + \cos \lambda \sin \theta \sin \alpha \right), -2\Omega \sin \lambda \sin \alpha \right) .
\]  

Figure 53 shows the time series of forecast errors of the height for a 5-day integration in the Williamson test case 2 with $\alpha = \pi/2 - 0.05$ in the models at the resolutions $j^0 = 64$, using no horizontal diffusion. The normalized $L_1, L_2,$ and $L_\infty$ errors are almost the same among between the new DFS models using Grids [0], [1] and [−1], the old DFS model, and the SH model. Table 6 shows the normalized $L_2$ errors of the predicted height after a 5-day integration. The errors are almost the same between the old DFS, new DFS, and SH models at each resolution.

### Williamson test case 5

The Williamson test case 5 simulates zonal flow over an isolated mountain. Figure 64 shows the predicted height after a 15-day integration in Williamson test case 5 with $h_0 = 5960$ m. The result of the high-resolution SH model at the resolutions $j^0 = 960$, $N = 958, 1020, 960$ grid points is regarded as the reference solution. Horizontal diffusion is not used. The errors with respect to the reference solution are almost the same for the new DFS models, the old DFS model, and the SH model at the resolutions $j^0 = 64$ with around 128 × 64 grid points. Table 7 shows the normalized $L_2$ errors of the predicted height after a 15-day integration. The errors are almost the same between the old DFS, new DFS, and SH models at each resolution. The errors do not decrease when the resolution increases, which is different from the results in the other test cases. This may be because the mountain topography is not a differentiable function, and the mountain is added impulsively on to a initially balanced flow (Galewsky et al. 2004).

Figure 75 shows the longitudinal distributions of meridional wind at the grid points near the South Pole after a 15-day integration in the old and new DFS models using Grid [0] at the resolutions $j^0 = 64$ with 128 × 64 and $j^0 = 960, 1020, 960$ grid points. While the zonal wavenumber 1 component is dominant in the new DFS model at the resolutions $j^0 = 64$ and $N = 63128 × 64$ grid points, high zonal wavenumber noise appears in the old DFS model at the same resolution with 128 × 64 grid points. One possible reason is that the latitudinal derivative at the grid points can be unrealistically large in the old
DFS method even when $N' = J^3$ for even $m$ ($\geq 2$) as described in Sect. 4 (Fig. 3b). The difference is because the new DFS expansion method with the least-squares method improves numerical stability does not have this problem. By using the new expansion method with the least-squares method, the high zonal wavenumber noise does not appear even in the model that does not use the new DFS basis functions in Eq. (7) but uses the same DFS basis functions as in Eq. (117) except that the basis function for odd $m (\geq 3)$ is $\sin \theta \cos n \theta$ instead of $\sin^{n} \theta \sin n \theta$. The result of this model is almost the same as that of the new model (Figure not shown). In the old DFS model at high resolution with $J^0 = 960$ and $N = 959\times 1920 = 960$ grid points, the high wavenumber noise is not seen in Fig. 75. The higher the resolution, the smaller the high wavenumber noise becomes. Figure 86 shows the kinetic energy spectra of the horizontal winds (Lambert, 1984) after a 15-day integration in Williamson test case 5. The kinetic energy spectra in the DFS models are calculated from the SH expansion coefficients, which are obtained by firstly calculating the Gaussian grid-point values from the DFS coefficients using Eq. (108) for the new DFS method and Eq. (6) for the old DFS method, and secondly calculating the SH expansion coefficients from the Gaussian grid-point values by using a forward Legendre transform. In the old DFS model with $J^0 = 64$ and $N = 6322 \times 64$ grid points, the high wavenumber components are larger than in the other models, which is related to the high wavenumber noise near the South Pole in Fig. 75. In the old DFS model with $J^0 = 960$ and $N = 63 \times 1920$ grid points, the high wavenumber components are a little larger than in the other models, but the differences are slight.

Figure 97 shows the predicted height after a 15-day integration in Williamson test case 5, which is the same as Fig. 64 except for the truncation wavenumber $N' \approx 2J^3/3$. In our semi-implicit semi-Lagrangian models, we usually use $N$ satisfying $N' \approx J^3/3 - 1$ ($J$ is the number of longitudinal grid points), which is called linear truncation. However, here $N$ is determined to satisfy $N' \approx 2J^3/3$ to eliminate aliasing errors with quadratic nonlinearity (Orszag, 1971), which is called quadratic truncation. When using the quadratic truncation $N = 42$, the new DFS models with Grids [0], [1], and [−1] are stable without horizontal diffusion, but the old DFS model without strong high-order horizontal diffusion is unstable. The numerical instability in the old DFS model probably occurs because of the high-wavenumber oscillations due to the quadratic wavenumber truncation for even $m (\neq 0)$ (see, as explained in Sect. 4) as in the Williamson test case 1 with the Eulerian model. The results of the new DFS models are almost the same as those of the SH model. Table 8 is the same as Table 7 except for $N' \approx 2J^3/3$. The results of the new DFS models and the SH model with $N' \approx 2J^3/3$ in Table 8 are very similar to those with $N' \approx J^3/3 - 1$ in Table 7 when $J^0$ is the same. Figure 108 shows the kinetic energy spectrum of the horizontal winds after a 15-day integration in Williamson test case 5, which is the same as Fig. 86 except for the truncation wavenumber $N' \approx 2J^3/3$. At the resolution $J^0 = 64$ and $N = 42$ with $128 \times 64$ grid points, the high wavenumber components are a little larger in the SH model than in the new DFS model. At the resolution $J^0 = 960$ and $N = 639 \times 1920 = 960$ grid points, small oscillations appear in the high wavenumber region in the SH model, but not in the new DFS models. In the SH model, the wind components $u$ and $v$ divided by $\sin \theta$ are transformed from grid space to spectral space (Ritchie, 1988; Temperton, 1991), which seems to be the cause of the small oscillations in the high wavenumber region. Another way to transform $u$ and $v$ from grid space to spectral space in the SH model is to use the vector harmonic transform.
which avoids dividing $u$ and $v$ by $\sin \theta$ and improves the stability of the model (Swarztrauber, 2004). This approach is similar to the expansion method for $u$ and $v$ using the least-squares method in the new DFS method described in Sects. 3.6 and 2.8, and probably solves the problem of small oscillations with the high wavenumber components in the SH model. Alternatively, using $D$ and $\zeta$ instead of $u$ and $v$ as prognostic variables may eliminate the small oscillations.

## 6.6 Williamson test case 6

Figure 11 shows the predicted height after a 14-day integration in Williamson test case 6. The error is similar between the old and new DFS models using Grid [0] and the SH model. The error in the new DFS model using Grid [1] is the smallest. This is probably because Grid [1] has grid points at the poles, where the minimum height exists, and on the equator, where the maximum height exists. The error in the new DFS model using Grid [−1] is the second smallest. This is probably because Grid [−1] has grid points on the equator, where the maximum height exists. Table 9 shows the normalized $L^2$ errors of the predicted height after a 14-day integration. The errors in the old and new DFS models using Grid [0] and in the SH model are very close.

## 6.7.4 Galewsky test case

The Galewsky test case simulates a barotropically unstable mid-latitude jet. Figure 12 shows the predicted vorticity after a 6-day integration in the Galewsky test case for the models at 1.3 km resolution with $J^0 = 15360, 30720 \times 15360$ grid points and the quadratic truncation $N = 10239$, without horizontal diffusion. The result in the new DFS model using Grid [0] is almost the same as in the SH model. The old DFS model is unstable for the same reason as that shown in Sect. 6.5 (Fig. 9). Figure 13 shows the kinetic energy spectrum of horizontal winds after a 6-day integration in the Galewsky test case. The results are almost the same for the DFS models using Grid [0], [1] and [−1], and the SH model, but small oscillations appear near the truncation wavenumber in the SH model. This is probably for the same reason as in Williamson test case 5 in Fig. 10.

The results of the Galewsky-like test case using the north-south symmetric initial conditions are shown in Sect. S3 in the supplement.

