

Article

Comparisons of Two RBF-FD Approaches Based on Polyharmonic Splines for Solving the Shallow Water Equations on the Sphere

Fuqiang Lu ¹, Kaimeng Ding ^{2,*} , Yuanyuan Zhang ¹  and Zhenjie Yu ³

¹ School of Computer Science and Information Engineering, Changzhou Institute of Technology, Changzhou 213032, China

² School of Networks and Tele-Communications Engineering, Jinling Institute of Technology, Nanjing 211169, China

³ School of Artificial Intelligence, China University of Geosciences Beijing, Beijing 100080, China

* Correspondence: dkm@jit.edu.cn; Tel.: +86-18168092159

Abstract

In the last decade, the radial basis function-generated finite difference (RBF-FD) method has been widely used for numerical solutions of partial differential equations on smooth surfaces. In this paper, we present a comparison between two approaches for approximating spherical differential operators using the RBF-FD method based on polyharmonic splines. First, we compare common differential operators, including the surface gradient and the Laplacian operator on the sphere, employing polyharmonic splines with different degrees of augmented multivariate polynomials. Second, we examine the approximation errors for various node layouts and spatial resolutions. In most cases, the RBF-FD method with a constant along the normal direction yields more accurate results. To maintain consistency with the high-order RBF-FD spatial discretization, we employ high-order time integration methods, including two Runge–Kutta schemes and exponential Rosenbrock schemes. Finally, we apply the method to solve the spherical shallow water equations using a series of standard test cases, such as steady geostrophic flow, zonal flow over an isolated mountain, Rossby–Haurwitz wave propagation and perturbed zonal jet flow. The simulation results demonstrate that the proposed method is capable of reproducing the evolution of nonlinear dynamics successfully.

Keywords: RBF-FD methods; polyharmonic spline; stencil size; shallow water equations

1. Introduction

In recent decades, the radial basis function (RBF) method has been one of the most popular mesh free methods [1]. It is widely used in data interpolation on scatter points, 3D surface reconstruction, numerical approximation for PDEs, etc. Compared to the numerical methods based on mesh, such as the finite difference method, the finite element method, etc., the RBF method does not need the mesh's topography and metrics. However, the differential matrix generated by the global RBF method is dense, and as the number of scattered points increases, the matrix tends to become increasingly ill conditioned. In order to relieve the problem, Radial Basis Function-generated Finite Differences (RBF-FD) was presented [2].

In the past decade, several improvements of the RBF-FD method have been proposed: for instance, RBF-HFD (Hermite Finite Difference) accounting for derivative information for added accuracy [3], overlapped RBF-FD with improved computational complexity [4]



Academic Editor: Tomeu Rigo

Received: 16 September 2026

Revised: 12 October 2026

Accepted: 13 October 2026

Published: 18 October 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article

distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

Attribution (CC BY) license.

and oversampled RBF-FD with increased stability properties [5]. However, for the purposes of this paper, we limit our analyses to the simplest and most popular original version [2].

When implementing RBF-FD methods, the basis function should be determined first, usually, the commonly used RBFs are infinitely smooth function, such as Gauss, Multi-quadrics, Inverse Multiquadrics, etc.; however, the basis function includes a shape parameter and the parameter's choice plays a crucial role in the conditioning of interpolation matrices as well as accuracy. As a result, the need to balance accuracy and conditioning through the tuning of the shape parameter becomes a problem. The use of the polyharmonic spline (PHS) basis function with polynomials eliminates this requirement [6]. Instead of tuning shape parameters to accommodate different space resolutions, the only parameter selection required is the degree of the PHS and augmented polynomials, which is pre-selected and remains constant. Therefore, PHS basis becomes a more and more popular choice in the RBF-FD method [7–9].

In this paper, we consider the shallow water equations (SWEs) on a sphere, which are expressed as follows in the 3D Cartesian coordinates [10]:

$$\frac{\partial h}{\partial t} + \nabla \cdot (\vec{V}(h - h_s)) = 0 \quad (1)$$

$$\frac{\partial \vec{V}}{\partial t} + \vec{V} \cdot \nabla \vec{V} + f(\vec{x} \times \vec{V}) + g \nabla h = 0 \quad (2)$$

where $\vec{V} = (u, v, w)$ denotes the wind field at the position vector $\vec{x} = (x, y, z)$ on the sphere, $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)$ denotes the gradient operator in 3D Cartesian coordinate, h denotes the total height and h_s denotes the topography, respectively, f is the Coriolis force and g is gravity constant. The projection operator $\mathbf{P}$ maintains the flow tangent to the sphere surface and is defined as [10,11]:

$$\mathbf{P} = \mathbf{I} - \vec{n} \vec{n}^T = \begin{bmatrix} 1 - x^2 & -xy & -xz \\ -xy & 1 - y^2 & -yz \\ -xz & -yz & 1 - z^2 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_x^T \\ \mathbf{p}_y^T \\ \mathbf{p}_z^T \end{bmatrix} \quad (3)$$

where $\vec{n}$ denotes outward unit normal vector at the position vector $\vec{x}$. Obviously, the $\vec{n}$ is exactly the $\vec{x}$ on a unit sphere. To maintain the motion tangent to the sphere surface, we confine the gradient and divergence operators to the surface by redefining $\nabla \rightarrow \mathbf{P}\nabla$ in (1) and (2). We then project the entire right-hand side of (2) onto the surface of the sphere as well. For the $\frac{\partial u}{\partial t}$ equation, this gives us

$$\frac{\partial u}{\partial t} = -\mathbf{p}_x \cdot \begin{bmatrix} u(\mathbf{p}_x \cdot \nabla)u + v(\mathbf{p}_y \cdot \nabla)u + w(\mathbf{p}_z \cdot \nabla)u \\ u(\mathbf{p}_x \cdot \nabla)v + v(\mathbf{p}_y \cdot \nabla)v + w(\mathbf{p}_z \cdot \nabla)v \\ u(\mathbf{p}_x \cdot \nabla)w + v(\mathbf{p}_y \cdot \nabla)w + w(\mathbf{p}_z \cdot \nabla)w \end{bmatrix} + f \begin{bmatrix} yw - zv \\ zu - xw \\ xv - yu \end{bmatrix} + g \begin{bmatrix} \mathbf{p}_x \cdot \nabla \\ \mathbf{p}_y \cdot \nabla \\ \mathbf{p}_z \cdot \nabla \end{bmatrix} h = -\mathbf{p}_x \cdot RHS \quad (4)$$

Similarly,

$$\frac{\partial v}{\partial t} = -\mathbf{p}_y \cdot RHS \quad (5)$$

$$\frac{\partial w}{\partial t} = -\mathbf{p}_z \cdot RHS \quad (6)$$

$$\frac{\partial h}{\partial t} = -(\mathbf{P}\nabla) \cdot (\vec{V}(h - h_s)) \quad (7)$$

It is clear that the only differential operator that needs to be discretized is the projected spherical gradient operator $\mathbf{P}\nabla$. In Section 2, two RBF-FD approaches are employed to approximate the projected spherical gradient and Laplacian. Comparisons are given for the approximation of the gradient and Laplacian operators using PHS basis functions of

varying powers, combined with augmented polynomials of different degrees. The two approaches are then tested on various spherical scattered node distributions, and the convergence of the approximation errors is demonstrated under spatial refinement.