## 5.5.8 Elapsed time

Figure 14 shows the elapsed time for the 15-day integration in the Williamson test case 5 in the SH model and the new DFS model using Grid [0] at 20 km resolution with $J^0 = 960, 1920 \times 960$ grid points and $N = 958$ (SH) or $N = 959$ (DFS), and that for the 6-day integration in the Galewsky test case at 1.3 km resolution with $J^0 = 15360, 30720 \times 15360$ grid points.
and \( N = 10239 \). We use one node (with two Intel Xeon Gold 6248 CPUs with 20 cores per CPU) of the FUJITSU Server PRIMERGY CX2550 M5 in the MRI. The source code written in Fortran is compiled with the Intel compiler. OpenMP parallelization is used, but MPI parallelization is not used. The elapsed time in the SH model is larger than in the DFS model, although the Legendre transform used in the SH model is highly optimized for Intel AVX512. The higher the resolution, the larger is the difference of the elapsed time between the models. This is because the Legendre transform used in the SH model requires \( O(N^3) \) operations while the Fourier cosine and sine transforms used in the DFS model require only \( O(N^2 \log N) \) operations. If the fast Legendre transform, which requires only \( N^2 (\log N)^2 \) operation, is used instead of the usual Legendre transform in the SH model, the difference of the elapsed time between the models will be reduced at high resolutions. We have not tested the fast Legendre transform yet because we do not have subroutines for the fast Legendre transform.

### Conclusions and perspectives

We have developed the new DFS method to improve the numerical stability of the DFS model, which has the following two improvements:

1. A new expansion method with the least-squares method is used to calculate the expansion coefficients so that the error due to the meridional wavenumber truncation is minimized. The method also avoids dividing by \( \sin \theta \) before taking the forward Fourier cosine or sine transform.

2. New DFS basis functions that guarantee that not only scalar variables, but also vector variables and the gradient of scalar variables, are continuous at the poles.

The equations obtained with the least-squares method are equivalent to those obtained with the Galerkin method. We also use the Galerkin method to solve partial differential equations such as the Poisson equation, the Helmholtz equation, and the shallow water equations.

To test the new DFS method, we conducted experiments for the Williamson test cases 2, 5 and 65, and the Galewsky test case in semi-implicit semi-Lagrangian shallow water models using the new DFS method with the three types of equally spaced latitudinal grids with or without the poles. We also ran the Williamson test case 1, which simulates a cosine-bell advection, in the Eulerian and semi-Lagrangian advection models. We compared the results between the new DFS models using the new DFS method and the old DFS model using the method of Yoshimura and Matsumura (2005) with the basis functions of Cheon (2000a, 2000b), and with the SH model.

The high zonal wavenumber noise of the meridional wind appears near the poles in the old DFS model, but not in the new DFS models in the Williamson test case 5. This is because the new DFS expansion method with the least-squares method improves the model’s stability. One possible reason is that the latitudinal derivative at the grid points can be unrealistically large in the old DFS method even when the truncation wavenumber \( N' \) for even \( m \neq 0 \) is equal to the number of latitudinal grid points \( J \), while the new DFS expansion method with the least-squares method does not have this problem. In the old DFS model, \( N' < J \) is a truncation wavenumber \( N' \) lower than the number of latitudinal grid points \( J \) for even \( m \neq 0 \).
causes numerical instability. In the new DFS model, an arbitrary meridional wavenumber truncation \( N < J \) can be used without the stability problem because the error due to meridional wavenumber truncation is small when using the new DFS expansion method with the least-squares method. This is one of the merits of the new DFS method because the quadratic truncation \( (N \equiv 2/3) \) or the cubic truncation \( (N \equiv J/2) \) is usually used in the Eulerian model and is also becoming to be used in the semi-Lagrangian model instead of the linear truncation \( (N \equiv J - 1) \) for stability and efficiency at high resolutions (Hotta and Ujiie, 2018; Dueben et al., 2020). We have also confirmed that in the new DFS model, stable integration is possible in all test cases shown here even without using the zonal Fourier filter unlike in the old DFS model. Thus, the numerical stability of the semi-implicit semi-Lagrangian model using the new DFS method is very good. In the Williamson test cases 1, the Eulerian advection model using the new DFS method also gives stable results without horizontal diffusion but with a zonal Fourier filter. The Eulerian advection model using the old DFS method is unstable without horizontal diffusion or with the weak fourth-order horizontal diffusion. In the old DFS model, the use of the semi-Lagrangian scheme is important for numerical stability. On the other hand, the advection model using the new DFS method is stable, even when the Eulerian scheme is used instead of the semi-Lagrangian scheme. The Eulerian shallow water model using the new DFS method without horizontal diffusion is also likely to be stable, although we have not tested it yet.

The results of the new DFS shallow water model are almost the same as the SH shallow water model. But in the SH shallow water model without horizontal diffusion, small oscillations appear in the high wavenumber region of the kinetic energy spectrum in some cases, unlike in the new DFS model. This seems to be because the wind components \( u \) and \( v \) divided by \( \sin \theta \) are transformed from grid space to spectral space in the SH model. The small oscillation problem with the SH model can probably be eliminated by using the vector harmonic transform, which is similar to the expansion method for \( u \) and \( v \) using the least-squares method in the new DFS model. Alternatively, using divergence and vorticity instead of \( u \) and \( v \) as prognostic variables may eliminate the small oscillations.

The elapsed time in the new DFS model is shorter than in the SH model especially at high resolution because the Fourier transform requires only \( O(N^2 \log N) \) operations, and the Legendre transform in the SH model requires \( O(N^3) \) operations. We have executed our shallow water models on Intel CPUs. The execution on GPUs is one important topic, but we have not tested our models on GPUs because the execution on GPUs is not an easy task. MPI parallelization is another important topic. However, in our shallow water models, we use only OpenMP parallelization, not MPI parallelization for the simplicity of the source code.

We developed hydrostatic and nonhydrostatic global atmospheric models using the old DFS method (Yoshimura and Matsumura, 2005; Yoshimura, 2012) and conducted typhoon prediction experiments in the nonhydrostatic global atmospheric model using the old DFS method in the Global 7 km mesh nonhydrostatic Model Intercomparison Project for improving TYphoon forecast (TYMIP-G7; Nakano et al., 2017). We have already developed a nonhydrostatic (or hydrostatic) atmospheric model using the new DFS method, where both OpenMP and MPI parallelization are used. We will describe the nonhydrostatic DFS model and the MPI parallelization, which will be described in another paper after improving the nonhydrostatic dynamical core as needed.
Supplement. The supplement related to this article is available online at...

Code availability. The source codes of the DFS and SH shallow water models are available in the supplement to the article and are licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) license. These models utilize the Netlib BIHAR library and the ISPACK library. The Netlib BIHAR library is available at https://www.netlib.org/bihar/ and is also included in the supplement. The ISPACK library is available at https://www.gfd-dennou.org/arch/ispack-3.0.1.tar.gz.

Data availability. The results of model experiments are available at https://climate.mri.jma.go.jp/prepub/archives/Yoshimura_DFS_SW_Testcase_2021/.

Appendix A: Trigonometric identities

We list here the trigonometric identities used in transforming the expressions in this paper.

The following identities are satisfied:

\[ \sin n\theta \cos n'\theta = \frac{1}{2} [\sin(n+n')\theta + \sin(n-n')\theta] \]  \hspace{1cm} (A1a)

\[ \cos n\theta \sin n'\theta = \frac{1}{2} [\sin(n+n')\theta - \sin(n-n')\theta] \]  \hspace{1cm} (A1b)

\[ \cos n\theta \cos n'\theta = \frac{1}{2} [\cos(n+n')\theta + \cos(n-n')\theta] \]  \hspace{1cm} (A1c)

\[ \sin n\theta \sin n'\theta = \frac{1}{2} [-\cos(n+n')\theta + \cos(n-n')\theta] \]  \hspace{1cm} (A1d)

From Eq. (A1), the following orthogonal relations in longitude are derived:

\[ \sin \theta \cos n\theta = \frac{1}{2} [\sin(n+1)\theta - \sin(n-1)\theta] \]  \hspace{1cm} (A2a)

\[ \sin \theta \sin n\theta = \frac{1}{2} [-\cos(n+1)\theta + \cos(n-1)\theta] \]  \hspace{1cm} (A2b)

\[ \sin^2 \theta \sin n\theta = \frac{1}{4} [-\sin(n-2)\theta + 2 \sin n\theta - \sin(n+2)\theta] \]  \hspace{1cm} (A2c)

\[ \sin^2 \theta \cos n\theta = \frac{1}{4} [-\cos(n-2)\theta + 2 \cos n\theta - \cos(n+2)\theta] \]  \hspace{1cm} (A2d)

\[ \sin^4 \theta \sin n\theta = \frac{1}{16} [\sin(n-4)\theta - 4 \sin(n-2)\theta + 6 \sin n\theta - 4 \sin(n+2)\theta + \sin(n+4)\theta] \]  \hspace{1cm} (A2e)

From Eq. (A1), the following orthogonal relations in longitude are derived:
Similarly, from Eq. (A1), the following orthogonal relations in latitude are derived:

\[ \int_0^{2\pi} \cos n\theta \cos n'\theta \, d\theta = \begin{cases} \frac{\pi}{2} & \text{for } n = n' = 0 \\ \frac{1}{2} \pi & \text{for } n = n' \neq 0 \\ 0 & \text{for } n \neq n' \end{cases} \quad (A4a) \]

\[ \int_0^{2\pi} \sin n\theta \sin n'\theta \, d\theta = \begin{cases} \frac{1}{2} \pi & \text{for } n = n' = 0 \\ 0 & \text{for } n \neq n' \end{cases} \quad (A4b) \]

By using Eq. (A1), the following relations are derived:

\[
\sin \theta \frac{\partial}{\partial \theta} (\sin^1 \theta \cos n\theta) = \frac{n + l}{4} \sin^{l+1} \theta \cos (n + 1)\theta - \frac{n - l}{4} \sin^{l-1} \theta \cos (n - 1)\theta
\]

\[
\sin \theta \frac{\partial}{\partial \theta} (\sin^1 \theta \sin n\theta) = \frac{n + l}{4} \sin^{l+1} \theta \sin (n + 1)\theta - \frac{n - l}{4} \sin^{l-1} \theta \sin (n - 1)\theta
\]

\[
\sin \theta \frac{\partial}{\partial \theta} (\sin n\theta \cos n\theta) = \frac{(n + l)(n + l + 1)}{4} \sin^l \theta \cos (n + 2)\theta - \frac{(n - l)(n - l - 1)}{4} \sin^l \theta \cos (n - 2)\theta
\]

\[
\sin \theta \frac{\partial}{\partial \theta} (\sin n\theta \sin n\theta) = \frac{(n + l)(n + l + 1)}{4} \sin^l \theta \sin (n + 2)\theta - \frac{(n - l)(n - l - 1)}{4} \sin^l \theta \sin (n - 2)\theta
\]

\[
\frac{\partial}{\partial \theta} (\sin^2 \theta \sin n\theta) = -\frac{n - 2}{4} \cos (n - 2)\theta + \frac{2n}{4} \cos n\theta - \frac{n + 2}{4} \cos (n + 2)\theta
\]

Appendix B: Discrete Fourier cosine and sine transforms in latitude

Forward discrete Fourier cosine and sine transforms are performed in Eqs. (2319) and (573), and inverse discrete Fourier cosine and sine transforms are performed in Eqs. (139), (5247) and (48), in the latitudinal direction.
calculation of the discrete cosine and sine transforms in Grids [0], [1], and [−1] is shown below. Here, \( g(\theta_j) \) and \( h(\theta_j) \) are grid-point values, and \( g_n \) and \( h_n \) are expansion coefficients.

When using Grid [0], forward and inverse and forward discrete cosine transforms are performed as

\[
g(\theta_j) = \sum_{n=0}^{p-1} g_n \cos n\theta_j. \tag{B1a} \\
g_n = \frac{1}{p} \sum_{j=0}^{p-1} g(\theta_j) \cos n\theta_j, \quad b \equiv \begin{cases} 1 & \text{for } n = 0 \\ 2 & \text{for } n \neq 0. \end{cases} \tag{B1b} 
\]

When using Grid [0], forward and inverse and forward discrete sine transforms are performed as

\[
h(\theta_j) = \sum_{n=1}^{p} h_n \sin n\theta_j. \tag{B2a} \\
h_n = \frac{1}{p} \sum_{j=0}^{p-1} h(\theta_j) \sin n\theta_j, \quad b \equiv \begin{cases} 1 & \text{for } n = 0 \\ 2 & \text{for } n \neq 0. \end{cases} \tag{B2b} 
\]

When using Grid [1], forward and inverse and forward discrete cosine transforms are performed as

\[
g(\theta_j) = \sum_{n=0}^{\rho} g_n \cos n\theta_j. \tag{B3a} \\
g_n = \frac{1}{\rho} \sum_{j=0}^{\rho} g(\theta_j) \cos n\theta_j, \quad b \equiv \begin{cases} 1 & \text{for } n = 0, \rho \\ 2 & \text{for } 0 < n < \rho. \end{cases} \tag{B3b} 
\]

When using Grid [1], forward and inverse and forward discrete sine transforms are performed as

\[
h(\theta_j) = \sum_{n=1}^{\rho} h_n \sin n\theta_j, \quad h(\theta_0) = h(\theta_\rho) = 0. \tag{B4a} \\
h_n = \frac{2}{\rho} \sum_{j=0}^{\rho-1} h(\theta_j) \sin n\theta_j, \quad b \equiv \begin{cases} 1/2 & \text{for } 0 < j < \rho. \end{cases} \tag{B4b} 
\]
Grid $-1$ is the same as Grid $1$, except that there are no grid points at the North and South poles. The zonal wavenumber components of scalar variables at the poles are zero except for $m = 0$ (See Eq. (108)), and those of vector variables at the poles are zero except for $m = 1$ (See Eqs. (5247) and (48)). When we use Grid $-1$ and the values at the poles are known to be zero, forward and inverse discrete cosine transforms can be performed using Eq. (B3) and forward and inverse discrete sine transforms can be performed using Eq. (B4) in the same way as for Grid $1$. When we use Grid $-1$ and the values at the poles are unknown (i.e., the zonal wavenumber components of scalar variables for $m = 0$, and those of vector variables for $m = 1$), the inverse discrete cosine transform can be performed like Eq. (B3b)

$$g(\theta_j) = \sum_{n=0}^{J-2} g_n \cos n\theta_j.$$  

(55)

where $n$ is from $0$ to $J-2 = (f-1)$ because the number of the meridional grid points is $f-1 = (f-1)$ in Grid $-1$ (See Eq. (B3b)). However, the forward discrete cosine transform cannot be performed like using Eq. (B3a). We can calculate the expansion coefficients $g_n$ from $g(\theta_j)$ in the following way. Eq. (B55b) is multiplied by $\sin \theta_j$, and we define $\tilde{g}(\theta_j)$ as

$$\tilde{g}(\theta_j) \equiv g(\theta_j) \sin \theta_j = \sum_{n=0}^{J-2} g_n \sin \theta_j \cos n\theta_j.$$  

(56)

Since the values at the poles are zero, we can expand $\tilde{g}(\theta_j)$ as

$$\tilde{g}(\theta_j) \equiv g(\theta_j) \sin \theta_j = \sum_{n=1}^{J-1} \tilde{g}_n \sin n\theta_j.$$  

(76)

The expansion coefficients $\tilde{g}_n$ can be obtained from $\tilde{g}(\theta_j)$ in the same way as in Eq. (B4b) by forward discrete sine transform:

$$\tilde{g}_n = \frac{2}{J-1} \sum_{j=1}^{J-1} \tilde{g}(\theta_j) \sin n\theta_j.$$  