In Section 3, for the time discretization, high-order time integration methods are used, including two explicit Runge–Kutta schemes and two exponential Rosenbrock schemes, which offer a favorable trade-off between high accuracy and computational cost.

In Section 4, the results of numerical experiments for shallow water equations are shown by a series of benchmarks, including the steady geostrophic flow, the zonal flow past an isolated mountain, the Rossby–Haurwitz wave propagation and the barotropic instability. In each case, we show the comparison between the numerical results and reference solution. The results validate the method could reproduce the spherical shallow water equations dynamics successfully.

2. Numerical Methods

For approximation of (1)–(2), it can be seen that the gradient operator should be discretized firstly. For the approximation of the projected spherical gradient using the RBF-FD method in 3D Cartesian coordinates, two approaches are adopted in this paper. The first approach is a transformation-based method: for a given point, the gradient is approximated on a local quasi-2D plane via a Householder transformation, after which the result is mapped back to the sphere [12]. The second approach, known as the RBF-FD constant-along-normal (CAN) method, approximates the gradient in a local stencil in 3D Cartesian coordinates by incorporating several layers of ghost nodes along the normal direction through the center node [13].

To compare these two RBF-FD approaches, a standard smooth test function defined on the sphere is considered, and the RBF-FD method is employed to approximate both the gradient operator and the Laplacian operator of this function. In our numerical experiments, we evaluate even and odd-order PHS basis functions with various powers, and accuracy assessments are conducted under different orders of augmented polynomials. Subsequently, the two methods are further tested on different types of scattered node distributions on the sphere, and the convergence behavior of the approximation errors for both the gradient and Laplacian operators is examined under spatial refinement of the node sets.

Consider a discrete set of nodes $\mathbf{X} = \{\vec{x}_i\}_{i=1}^N$ placed on the unit sphere surface Γ . Assume that $u : \Gamma \rightarrow \mathbb{R}$ is a differentiable function sampled on $\mathbf{X}$ and the nodes $\vec{x}_{i1}, \dots, \vec{x}_{in}$ are n nearest neighbors to $\vec{x}_i$ measured by the Euclidean distance in $\mathbb{R}^3$. We refer to $\vec{x}_i$ and its n nearest nodes as the stencil on the sphere surface corresponding to $\vec{x}_i$. If we let $\phi_i(r(\vec{x}))$ be any RBF centered at $\vec{x}_i$, and $r(\vec{x})$ denotes the distance function, the corresponding local RBF interpolant $\tilde{u}$ takes the following form:

$$\tilde{u}(\vec{x}) = \sum_{j=1}^n \alpha_j \phi(r(\vec{x} - \vec{x}_{ij})) + \sum_{j=1}^k \varrho_j P_j(\vec{x}), \quad \sum_{j=1}^n \alpha_j P_j(\vec{x}_{ij}) = 0 \quad (8)$$

where the second equation is the polynomial-consistency constraint, and $P(\vec{x})$ denotes the augmented multivariate polynomials.

Since the RBF ϕ_i depends only on the distance $r(\vec{x}) = \|\vec{x} - \vec{x}_i\|$, its ordinary 3D Cartesian gradient is obtained by the chain rule:

$$\nabla \phi_i(r(\vec{x})) = \frac{\vec{x} - \vec{x}_i}{r(\vec{x})} \frac{\partial \phi_i(r(\vec{x}))}{\partial r} \quad (9)$$

Assume we want to approximate a linear operator $\mathcal{L}$, e.g., the components of the gradient $\frac{\partial}{\partial x}$, $\frac{\partial}{\partial y}$ or $\frac{\partial}{\partial z}$, at $\vec{x}_i$ by a linear combination of weighted function values at the nearest n nodes, which form the stencil, i.e.,

$$\mathcal{L}(\tilde{u}(\vec{x})) = \sum_{j=1}^n w_j u_j \quad (10)$$

For instance, if one includes linear polynomials in a three-dimensional case, the weights for the ordinary x -derivative $\mathcal{L} = \partial/\partial x$ at $\vec{x}_i$ are obtained by solving

$$\begin{bmatrix} \phi(\|\vec{x}_1 - \vec{x}_1\|) & \phi(\|\vec{x}_1 - \vec{x}_2\|) & \cdots & \phi(\|\vec{x}_1 - \vec{x}_n\|) & 1 & x_1 & y_1 & z_1 \\ \phi(\|\vec{x}_2 - \vec{x}_1\|) & \phi(\|\vec{x}_2 - \vec{x}_2\|) & \cdots & \phi(\|\vec{x}_2 - \vec{x}_n\|) & 1 & x_2 & y_2 & z_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi(\|\vec{x}_n - \vec{x}_1\|) & \phi(\|\vec{x}_n - \vec{x}_2\|) & \cdots & \phi(\|\vec{x}_n - \vec{x}_n\|) & 1 & x_n & y_n & z_n \\ 1 & 1 & \cdots & 1 & 0 & 0 & 0 & 0 \\ x_1 & x_2 & \cdots & x_n & 0 & 0 & 0 & 0 \\ y_1 & y_2 & \cdots & y_n & 0 & 0 & 0 & 0 \\ z_1 & z_2 & \cdots & z_n & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \\ w_{n+1} \\ w_{n+2} \\ w_{n+3} \\ w_{n+4} \end{bmatrix} = \begin{bmatrix} \left. \frac{x_i - x_1}{r_{i1}} \frac{\partial \phi_1(r)}{\partial r} \right|_{r=r_{i1}} \\ \left. \frac{x_i - x_2}{r_{i2}} \frac{\partial \phi_2(r)}{\partial r} \right|_{r=r_{i2}} \\ \vdots \\ \left. \frac{x_i - x_n}{r_{in}} \frac{\partial \phi_n(r)}{\partial r} \right|_{r=r_{in}} \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (11)$$

where $r_{ij} = \|\vec{x}_i - \vec{x}_j\|$ and only the former n weights $w_1, w_2, \dots, w_n$ should be used [14]. Solving this system at every node assembles the ordinary gradient differentiation matrix G_x ; the matrices G_y and G_z are obtained in the same manner by replacing the last four entries of the right-hand side with $(0, 0, 1, 0)$ and $(0, 0, 0, 1)$, respectively. This construction generalizes directly to higher dimensions and higher polynomial orders.

Once the three ordinary gradient matrices are assembled, the ordinary gradient of u at $\vec{x}_i$ is approximated by

$$\nabla u(\vec{x}_i) \approx \begin{bmatrix} G_x \\ G_y \\ G_z \end{bmatrix} u. \quad (12)$$

Because u is defined only on the sphere, only the component of ∇u tangential to the sphere is physically meaningful. The spherical (surface) gradient is therefore obtained by applying the tangential projection $\mathbf{P}_i = \mathbf{I} - \vec{x}_i \vec{x}_i^T$ to the ordinary gradient, i.e.,

$$\nabla_{\Gamma} u(\vec{x}_i) = \mathbf{P}_i \nabla u(\vec{x}_i) = \mathbf{P}_i \begin{bmatrix} G_x \\ G_y \\ G_z \end{bmatrix} u. \quad (13)$$

Denoting the three rows of the projected gradient operator by G_x^{Γ} , G_y^{Γ} and G_z^{Γ} , i.e.,

$$\begin{bmatrix} G_x^{\Gamma} \\ G_y^{\Gamma} \\ G_z^{\Gamma} \end{bmatrix} = \mathbf{P}_i \begin{bmatrix} G_x \\ G_y \\ G_z \end{bmatrix}, \quad (14)$$

For a tangent vector field $\vec{V} = (u, v, w)$, its discrete surface divergence becomes

$$\nabla_{\Gamma} \cdot \vec{V} = G_x^{\Gamma} u + G_y^{\Gamma} v + G_z^{\Gamma} w, \quad (15)$$

and the surface Laplacian (Laplace–Beltrami operator) is approximated by applying the surface divergence to the surface gradient, i.e.,

$$\Delta_{\Gamma} = \nabla_{\Gamma} \cdot \nabla_{\Gamma} = G_x^{\Gamma} G_x^{\Gamma} + G_y^{\Gamma} G_y^{\Gamma} + G_z^{\Gamma} G_z^{\Gamma}. \quad (16)$$

Although the matrices G_x^{Γ} , G_y^{Γ} and G_z^{Γ} defined in (14) contain the node-dependent projection matrix $\mathbf{P}_i$, the composition in (16) is well defined and does not require any differentiation of $\mathbf{P}$. The rightmost factors act first and map the nodal vector u to the three Cartesian components of the tangential field $\nabla_{\Gamma} u \approx (G_x^{\Gamma} u, G_y^{\Gamma} u, G_z^{\Gamma} u)^T$, whose i -th row uses $\mathbf{P}_i$ evaluated at the node where the gradient is computed. The leftmost factors then apply the discrete projected divergence (15) to this tangential field, and their i -th row again uses $\mathbf{P}_i$ at the node where the divergence is computed. Since, for the Euclidean extension of a vector field $\vec{V}$ off the surface, the projected divergence coincides with the intrinsic surface divergence, $(\mathbf{P}\nabla) \cdot \vec{V} = \nabla_{\Gamma} \cdot \vec{V}$, Equation (16) is exactly the discrete counterpart of $\Delta_{\Gamma} u = \nabla_{\Gamma} \cdot (\nabla_{\Gamma} u) = (\mathbf{P}\nabla) \cdot (\mathbf{P}\nabla u)$. In this chain, $\mathbf{P}$ is always evaluated at the center node of the corresponding stencil and is never differentiated, so no additional normal-derivative terms of $\mathbf{P}$ arise.

In this paper, RBF is specified as the polyharmonic spline [15]:

$$\phi(r) = \begin{cases} r^k, & \text{if } k \text{ is an odd number,} \\ r^k \ln(r), & \text{if } k \text{ is an even number.} \end{cases} \quad (17)$$

To compare these two RBF-FD approaches, we define a standard smooth function on the sphere; a concrete example for the test scalar function is chosen as follows [16]:

$$u(\lambda, \theta) = \cos(m\lambda)(\cos(n\theta))^4 \quad (18)$$

where λ, θ denote the longitude–latitude coordinates, and m, n are two integers. Additionally, the spherical gradient of the test function is

$$\nabla u(\lambda, \theta) = \left[\frac{-m \sin(m\lambda)(\cos(n\theta))^4}{\cos(\theta)} \quad -4n \cos(m\lambda)(\cos(n\theta))^3 \sin(n\theta) \right] \quad (19)$$

Once the numerical solutions for the gradient and Laplacian of the test function are obtained, these would be compared with its analytical solutions. It is common to consider the point-wise errors of approximation for the gradient and the Laplacian operators:

$$\begin{aligned} err_{grad}(u_i) &= \nabla \tilde{u}_i - grad(u_i), \\ err_{Lap}(u_i) &= \Delta \tilde{u}_i - Lap(u_i), \end{aligned} \quad (20)$$

where $\nabla \tilde{u}_i, \Delta \tilde{u}_i$ denote the numerical solutions by the RBF-FD method, $grad(u_i), Lap(u_i)$ denote the analytical solutions for the gradient and Laplacian operator of the test function $u(\lambda_i, \theta_i)$, respectively, and (λ_i, θ_i) denotes the longitude–latitude coordinate of $\vec{x}_i$ on the unit sphere surface. Then, we specify the L_{∞} norms and L_2 norms of absolute error:

$$\begin{aligned} L_{\infty}(err_*(u_i)) &= \max_i |err_*(u_i)|, \\ L_2(err_*(u_i)) &= \sqrt{\frac{1}{N} \sum_{i=1}^N |err_*(u_i)|^2}, \end{aligned} \quad (21)$$

where the subscript $*$ denotes the gradient operator or Laplacian operator, and N denotes the total number of scatter nodes on the sphere.

2.1. PHS with Augmented Polynomials

Both RBF-FD approaches described in Section 2 are implemented based on PHS basis functions. The CAN approach additionally requires two parameters: the number of ghost-node layers placed along the normal direction through each center node and the scaling parameter. A visualization of both RBF-FD approaches for calculating RBF-FD weights at a reference node $\vec{x}_i$ (in red) and its nearest n_s neighbors is shown in Figure 1; both RBF-FD approaches have been applied to the solution of differential equations on the sphere.

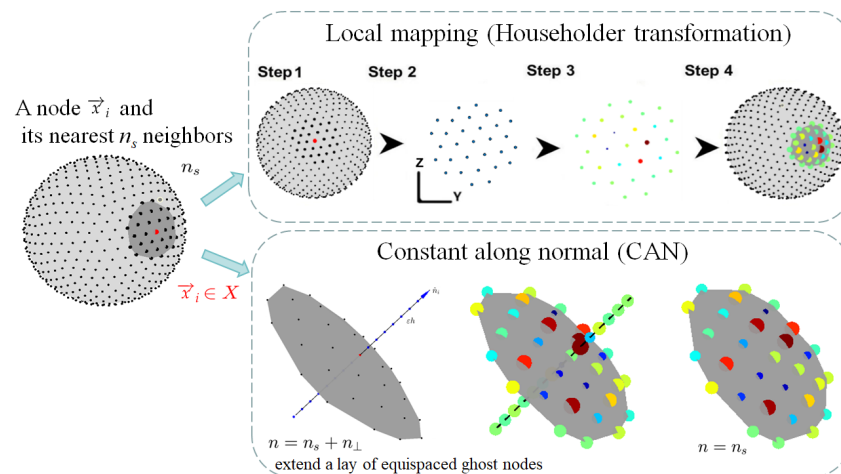


Figure 1. A visualization of two RBF-FD approaches for calculating RBF-FD weights at a reference node $\vec{x}_i$ (in red) and its nearest n_s neighbors (within the dark gray area).