(82)

From Eqs. (B25) and (B76), we obtain

$$\sum_{n=0}^{J-2} g_n \sin \theta \cos n\theta = \sum_{n=1}^{J-1} \tilde{g}_n \sin n\theta_j.$$  

(98a)

By using Eq. (A2a), we obtain

$$\sum_{n=0}^{J-2} g_n \sin \theta \cos n\theta = \left( g_0 - \frac{B_1}{2} \sin \theta \right) + \sum_{n=1}^{J-1} \left( \frac{B_{n-1}}{2} - \frac{g_{n+1} - g_n}{2} \right) \sin n\theta + \frac{B_{J-2} \sin (J-1)\theta - 2 B_{J-2} \sin (J-1)\theta}{2} - \frac{B_{J-2} \sin (J-1)\theta}{2}.$$  

(108)

By substituting Eq. (B109) into Eq. (B98a) and comparing the left and right sides of the equation, we obtain
We can calculate $\phi(\theta_i)$ from $\eta(\theta_i)$ using Eq. (B6), calculate $\tilde{\phi}$ from $\phi(\theta_i)$ using Eq. (B28), and calculate $\phi_n$ from $\tilde{\phi}_n$ using Eq. (B110).

**Appendix C: The upper limit of the meridional truncation wavenumber $N$**

In the new DFS method, the meridional truncation wavenumber $N$ is used for the new DFS meridional basis functions in Eqs. (12), and for the discrete cosine or sine transform of a scalar variable (Eqs. (13) and (23)), derivatives of a scalar variable (Eqs. 18 and 20) and a wind vector (Eqs. 52 and 57). In Grid [0], the upper limit of $N$ is $J^o - 1$ for each $m$ because the discrete cosine transform in Eq. (B1), where the maximum value of $n$ is $f^o - 1$, is used for a scalar variable when $m$ is even, and for vector components when $m$ is odd. In Grid [1], the upper limit of $N$ is $J^o - 1$ for each $m$ because the discrete sine transform in Eq. (B4), where the maximum value of $n$ is $f^o - 1$, is used for a scalar variable when $m$ is odd, and for vector components when $m$ is even. In Grid [−1], the upper limit of $N$ is $f^o - 1$ for $m \geq 2$ because of the same reason as in Grid [1]. However, for $m = 0$ or 1 in Grid [−1], the upper limit of $N$ is $f^o - 2$ because the discrete cosine transform in Eq. (B5), where the maximum value of $n$ is $f^o - 2$, is used for a scalar variable when $m = 0$, and for vector components when $m = 1$. Thus, the upper limit of $N$ is $J^o - 1$, except that the upper limit of $N$ for $m = 0$ or 1 in Grid [−1] is $J^o - 2$. For example, in the model using the new DFS method with Grid [−1] at the resolution $J^o = 64$ and $N = 63$, we set $N = 63$ for $m \geq 2$ but $N = 62$ for $m = 0$ or 1.

**Appendix D: Equations for the derivation of Eqs. (29) and (62)**

$\tilde{T}_{n,m}^{(X)}$ in Eq. (23) is calculated by the forward Fourier cosine or sine transform as

$$
\tilde{T}_{n,m}^{(X)} = \frac{\pi}{b} \int_0^{\pi} \tilde{T}_{n}^{(X)}(\theta) \cos n \theta \, d\theta \quad b \equiv \begin{cases} 
1 & \text{for } n = 0, \\
2 & \text{for } n \neq 0, \text{ for even } m, \\
2 & \text{for } n \neq 0, \text{ for odd } m,
\end{cases}
$$

The equations for the forward discrete cosine or sine transform are described in Appendix B. From Eq. (23) and (A5),

$$
\frac{b}{\pi} \int_0^{\pi} \cos n \theta \, \tilde{T}_{m}^{(X)}(\theta) \, d\theta = \tilde{T}_{m,n}^{(X)} \quad (n = 0, \ldots, N) \text{ for even } m,
$$

(D2a)
\[ \frac{\pi}{2} \int_0^\pi \sin n\theta \, T_n^m(\theta) \, d\theta = \int_0^\pi \cos n\theta \, T_n^m(\theta) \, d\theta \quad (n = 1, \ldots, N) \text{ for odd } m \] (D2b)

are also derived. From Eqs. (D1) and (D2),

\[ \int_0^\pi \cos n\theta \, T_n^m(\theta) \, d\theta = \int_0^\pi \sin n\theta \, T_n^m(\theta) \, d\theta \quad (n = 0, \ldots, N) \text{ for even } m \] (D3a)

\[ \int_0^\pi \sin n\theta \, T_n^m(\theta) \, d\theta = \int_0^\pi \cos n\theta \, T_n^m(\theta) \, d\theta \quad (n = 1, \ldots, N) \text{ for odd } m \] (D3b)

are satisfied. From Eqs. (D3), (11) and (A2a–c), we derive

\[ \int_0^\pi S_n(\theta) T_n^m(\theta) \, d\theta = \int_0^\pi S_n(\theta) T_n^m(\theta) \, d\theta. \] (D4)*

From Eqs. (28) and (D4), we derive Eq. (29).

We can also derive the following equations from Eq. (57) in the similar way to the derivation of (D3):

\[ \int_0^\pi \sin n\theta \, u_n^m(\theta) \, d\theta = \int_0^\pi \sin n\theta \, u_n^m(\theta) \, d\theta \quad (n = 1, \ldots, N) \text{ for even } m, \] (D5a)

\[ \int_0^\pi \cos n\theta \, u_n^m(\theta) \, d\theta = \int_0^\pi \cos n\theta \, u_n^m(\theta) \, d\theta \quad (n = 0, \ldots, N) \text{ for odd } m \] (D5b)

From Eqs. (D5), (11), and (A2a–c), we derive

\[ \int_0^\pi \frac{m S_n(\theta)}{\sin \theta} \, S_n^m(\theta) \, d\theta = \int_0^\pi \frac{m S_n(\theta)}{\sin \theta} \, S_n^m(\theta) \, d\theta, \] (D6a)

\[ \int_0^\pi \frac{\partial S_n(\theta)}{\partial \theta} \, S_n^m(\theta) \, d\theta = \int_0^\pi \frac{\partial S_n(\theta)}{\partial \theta} \, S_n^m(\theta) \, d\theta. \] (D6b)

We can also derive the same equations as Eq. (D6) except that \( u \) is replaced with \( v \). Equation (D6) are used to derive Eq. (62).

Here we derive Eq. (102a) for odd \( m \geq 3 \) from Eq. (29). Eq. (102b,c,d) can be derived similarly.

By using Eqs. (112a) and (A2c,e), the following equations are derived.

(L.h.s. of Eq. (29) for odd \( m \geq 3 \) = \[ \int_0^\pi S_n(\theta) T_n^m(\theta) \, d\theta = \int_0^\pi \sin \theta \, \sin n\theta \sum_{n'=1}^{N-2} T_n^{m,n'} \sin \theta \, \sin n' \theta \, d\theta \]

= \[ \frac{\pi}{2} \int_0^\pi \sum_{n'=1}^{N-2} T_n^{m,n'} \sin(n'-4)\theta - 4\sin(n'-2)\theta + 6\sin(n'-4)\theta - 4\sin(n'-4)\theta \, d\theta \]

= \[ \frac{\pi}{2} \int_0^\pi \sum_{n'=1}^{N-2} T_n^{m,n'} \sin \theta \, \sin \theta \, \sum_{n'=1}^{N-2} T_n^{m,n'} \sin \theta \, \sin \theta \, d\theta \]

= \[ \frac{\pi}{2} \int_0^\pi \sum_{n'=1}^{N-2} \frac{10T_{m,n'} - 7T_{m,n'} + 16T_{m,n'} \sin \theta + 5T_{m,n'} - 4T_{m,n'} + 16T_{m,n'} \sin \theta + 5T_{m,n'} - 4T_{m,n'} + 16T_{m,n'} \sin \theta}{16} \sin \theta \, d\theta \]

= \[ \frac{\pi}{2} \int_0^\pi \sum_{n'=1}^{N-2} \frac{10T_{m,n'} - 7T_{m,n'} + 16T_{m,n'} \sin \theta + 5T_{m,n'} - 4T_{m,n'} + 16T_{m,n'} \sin \theta + 5T_{m,n'} - 4T_{m,n'} + 16T_{m,n'} \sin \theta}{16} \sin \theta \, d\theta \]

52
From Eqs. (29), (EC1), (EC2) and (A4b), Eq. (59a) are derived.

Appendix F: Zonal Fourier filter

2.11 Zonal Fourier filter

In a regular longitude–latitude grids, the longitudinal grid spacing becomes narrow at high latitudes. In DFS methods, the zonal Fourier filter (Merilees 1974; Boer and Steinberg 1975; Cheong 2000a), which filters out the high zonal wavenumber components at high latitudes, is usually used to obtain a more uniform resolution. The use of a reduced grid (Hortal and Simmons, 1991; Juang, 2004; Miyamoto, 2006; Malardel, 2016) has a similar effect to the zonal Fourier filter. In our atmospheric model using the old DFS method (Yoshimura and Matsumura, 2005; Yoshimura, 2012), we use the reduced grid of Miyamoto (2006).

In this study, we use the longitude–latitude grid with the zonal Fourier filter, not the reduced grid, for the simplicity of the source code. We set the largest zonal wavenumber $M_i$ at each latitude as

$$M_i(\theta_i) = \min[M, 60 + M \sin(\theta_i)],$$

where we use the value $M = 20$ in the DFS shallow water models to make the resolution similar to that in the reduced grid of Miyamoto (2006). The values of $T_{n1}(\theta)$ and $T_{n2}(\theta)$ in Eq. (62) are set to zero for $m > M_i(\theta_i)$ during the spectral transform. We use the value $M = 1$ in the DFS Eulerian advection model, we use the value $M = 1$ as described in Sect. 6.3. The use of a reduced grid (Hortal and Simmons, 1991; Juang, 2004; Miyamoto, 2006) has a similar effect to the zonal Fourier filter.
Appendix GB: Calculation of global mean and latitudinal area weight

The global mean value of $T_{\delta,0}(\lambda, \theta)$ in Eq. (1544a) can be calculated in spectral space by the following equation (Cheong 2000a):

$$G = \frac{1}{4\pi} \int_0^{2\pi} \int_0^{\pi} \left( \sum_{n=0}^{N} C_n(\theta) \cos m \lambda + \sum_{m=1}^{M} S_m(\theta) \sin m \lambda \right) \sin \theta \, d\lambda \, d\theta$$

$$= \frac{1}{2} \int_0^{\pi} \sum_{n=0}^{N} T_{n,n=0} \cos n \theta \, n \, d\theta = \sum_{n=0}^{N} T_{n,n=0} \frac{1}{1-n^2}.$$ (G11)

The latitudinal area weight at each latitude $\theta_j$ is calculated as follows:

1. The latitudinal distribution of $T_{n,n=0}(\theta_j)$ for each $j$ is given as

$$T_{n,n=0}(\theta_j) = \begin{cases} 1 & \text{for } k = j \\ 0 & \text{for } k \neq j \end{cases} \quad (0 \leq k \leq J-1).$$ (G12)

2. From $T_{n,n=0}(\theta_j)$, the meridional expansion coefficients $T_{n,j}(\theta)$, $0 \leq n \leq N$, are calculated by forward discrete cosine transform described in Sect. 2.10 Appendix B.

3. The value of $G$ calculated from $T_{n,n=0}(\theta_j)$ using Eq. (G11) is considered as the latitudinal area weight $w(\theta_j)$ at latitude $\theta_j$.

In Grid 00 and Grid 11, the distribution of $w(\theta_j)$ is smooth. However, in Grid 1−1, the distribution of $w(\theta_j)$ is not smooth because of the irregularity with Grid 1−1 (See Eqs. (B5)−(B11) in Appendix B).

The latitudinal area weight $w(\theta_j)$ is used, for example, to calculate the global mean in the grid space.

Appendix H: Derivation of Eq. (63) from Eq. (62)

Here we describe the derivation of Eq. (63d) for odd $m (\geq 3)$ from Eq. (62a). Eq. (63b,c) can be derived similarly.

From Eqs. (62a), (11), (52), (57), (A2b) and (A5e), and from the same equations as Eqs. (53b,c,d) except that $u_{\delta,0}^m$, $v_{\delta,0}^{m,0}$, and $\psi_{\delta,0}^{m,0}$, respectively, and the same equations as Eqs. (53a,b,c,d) except that $u_{\delta,0}^m$, $x_{\delta,0}^m$, and $\psi_{\delta,0}^m$, respectively, we derive the following equation for odd $m \geq 3$:

$$\begin{align*}
\frac{1}{4} \int_0^{\pi} \left( -m \cos(n+1)\varphi + \cos(n-1)\varphi \right) & \frac{1}{2} \left[ -m x_{\delta,0}^m - 3 x_{\delta,0}^m \cos \theta + 3 \psi_{\delta,0}^m - \psi_{\delta,0}^m - \psi_{\delta,0}^m \cos \theta \right] \\
& + \sum_{n'=1}^{N} \left[ m x_{\delta,0}^{n'-1,m} - x_{\delta,0}^{n'+1,m} + m \psi_{\delta,0}^{n'-2,m} + 2 \psi_{\delta,0}^{n'-2,m} \right] \cos \theta \\
& - \left( -n - 2 \cos(n-2)\varphi + 2 \cos \theta \right) \frac{n + 2}{4} \cos \theta + \frac{n + 2}{4} \cos(n+2)\varphi \right) \frac{1}{4} \left[ m x_{\delta,0}^m - \psi_{\delta,0}^m + \frac{m \psi_{\delta,0}^m}{2} - \psi_{\delta,0}^m + \frac{3 \psi_{\delta,0}^m}{4} + x_{\delta,0}^m \cos \theta \right].
\end{align*}$$ (H13)
When \( n \geq 4 \), by using Eq. (A4a), Eq. (H1) can be converted into

\[
\int_0^\infty \left( \frac{m}{2a} \cos(n+1)\theta \right) \left( \frac{m(x_{n-2,m} - x_{n+2,m})}{2} + \frac{(n+1)(-\psi_{n-3,m} - 2\psi_{n+1,m} - \psi_{n+3,m})}{4} - \tilde{u}_{n+1,m}^n \right) \cos(n+1)\theta d\theta = 0 \quad \text{(H1)}
\]

\[
\int_0^\infty \left( \frac{m}{2a} \cos(n-1)\theta \right) \left( \frac{m(x_{n-2,m} - x_{n+2,m})}{2} + \frac{(n-1)(-\psi_{n-3,m} - 2\psi_{n+1,m} - \psi_{n+3,m})}{4} - \tilde{u}_{n-1,m}^{n-1} \right) \cos(n-1)\theta d\theta = 0 \quad \text{(H2)}
\]

From Eq. (H2) and (A4a), Eq. (63d) for \( n \geq 4 \) is derived. Equation (63d) for \( n \leq 3 \) can also be derived from (H1) and (A4a).