In order to specifically evaluate their performance in approximating the gradient and Laplacian operators, test function (18) is employed. Two integers are set as $m = 2$, $n = 1$, and scatter nodes are distributed by spherical triangle mesh based on the icosahedron with refinement level 4. For the RBF-FD CAN approach, the layer of each center point along the normal direction is set as 15, and the scaling parameter is 0.01. These values follow the recommendations in [13], where several layers of ghost nodes are placed along the normal method, and the along-normal spacing is chosen as a small fraction of the characteristic node spacing so that the off-surface RBF stencil remains well conditioned. With 15 layers and a scaling parameter of 0.01, the local interpolation problem is resolved accurately through the surface while the condition number of the local linear systems stays moderate. Varying the number of layers in the range 8–25 and the scaling parameter between 0.005 and 0.05 produced error curves that are qualitatively the same and did not change the relative comparison between the two approaches; too few layers or too small a scaling parameter degrade the accuracy through ill conditioning, whereas too many layers or too large a scaling parameter increases the computational cost without improving the accuracy. These values were therefore kept fixed for all comparisons in this work.

The performance of two RBF-FD approaches—local mapping and CAN—is compared in terms of the maximum absolute error of approximations for gradient and Laplacian operators. The study systematically varies with increasing stencil size and the degree of appended polynomials (PHS + poly2 through PHS + poly5; polyX denotes a multivariate polynomial of degree X).

- PHS + poly2 (Figures 2 and 3): Both approaches display oscillatory errors with increasing stencil size.
- PHS + poly3 (Figures 4 and 5): The oscillatory trend of maximum absolute error remains pronounced for both gradient and Laplacian operators.
- PHS + poly4 (Figures 6 and 7): Error fluctuations persist, independent of which approach is used.

- PHS + poly5 (Figures 8 and 9): The oscillatory behavior is still observed, confirming that increasing polynomial degree does not suppress the oscillatory nature of the error.

Across all cases, the maximum absolute error of approximations for the gradient or Laplacian operator exhibits an oscillatory behavior rather than monotonic convergence, irrespective of whether the RBF-FD local mapping or the constant-along-normal (CAN) approach is employed [17].

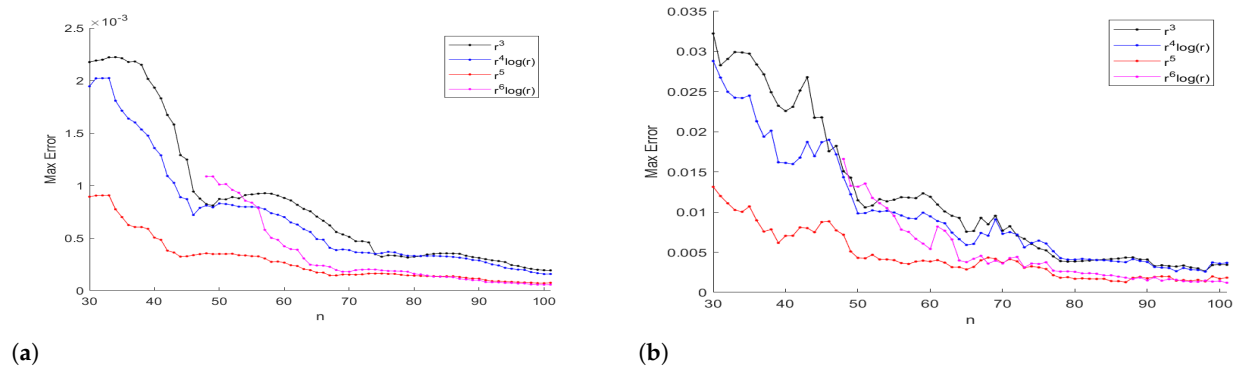


Figure 2. Max absolute error for two differential operators approximated by RBF-FD local mapping (Householder transformation) approach based on PHS + poly2 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

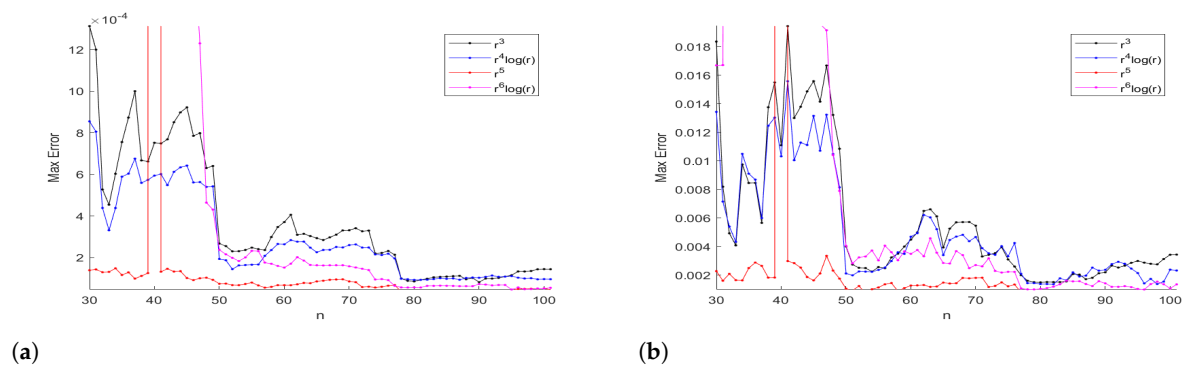


Figure 3. Max absolute error for two differential operators approximated by RBF-FD CAN approach based on PHS + poly2 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

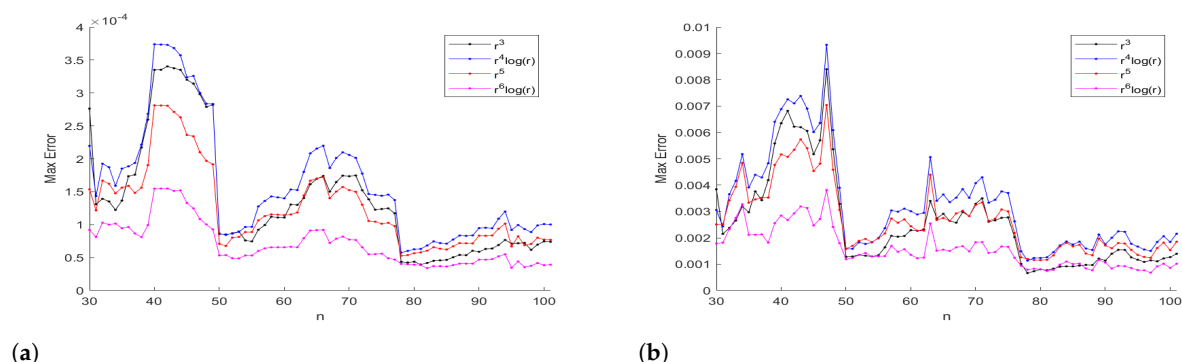


Figure 4. Max absolute error for two differential operators approximated by RBF-FD local mapping (Householder transformation) approach based on PHS + poly3 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