Appendix I: 4.2 Two-time-level semi-implicit semi-Lagrangian scheme for Time integration method

A two-time-level semi-implicit semi-Lagrangian scheme (e.g., Temerton et al., 2001) and the Stable Extrapolation Two-Time-Level Scheme (SETTLS; Hortal, 2002) are adopted to discretize the shallow water equations in Eqs. (98) and (99) in time as

\[
\begin{align*}
(v + 2D \times r)^+ - (v + D \times r)^0 &= \frac{\beta_x (\nabla h^+ + \nabla h^0)}{2} - \beta_x (\nabla h^+ + \nabla h^0) + \beta_x (\nabla h^0 + \nabla h^+)^0 \quad \text{(I.G1)}
\end{align*}
\]

\[
\frac{h^+ - h^0}{\Delta t} = \frac{1}{2} \left[ \frac{[h - h_x]^0 + [h - h_x]^+}{2} + \frac{[v \cdot \nabla h_x]^0 + [v \cdot \nabla h_x]^+}{2} \right] + \beta_x \frac{[hD h^+ + hD h^0]}{2} - \beta_x \frac{[hD h^0 + hD h^+]^0}{2} \quad \text{(I.G2)}
\]

where

\[
D \equiv \nabla \cdot v = \frac{1}{a \cos \phi} \frac{\partial u}{\partial \lambda} + \frac{1}{a \cos \phi} \frac{\partial v}{\partial \phi}
\]

is horizontal divergence; \( \Delta t \) is a timestep; the superscripts \(-, 0, +\) mean past time \((t - \Delta t)\), present time \((t)\), and future time \((t + \Delta t)\), respectively, and the superscript \((+)^0\) means future time \((t + \Delta t)\), extrapolated in time, for example, \(h^{(+)^0} = \)}
$2h^3 - h^2$ the subscript D means the departure point, and the absence of the subscript D means the arrival point; $\bar{h}$ is a constant value of height for semi-implicit linear terms; $\beta_v$ and $\beta_{\omega}$ are second-order decentering parameters (Yukimoto et al., 2011). Using $\beta_v$ and $\beta_{\omega}$ larger than 1.0 (e.g., 1.2) increases the effect of the semi-implicit scheme improving computational stability, but $\beta_v = \beta_{\omega} = 1.0$ is used here because $\bar{h}$ larger than $\bar{h}$ is enough for stable calculations in the shallow water model.

The departure point $x_\|_{\pm}$ is the upstream horizontal position from the arrival point $\psi$ along the wind vector between present time ($t$) and future time ($t + \Delta t$). Here, the arrival point $\psi$ is on a grid point, and the departure point $x_\|$ is not generally on a grid point. Since the right-hand sides of Eqs. (IG1) and (IG2) are the time average between present time ($t$) and future time ($t + \Delta t$) and the spatial average between the departure point and the arrival point, these equations have second-order precision in time and space. In SETTLS, $x_\|$ is calculated using

$$x_\| = x - \frac{\nu_0^{(s)} + \nu_0^{(0)}}{2} \Delta t. \quad (\text{IG4})$$

However, when $\Delta t$ is longer than 30 minutes, using $\nu_0^{(s)}$ extrapolated in time to calculate $x_\|$ causes numerical instability in our experiments. To avoid instability when $\Delta t$ is 1 hour, here we use

$$x_\| = x - \frac{\nu_0^{(s)} + \nu_0^{(s)}}{2} \Delta t, \quad (\text{IG5a})$$

$$\nu_0^{(s)} \equiv \nu_0^{(s)} + \frac{(2 \Omega \times r_{10}^{(s)} - 2 \Omega \times r - g [\psi h_0^{(s)} + \psi h_0^{(s)}]}{2} \Delta t, \quad (\text{IG5b})$$

instead of Eq. (IG4), where $\nu_0^{(s)}$ is a provisional future value obtained by discretizing Eq. (98106) in an explicit semi-Lagrangian scheme. From Eq. (IG5), we obtain

$$x_\| = x - \Delta t \left( \left( \nu^{(s)} + \frac{2 \Omega \times r - g (\psi h_0^{(s)} + \psi h_0^{(s)})}{4} \right) - \frac{2 \Omega \times r - g \Delta t \psi h_0^{(s)}}{4} \right), \quad (\text{IG6})$$

This method using a provisional future value to calculate $x_\|$ is similar to the method in Gospodinov et al., (2001). Since the value with the subscript D depends on $x_\|$, $x_\|$ is calculated iteratively from Eq. (IG6) (e.g., Ritchie, 1995; Temperton et al., 2001). Since $x_\|$ is not generally on the grid point, the value at $x_\|$ is calculated by spatial interpolation from nearby grid points. In the right-hand side of Eq. (IG6), the value at $x_\|$ with the subscript D is calculated by third-order Lagrange interpolation.

Eqs. (IG1) and (IG2) are transformed into

$$v^{(s)} + \frac{\beta_v \Delta t}{2} g \psi h^{(s)} = R_v, \quad (\text{IG7a})$$

$$R_v \equiv \left( \frac{\nu^{(s)} + 2 \Omega \times r - \frac{\Delta t}{2} g (\psi h^{(s)} - \beta_v \psi h^{(s)} + \beta_{\omega} \psi h^{(s)})}{4} \right) - \frac{2 \Omega \times r - \frac{\Delta t}{2} g (\psi h^{(s)} + \beta_v \psi h^{(s)})}{4} \Delta t, \quad (\text{IG7b})$$

$$h^{(s)} + \frac{\beta_{\omega} \Delta t}{2} h D^{(s)} = R_h, \quad (\text{IG8a})$$

$$R_h \equiv h^{(s)} + \frac{\Delta t}{2} \left[ \left( \frac{\nu^{(s)} + 2 \Omega \times r - \frac{\Delta t}{2} g (\psi h^{(s)} - \beta_v \psi h^{(s)} + \beta_{\omega} \psi h^{(s)})}{4} \right) - \frac{2 \Omega \times r - \frac{\Delta t}{2} g (\psi h^{(s)} + \beta_v \psi h^{(s)})}{4} \right] \Delta t, \quad (\text{IG8b})$$
In Eqs. (IG7b) and (IG8b), the values at \(x_D\) with the subscript \(D\) are calculated by fifth-order and third-order Lagrange interpolations, respectively, since high-order interpolation of wind vector components increases the accuracy of the model’s results in our experiments. From Eq. (IG7a), we obtain

\[
D^+ + \frac{\beta \Delta t}{2} g \nabla^2 h^+ = R_\nu, \\
\zeta^+ = R_\zeta,
\]

where

\[
\zeta \equiv k \cdot \nabla \times v = \frac{1}{a} \left( \frac{\partial v}{\cos \phi} - \frac{\partial u \cos \phi}{\partial \phi} \right)
\]

is vorticity, \(k \equiv r/|r|\) is the vertical unit vector, \(R_\nu \equiv \nabla \cdot R_v\) and \(R_\zeta \equiv k \cdot \nabla \times R_v\).

We calculate \(h^+\) and \(v^+\) using the spectral transform method and the Galerkin method with the new DFS method as follows.

1. The scalar variable \(R_\nu\) is transformed from grid space to spectral space using Eqs. (2319)–(3125). The components of the vector variable \(R_v = (R_x, R_y)\) in grid space are transformed to \(R_\zeta\) and \(R_\phi\) in spectral space using Eqs. (551)–(642), where \(R_\zeta\) and \(R_\phi\) are the velocity potential and the stream function of \(R_v\), respectively.

2. \(R_\zeta\) and \(R_\phi\) are calculated by

\[
R_\zeta = \nabla^2 R_\phi, \quad R_\phi = \nabla R_\zeta,
\]

using Eqs. (7589) and (7591), \(\zeta^+\) is obtained from \(R_\zeta\) using Eq. (IG10).

3. Equations (IG8a) and (IG12) are substituted into Eq. (IG9) and we obtain

\[
D^+ - \frac{\Delta t^2}{2} \beta \beta \delta g \nabla^2 D^+ = \nabla^2 \left( R_x - \frac{\Delta t}{2} \beta \beta \delta R_\nu \right).
\]

4. \(h^+\) is calculated from \(D^+\) and \(R_\nu\) using Eq. (IG8).

5. \(v^+\) and \(\zeta^+\) are calculated from \(D^+\) and \(\zeta^+\) by solving the Poisson equations

\[
\nabla^2 v^+ = D^+, \\
\nabla^2 \zeta^+ = \chi^+,
\]

using Eqs. (7589) and (7592).

6. \(v^+ = (u^+, v^+)\) is calculated from \(v^+\) and \(\zeta^+\) using Eq. (5349) for \(u^m\) and \(v^m\) and the similar equations for \(u^l\) and \(v^l\).