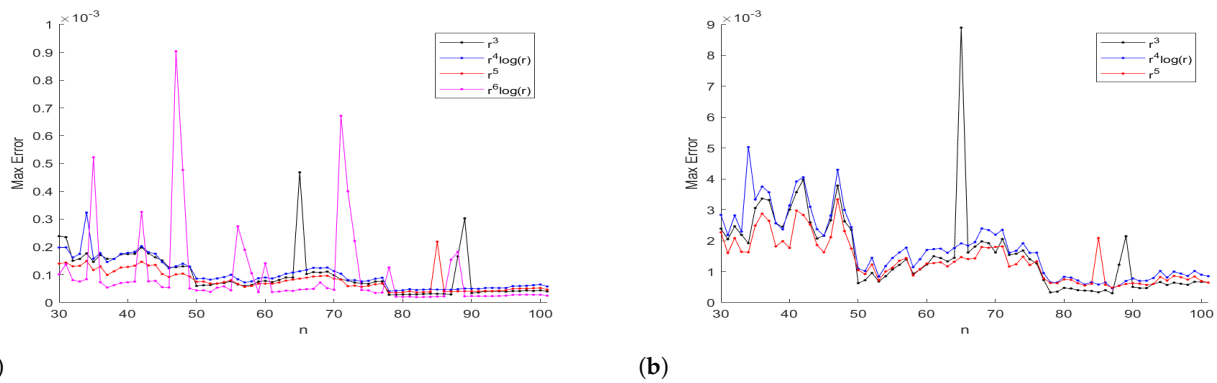


Figure 5. Max absolute error for two differential operators approximated by RBF-FD CAN approach based on PHS + poly3 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

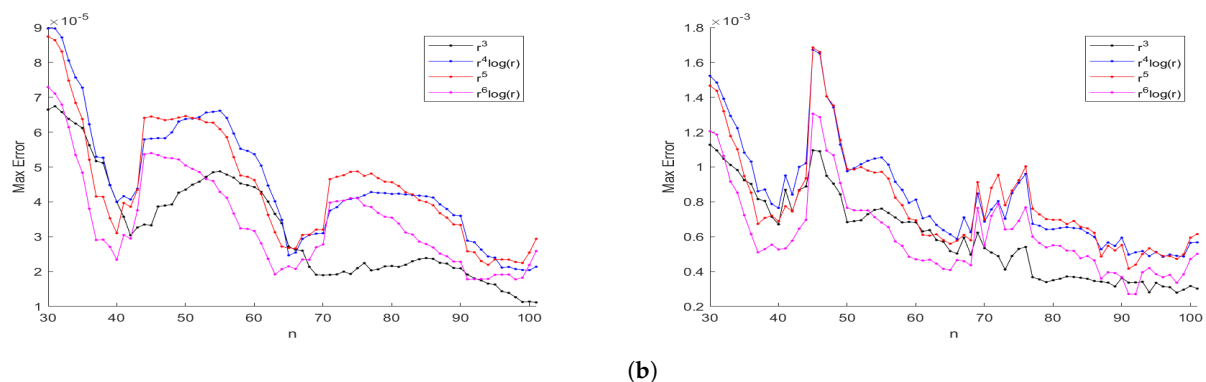


Figure 6. Max absolute error for two differential operators approximated by RBF-FD local mapping (Householder transformation) approach based on PHS + poly4 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

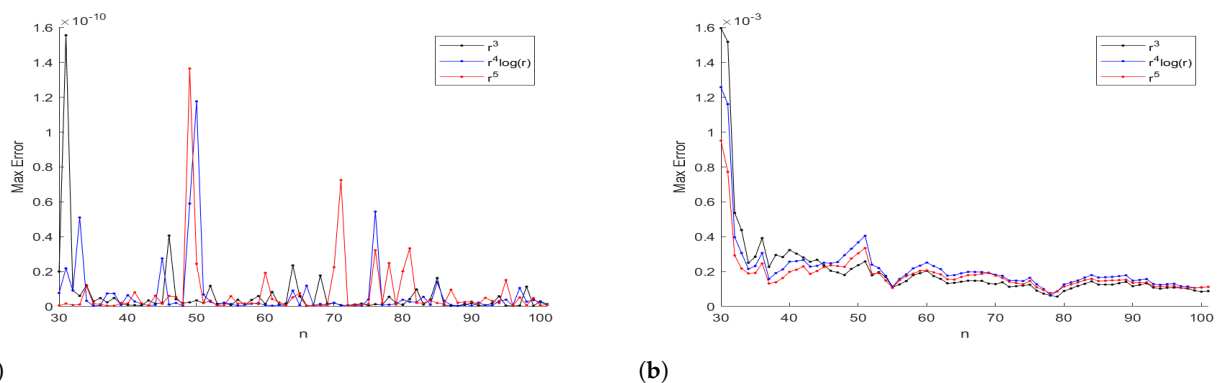


Figure 7. Max absolute error for two differential operators approximated by RBF-FD CAN approach based on PHS + poly4 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

The oscillatory dependence of the error on the stencil size is a known characteristic of PHS-based RBF-FD approximations [17,18]. It arises because the n nearest neighbours of a node are arranged in approximately concentric rings so that increasing n does not change the local point set smoothly: whenever a node from a farther ring enters the stencil, the local interpolation problem, the conditioning of the small linear system and the balance between the PHS and the polynomial parts all change abruptly. The pronounced jump observed near $n = 50$ in Figures 2–9 corresponds to the incorporation of such a farther ring of neighbours; since both approaches share the same surface node set and the same nearest-neighbour selection, the jump appears at essentially the same stencil size for both

methods. The somewhat more irregular behavior of the CAN approach is attributed to the additional off-surface ghost nodes, which make the local RBF systems more ill conditioned and therefore more sensitive to changes in the stencil than the Householder (local-mapping) approach, even though CAN generally attains a smaller error.

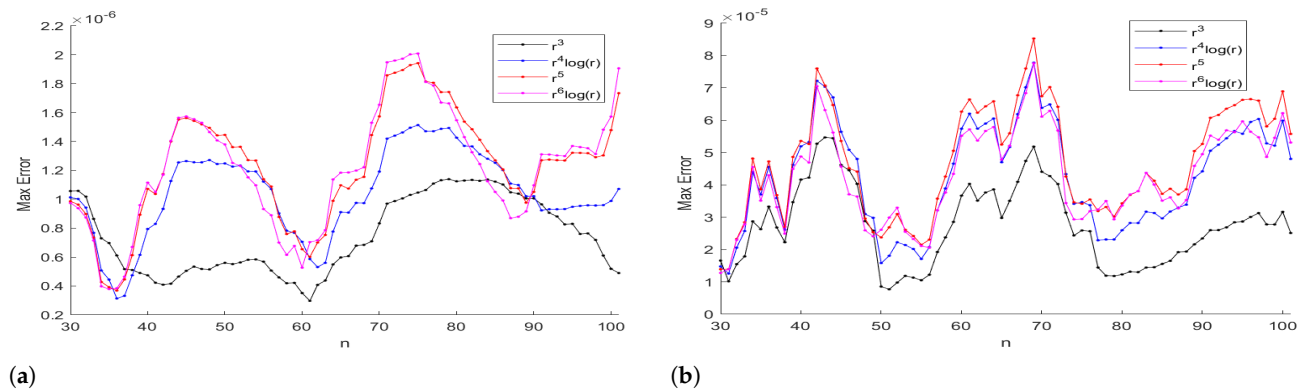


Figure 8. Max absolute error for two differential operators approximated by RBF-FD local mapping (Householder transformation) approach based on PHS + poly5 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

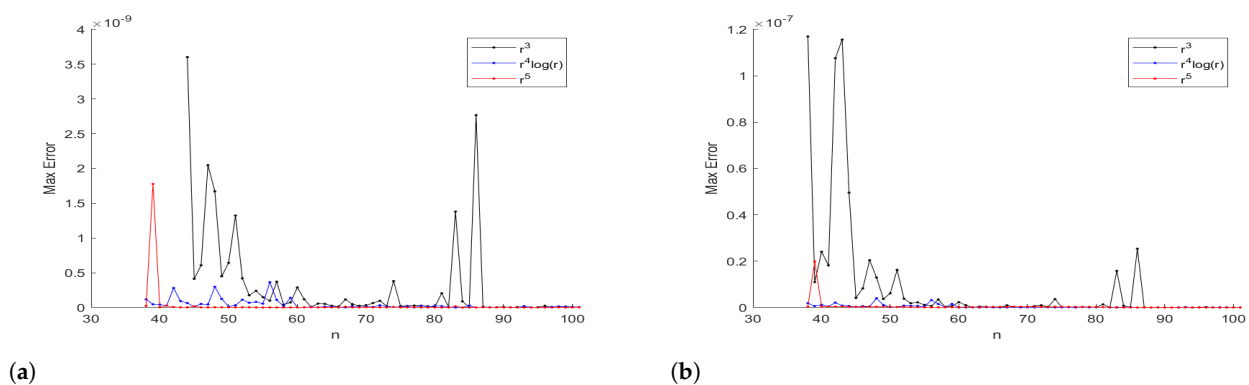


Figure 9. Max absolute error for two differential operators approximated by RBF-FD CAN approach based on PHS + poly5 with increasing stencil size. (a) Gradient operator; (b) Laplacian operator.

The following could be concluded:

1. The approximation error is mainly determined by the degree of augmented multivariate polynomials [19,20], generally, the higher the degree of the polynomials, the smaller the error.
2. When a specific PHS basis with fixed order polynomials is utilized, the approximation error oscillates with the increasing stencil size. This phenomenon persists regardless of which RBF-FD approach is used. It can be concluded that there is no single globally optimal stencil size, but local minima exist within certain ranges of the stencil size.
3. When the order of augmented polynomials is definite, either the power of the PHS basis is even or odd, which has little impact on the approximation error [18]. A higher value of power of the basis function tends to produce somewhat more accurate results but may cause greater instability.
4. When the order of augmented polynomials is low, the larger local stencil tends to generate smaller approximation error.
5. Generally, under the same conditions, the RBF-FD CAN approach produces more accurate results than the local mapping approach. In addition, the approximation for the spherical gradient operator exhibits much more accurate results than that of the spherical Laplacian operator.

2.2. Different Node Layout

The uniformity of the scattered points, i.e., whether the node distribution is regular, has a significant effect upon the accuracy, convergence behavior, stability and computational efficiency of the RBF-FD method. To address this, various strategies have been developed for generating quasi-uniform node sets on the spherical surface. Notable examples include the vertices of icosahedron-based spherical triangle mesh (Ico) [21], Fibonacci spherical nodes (Fib) [22], minimum energy configurations (Me) [23], maximum determinant point sets (Md) [24], symmetric t-designs (St) [25] and the Hierarchical Equal Area isoLatitude Pixelization (Healpix) [26].

Figure 10 shows the max error oscillations of the gradient and Laplacian operators approximated by the RBF-FD local mapping method based on PHS $r^5 + \text{poly}2$ on six different types of node layout mentioned above; the total node numbers are 2602 (Md), 2593 (St), 2700 (Healpix) and 2562 for other node sets. Figure 11 shows the same results but by the RBF-FD CAN approach. It can be seen that, in most cases of the local stencil size, the RBF-FD method implemented on Md, Me and Ico nodes produces more accurate results, while the case of Fib, St and Healpix nodes exhibits considerably less accurate behavior.

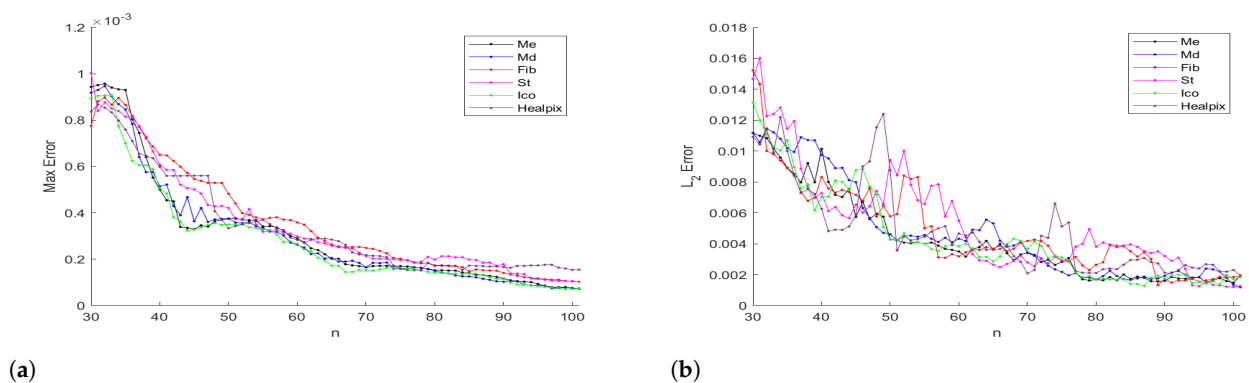


Figure 10. Max absolute error for gradient operator (a) and Laplacian operator (b) approximated by RBF-FD local mapping approach based on PHS + poly2 with increasing stencil size on six different types of node layout (Me, Md, Fib, St, Ico, Healpix nodes).