7. \(u^+, v^+, \phi^+ \), \(h^+, D^+\), and \(\nabla h^+\) in spectral space are transformed to grid space; \(h^+\) and \(D^+\) are transformed meridionally using Eqs. (138a) and (138b), \(u^+\), and \(v^+\) are transformed meridionally using Eq. (5348), \(h^+\) is transformed meridionally using Eqs. (134a)–(134d). \(h^+\) can also be calculated from \(h^m_n(\theta)\) and \(h^m_n(\theta)\) at the latitudinal grid.
points using Eq. (16) and additionally using Eq. (22) at the poles when using Grid [1], which is more efficient than using Eqs. (18) and (19) because the meridional inverse discrete cosine and sine transforms of $h^*$ become unnecessary.

Author Contributions.

HY developed a new DFS method and a shallow water model using the method, conducted model experiments, analysed data, and wrote the paper.

Competing Interests.

The author declares that there is no conflict of interest.

Acknowledgements.

We are grateful to Keiichi Ishioka (Kyoto University), Tadashi Tsuyuki (MRI), Daisuke Hotta (MRI), Masashi Ujiie (JMA), and other members of the MRI and JMA model development teams for their useful comments. We are also grateful to the referees for their comments that help to improve the paper quality. This work was supported by the Integrated Research Program for Advancing Climate Models (TOUGOU) Grant Number JPMX0717935561 from the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan.

References


Table 1. Normalized $L_2$ errors of Laplacian operator calculation ($\nabla^2 f$). We use the old DFS method with Grid [0], the new DFS methods with Grid [0], Grid [1] and Grid [-1], and the SH method. $J^0$ is the number of latitudinal grid points in Grid [0]. The truncation wavenumber $N \approx 2J^0/3$.

<table>
<thead>
<tr>
<th>Resolution Method</th>
<th>Old DFS [0]</th>
<th>DFS [0]</th>
<th>DFS [1]</th>
<th>DFS [-1]</th>
<th>SH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J^0=64$, $N=42$</td>
<td>4.1208E−3</td>
<td>2.3019E−3</td>
<td>2.2530E−3</td>
<td>2.6281E−3</td>
<td>2.0927E−3</td>
</tr>
<tr>
<td>$J^0=160$, $N=106$</td>
<td>2.2221E−4</td>
<td>2.3678E−4</td>
<td>2.3369E−4</td>
<td>2.3344E−4</td>
<td>2.1668E−4</td>
</tr>
<tr>
<td>$J^0=320$, $N=213$</td>
<td>3.8070E−5</td>
<td>3.7931E−5</td>
<td>3.8752E−5</td>
<td>3.8740E−5</td>
<td>3.7565E−5</td>
</tr>
<tr>
<td>$J^0=960$, $N=639$</td>
<td>2.4281E−6</td>
<td>3.5687E−6</td>
<td>3.5888E−6</td>
<td>3.5804E−6</td>
<td>2.3451E−6</td>
</tr>
</tbody>
</table>
Table 2. Same as Table 1 except that the global mean values of calculated $\nabla^2 f$ are shown.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$J=64, N=42$</td>
<td>3.4331E−26</td>
<td>2.0012E−26</td>
<td>1.2222E−25</td>
<td>6.4414E−25</td>
<td>1.8359E−27</td>
</tr>
</tbody>
</table>
Table 3. Same as Table 1 except that $L_2$ errors of the solution of the Helmholtz equation are shown.

<table>
<thead>
<tr>
<th>Resolution Method</th>
<th>Old DFS [0]</th>
<th>DFS [0]</th>
<th>DFS [1]</th>
<th>DFS [-1]</th>
<th>SH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J^0=64, N=42$</td>
<td>$7.5000E-4$</td>
<td>$7.0729E-4$</td>
<td>$7.3360E-4$</td>
<td>$7.5868E-4$</td>
<td>$6.4564E-4$</td>
</tr>
<tr>
<td>$J^0=160, N=106$</td>
<td>$1.7270E-5$</td>
<td>$1.7263E-5$</td>
<td>$1.5847E-5$</td>
<td>$1.5807E-5$</td>
<td>$1.0100E-5$</td>
</tr>
<tr>
<td>$J^0=220, N=213$</td>
<td>$1.0970E-6$</td>
<td>$1.0968E-6$</td>
<td>$1.2557E-6$</td>
<td>$1.2602E-6$</td>
<td>$7.3458E-6$</td>
</tr>
</tbody>
</table>
Table 4. Normalized $L_2$ errors of the predicted height after a 12-day integration in the Williamson test case 1 when using the Eulerian advection models. The truncation wavenumber $N \approx 2J/3$.

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Model</th>
<th>Old DFS [0]</th>
<th>DFS [0]</th>
<th>DFS [1]</th>
<th>DFS [-1]</th>
<th>SH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J=64, N=42$</td>
<td>Unstable</td>
<td>1.1557E+1</td>
<td>1.1559E+1</td>
<td>1.1559E+1</td>
<td>1.1554E+1</td>
<td></td>
</tr>
<tr>
<td>$J=320, N=213$</td>
<td>Unstable</td>
<td>2.4619E-2</td>
<td>2.4619E-2</td>
<td>2.4619E-2</td>
<td>2.4619E-2</td>
<td></td>
</tr>
<tr>
<td>$J=960, N=639$</td>
<td>Unstable</td>
<td>8.2424E-3</td>
<td>8.2424E-3</td>
<td>8.2424E-3</td>
<td>8.2424E-3</td>
<td></td>
</tr>
</tbody>
</table>
Table 5. Same as Table 4 except for using the semi-Lagrangian models and the truncation wavenumber $N = f^2 - 1$.

<table>
<thead>
<tr>
<th>Resolution - Model</th>
<th>Old DFS [0]</th>
<th>DFS [0]</th>
<th>DFS [1]</th>
<th>DFS [-1]</th>
<th>SH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J=64, N=63$</td>
<td>1.6782E-1</td>
<td>1.6782E-1</td>
<td>1.6795E-1</td>
<td>1.6849E-1</td>
<td>1.6464E-1</td>
</tr>
<tr>
<td>$J=160, N=159$</td>
<td>2.0076E-2</td>
<td>2.0076E-2</td>
<td>2.0074E-2</td>
<td>2.0080E-2</td>
<td>1.9887E-2</td>
</tr>
<tr>
<td>$J=320, N=319$</td>
<td>3.4033E-3</td>
<td>3.4033E-3</td>
<td>3.4029E-3</td>
<td>3.4033E-3</td>
<td>3.3855E-3</td>
</tr>
<tr>
<td>$J=960, N=959$</td>
<td>2.1503E-4</td>
<td>2.1503E-4</td>
<td>2.1503E-4</td>
<td>2.1504E-4</td>
<td>2.1514E-4</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
<td>-------------</td>
<td>---------</td>
<td>---------</td>
<td>----------</td>
</tr>
<tr>
<td>$J_0=64, \ N=63$</td>
<td></td>
<td>2.4468E−05</td>
<td>2.4468E−05</td>
<td>2.4453E−05</td>
<td>2.4434E−05</td>
</tr>
<tr>
<td>$J_0=159, \ N=159$</td>
<td></td>
<td>1.1402E−06</td>
<td>1.1402E−06</td>
<td>1.1402E−06</td>
<td>1.1402E−06</td>
</tr>
<tr>
<td>$J_0=219, \ N=219$</td>
<td></td>
<td>4.1918E−07</td>
<td>4.1918E−07</td>
<td>4.1918E−07</td>
<td>4.1918E−07</td>
</tr>
<tr>
<td>$J_0=959, \ N=959$</td>
<td></td>
<td>1.1800E−07</td>
<td>1.1800E−07</td>
<td>1.1800E−07</td>
<td>1.1800E−07</td>
</tr>
</tbody>
</table>
Table 7. Same as Table 5 except for the errors after a 15-day integration in the Williamson test case 5. The result of the high-resolution SH model with $j_0 = 960$ and $N = 958$ is regarded as the reference solution.