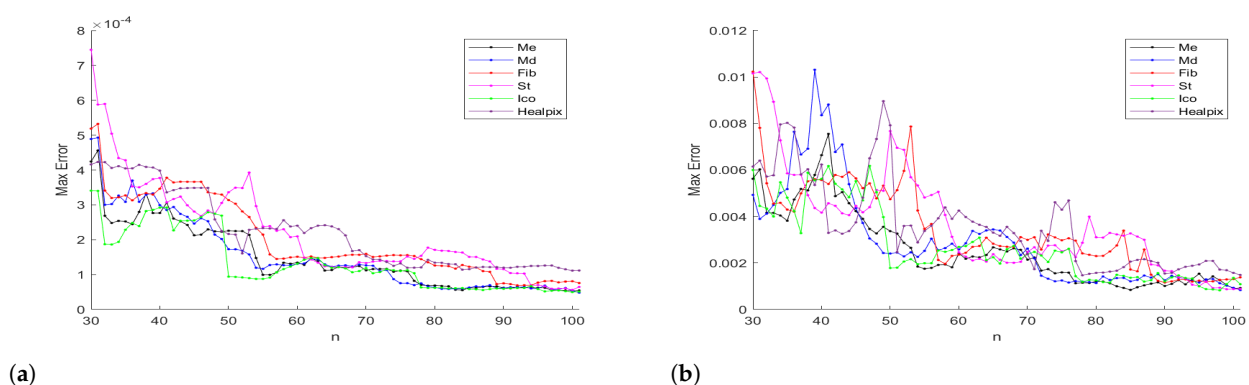


Figure 11. Max absolute error for gradient operator (a) and Laplacian operator (b) approximated by RBF-FD CAN approach based on PHS + poly2 with increasing stencil size on six different types of node layout (Me, Md, Fib, St, Ico, Healpix nodes).

2.3. Node Refinement

The next idea we explore is the convergence of the approximation error for the gradient and Laplacian operator. The Md nodes are utilized and refined with total node numbers 121, 484, 1936 and 7744. In Table 1, the L_∞ norm and L_2 norm of approximation errors for the gradient operator and Laplacian operator by the RBF-FD local mapping approach and

RBF-FD CAN approach based on PHS r^5 with second degree polynomials ($r^5 + \text{poly2}$) were presented in first eight rows; the local stencil size is set as 31. The same results but based on PHS r^5 with third-order polynomials ($r^5 + \text{poly3}$) were shown in the last eight rows.

Table 1. The L_∞ and L_2 norms of the approximation error for the gradient and Laplacian operator on a refinement of Md nodes by two RBF-FD approaches.

RBF-FD Approaches	Node Number	$L_\infty(\text{err}_{grad})$	$L_2(\text{err}_{grad})$	$L_\infty(\text{err}_{Lap})$	$L_2(\text{err}_{Lap})$
local mapping (PHS $r^5 + \text{poly2}$)	121	4.8844×10^{-2}	2.7161×10^{-2}	4.4164×10^{-1}	1.8366×10^{-1}
	484	4.4191×10^{-3}	2.2249×10^{-3}	3.7675×10^{-2}	1.2592×10^{-2}
	1936	1.1515×10^{-3}	5.9634×10^{-4}	1.4188×10^{-2}	5.0531×10^{-3}
	7744	3.2099×10^{-4}	1.5641×10^{-4}	4.6608×10^{-3}	1.5175×10^{-3}
CAN (PHS $r^5 + \text{poly2}$)	121	1.5727×10^{-2}	8.2440×10^{-3}	1.3053×10^{-1}	6.3709×10^{-2}
	484	2.4301×10^{-3}	1.1866×10^{-3}	3.1097×10^{-2}	1.4007×10^{-2}
	1936	3.5592×10^{-4}	1.3279×10^{-4}	5.0011×10^{-3}	1.5100×10^{-3}
	7744	1.4275×10^{-4}	2.6506×10^{-5}	3.1406×10^{-3}	4.8173×10^{-4}
local mapping (PHS $r^5 + \text{poly3}$)	121	4.0675×10^{-2}	2.2366×10^{-2}	4.3547×10^{-1}	1.8248×10^{-1}
	484	2.7745×10^{-3}	1.0808×10^{-3}	3.1747×10^{-2}	1.2438×10^{-2}
	1936	2.8851×10^{-4}	8.2426×10^{-5}	4.1018×10^{-3}	1.0463×10^{-3}
	7744	4.1045×10^{-5}	8.5454×10^{-6}	7.7491×10^{-4}	1.8719×10^{-4}
CAN (PHS $r^5 + \text{poly3}$)	121	1.1733×10^{-2}	5.2154×10^{-3}	8.9519×10^{-2}	4.2629×10^{-2}
	484	1.3937×10^{-3}	5.4859×10^{-4}	1.6496×10^{-2}	4.2143×10^{-3}
	1936	1.9573×10^{-4}	6.7904×10^{-5}	3.2519×10^{-3}	8.5762×10^{-4}
	7744	2.7417×10^{-5}	8.4096×10^{-6}	1.0291×10^{-3}	1.9991×10^{-4}

In Table 1, as the scatter nodes are refined, for each RBF-FD approach, the approximation for the gradient operator is more accurate than that for Laplacian operator. Compared with the RBF-FD local mapping approach, the RBF-FD CAN approach achieves slightly more accurate results.

From Figures 12–15, it can be seen that the convergence rate of L_2 norm of approximation errors for the gradient operator by two RBF-FD approaches remains consistent with the degree of augmented polynomials, while the L_2 norm convergence rate of approximation errors for the Laplacian operator is somewhat undesirable and unstable. In all, for RBF-FD methods based on PHS, the overall accuracy is predominantly governed by the degree of the augmented polynomials.

- Convergence rate for PHS + poly2 (Figures 12 and 13): The convergence rate of the L_2 norm of approximation errors for the gradient operator and Laplacian operator is about 2.
- Convergence rate for PHS + poly3 (Figures 14 and 15): The convergence rate of the L_2 norm of approximation errors for the gradient operator and Laplacian operator is about 3.

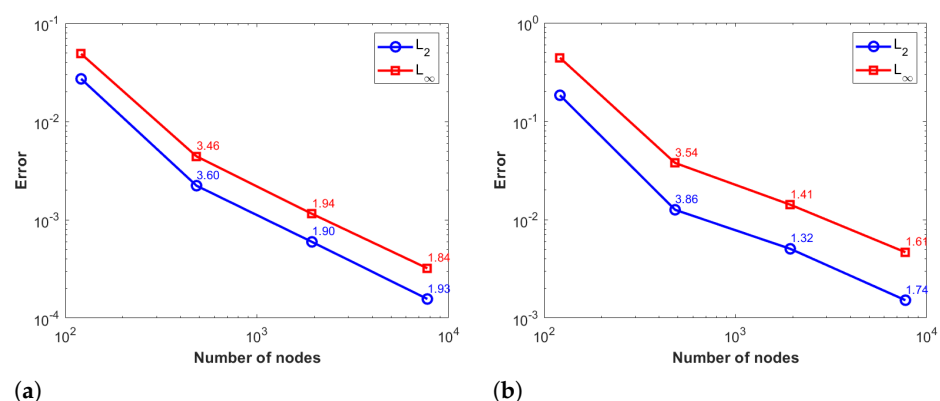


Figure 12. Convergence rate of L_∞ and L_2 norms of approximation errors for (a) the gradient operator and (b) Laplacian operator approximated by RBF-FD local mapping (Householder transformation) method based on PHS $r^5 + \text{poly2}$.