<table>
<thead>
<tr>
<th>Resolution - Model</th>
<th>Old DFS [0]</th>
<th>DFS [0]</th>
<th>DFS [1]</th>
<th>DFS [-1]</th>
<th>SH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J=64$, $N=63$</td>
<td>8.2998E-4</td>
<td>8.2972E-4</td>
<td>8.2559E-4</td>
<td>8.2533E-4</td>
<td>8.2575E-4</td>
</tr>
<tr>
<td>$J=320$, $N=319$</td>
<td>8.3815E-4</td>
<td>8.3815E-4</td>
<td>8.3813E-4</td>
<td>8.3807E-4</td>
<td>8.3812E-4</td>
</tr>
</tbody>
</table>
Table 8. Same as Table 7 except for $N \equiv 2j^0 / 3$.

<table>
<thead>
<tr>
<th>Resolution - Model</th>
<th>Old DFS [0]</th>
<th>DFS [0]</th>
<th>DFS [1]</th>
<th>DFS [-1]</th>
<th>SH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J=64, \ N=42$</td>
<td>Unstable</td>
<td>8.249E−4</td>
<td>8.253E−4</td>
<td>8.254E−4</td>
<td>8.258E−4</td>
</tr>
<tr>
<td>$J=200, \ N=259$</td>
<td>Unstable</td>
<td>8.381E−4</td>
<td>8.381E−4</td>
<td>8.381E−4</td>
<td>8.381E−4</td>
</tr>
<tr>
<td>Resolution</td>
<td>Model</td>
<td>Old DFS [0]</td>
<td>DFS [0]</td>
<td>DFS [1]</td>
<td>DFS [-1]</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
<td>------------</td>
<td>---------</td>
<td>---------</td>
<td>----------</td>
</tr>
<tr>
<td>$Z=64$, $N=63$</td>
<td>1.0319E−2</td>
<td>1.0361E−2</td>
<td>2.7824E−3</td>
<td>8.7423E−3</td>
<td>1.0118E−2</td>
</tr>
<tr>
<td>$Z=160$, $N=159$</td>
<td>2.7830E−3</td>
<td>2.7830E−3</td>
<td>1.5615E−3</td>
<td>2.0704E−3</td>
<td>2.7766E−3</td>
</tr>
</tbody>
</table>
Figure 1. Grid [0], Grid [1], and Grid [−1] are three ways of arranging equally spaced latitudinal grid points when the grid interval $\Delta \theta = \pi / 4$. Red circles show the positions of the grid points when the grid interval $\Delta \theta = \pi / 4$. 
Figure 2. Change in values at the grid points due to the meridional wavenumber truncation for (a) even $|m| \geq 2$, and (b) odd $|m| \geq 3$.

We use Grid [0] with the number of latitudinal grid points $J = 64$. Initial values (black) are meridionally transformed from grid space to spectral space, truncated with $N = 42$, and transformed back from spectral space to grid space. Blue: Orszag’s expansion method. Green: Cheong’s expansion method. Red: the new expansion method. (a) Values for even $|m| \geq 2$ when using the DFS method of Orszag (blue), the old DFS method (green), and the new DFS method (red) with Grid [0]. (b) Values for $m = 2$ (orange), 14 (deep sky blue), 30 (lime) when using the SH expansion method with the gaussian grid. (c) Same as (a) except for the values for odd $|m| \geq 3$. (d) Same as (b) except for the values for $m = 3$ (orange), 15 (deep sky blue), 31 (lime).
Figure 3. (a) Same as Fig 2(a) except for \( N = 63 \). (b) Same as (a) except that the values between grid points calculated from the expansion coefficients are also shown.
Figure 4. Predicted height (m) in the Eulerian models after a 12-day integration in Williamson test case 1. (a) New DFS model with Grid 10, (b) New DFS model with Grid 11, (c) New DFS model with Grid −1, (d) SH model. The number of longitudinal \( I \) and latitudinal \( J \) grid points is shown in the form \( I \times J \). In the upper figures, the black contour shows the predicted height, and the red contour shows the reference solution. In the lower figures, color shading shows the difference between the predicted height and the reference solution.
Figure 53. Time series of prediction error of height (m) for 5 days (120 hours) integration in Williamson test case 2 (\(\alpha = \pi/2 - 0.05\)). The number of longitudinal grid points \(I = 128\) and the number of latitudinal grid points in Grid [0] \(J_{\text{0}} = 64\times 64\) indicates the numbers of longitudinal and latitudinal grid points. \(N = 63\) is the truncation wavenumber. Solid, dashed, and dotted lines represent normalized \(L_1\), \(L_2\), and \(L_{\infty}\) errors, respectively. The colors blue, green, red, purple, and orange represent the models using SH, old DFS with Grid [0], new DFS with Grid [0], new DFS with Grid [1], and new DFS with Grid [−1], respectively.
Figure 64. Predicted height (m) after a 15-day integration in Williamson test case 5. (a) New DFS model with Grid [0]. (b) New DFS model with Grid [1]. (c) New DFS model with Grid [-1]. (d) Old DFS model with Grid [0]. (e) SH model. (f) SH model at high resolution, which is regarded as the reference solution. The number of longitudinal ($I$) and latitudinal ($J$) grid points is shown in the form $I \times J$. $N$ is the truncation wavenumber. Color shading shows the error with respect to the reference solution.
Figure 75. Longitudinal distributions of meridional wind (m s$^{-1}$) at the grid points near the South Pole after a 15-day integration in Williamson test case 5. Results of the models using Grid [0] with (a) $I = 128$, $J^0 = 64$ and $N = 63$ grid points and truncation wavenumber $N = 63$, and (b) $I = 1920$, $J^0 = 960$ and $N = 959$ grid points. Green (red) lines represent the old (new) DFS models.
Figure 86. Kinetic energy spectrum of horizontal winds (m²s⁻¹) after a 15-day integration in Williamson test case 5.

Results of the models with (a) \( I = 128, J = 64 \) around 128 \( \times \) 64 grid points and \( N = 63 \) (DFS) or \( N = 62 \) (SH), and (b) \( I = 1920, J = 960 \) around 1920 \( \times \) 960 grid points and \( N = 959 \) (DFS) or 958 (SH). The colors blue, green, red, purple, and orange represent the models using SH, old DFS with Grid [0], new DFS with Grid [0], new DFS with Grid [1], and new DFS with Grid [−1], respectively.
Figure 97. Same as Fig. 64, except with truncation wavenumber $N$. 
Figure 108. Same as Fig. 86, except with truncation wavenumber $N$. 
Figure 11. Same as Fig. 6 except for predicted height (m) after a 14-day integration in Williamson test case 6.
Figure 129. Predicted vorticity (s$^{-1}$) after a 6-day integration in the Galewsky test case. (a) The new DFS model with Grid [0], and (b) the SH model at 1.3 km resolution with $I = 30720, J^0 = 15360$ grid points and $N = 10239$. 

(a) DFS [0] 30720 x 15360  $N = 10239$

(b) SH 30720 x 15360  $N = 10239$
Figure 130. Kinetic energy spectrum of horizontal winds (m$^2$s$^{-2}$) after a 6-day integration in the Galewsky test case. (a) Results of the models with $I = 30720$, $J^0 = 15360$, and $N = 10239$ grid points. The colors blue, green, and red represent the models using SH, old DFS with Grid [0], and DFS with Grid [0], DFS with Grid [1], and DFS with Grid [-1], respectively. (b) As (a), but showing the high-wavenumber region.
Elapsed time (s) for (a) 15-day integration in Williamson test case 5 in the SH model and the new DFS model at 20 km resolution with $I = 1920$, $J = 960$, and $N = 959$, and (b) 6-day integration in the Galewsky test case at 1.3 km resolution with $I = 30720$, $J = 15360$, and $N = 10239$. There is no monitoring output during elapsed time measurement.