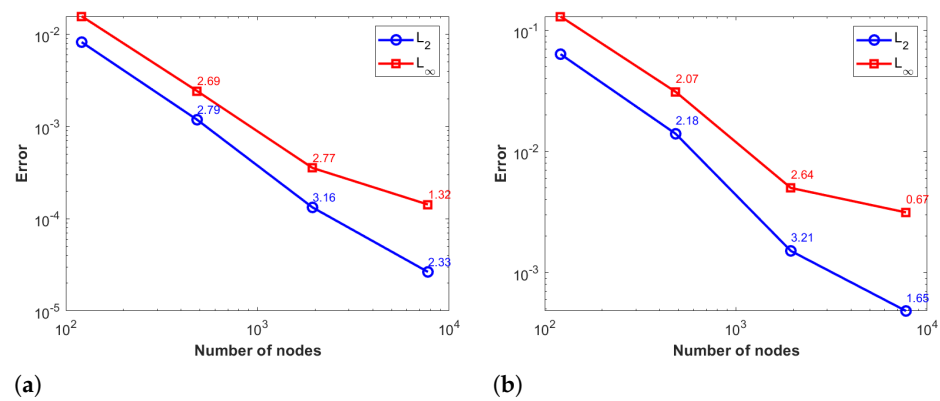


Figure 13. Convergence rate of L_∞ and L_2 norms of approximation errors for (a) the gradient operator and (b) Laplacian operator approximated by RBF-FD CAN approach based on PHS $r^5 + \text{poly}2$.

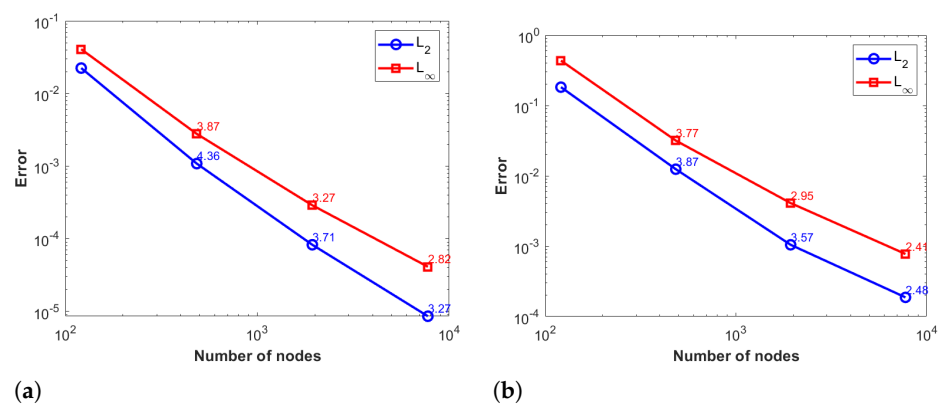


Figure 14. Convergence rate of L_∞ and L_2 norms of approximation errors for (a) the gradient operator and (b) Laplacian operator approximated by RBF-FD local mapping (Householder transformation) method based on PHS $r^5 + \text{poly}3$.

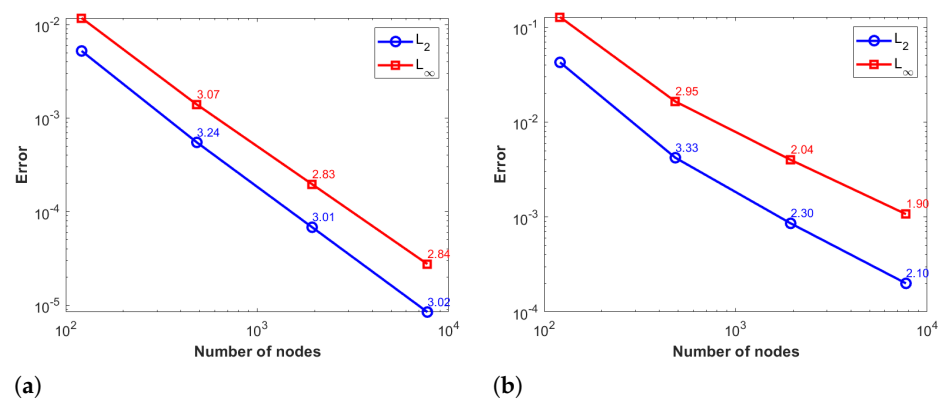


Figure 15. Convergence rate of L_∞ and L_2 norms of approximation errors for (a) the gradient operator and (b) Laplacian operator approximated by RBF-FD CAN approach based on PHS $r^5 + \text{poly}3$.

To quantify these observations, the empirical convergence order is estimated from the data in Table 1 by the finite-difference formula:

$$p = \frac{\ln(\text{err}_{i+1}) - \ln(\text{err}_i)}{\ln(h_{i+1}) - \ln(h_i)}, \quad (22)$$

where h_i, h_{i+1} denote the characteristic node spacing of two successive refinement and $\text{err}_i, \text{err}_{i+1}$ denote the corresponding error norms at these spacings. The convergence orders

of the L_∞ and L_2 error norms for the gradient operator and Laplacian operator obtained between successively refined node sets are listed in Table 2.

Table 2. Estimated convergence order p of the L_∞ and L_2 error norms with respect to the node spacing h , computed from (22) between successively refined node sets for the two RBF-FD approaches and two polynomial degrees.

RBF-FD Approaches	Node Range	Order of $L_\infty(err_{grad})$	Order of $L_2(err_{grad})$	Order of $L_\infty(err_{Lap})$	Order of $L_2(err_{Lap})$
local mapping (PHS $r^5 + \text{poly}2$)	121 $\rightarrow$ 484	3.47	3.61	3.55	3.87
	484 $\rightarrow$ 1936	1.94	1.90	1.41	1.32
	1936 $\rightarrow$ 7744	1.84	1.93	1.61	1.74
CAN (PHS $r^5 + \text{poly}2$)	121 $\rightarrow$ 484	2.69	2.80	2.07	2.19
	484 $\rightarrow$ 1936	2.77	3.16	2.64	3.21
	1936 $\rightarrow$ 7744	1.32	2.32	0.67	1.65
local mapping (PHS $r^5 + \text{poly}3$)	121 $\rightarrow$ 484	3.87	4.37	3.78	3.87
	484 $\rightarrow$ 1936	3.27	3.71	2.95	3.57
	1936 $\rightarrow$ 7744	2.81	3.27	2.40	2.48
CAN (PHS $r^5 + \text{poly}3$)	121 $\rightarrow$ 484	3.07	3.25	2.44	3.34
	484 $\rightarrow$ 1936	2.83	3.01	2.34	2.30
	1936 $\rightarrow$ 7744	2.84	3.01	1.66	2.10

It can be seen that the coarsest interval (The total number of nodes $N = 121$ to $N = 484$) is pre-asymptotic and noticeably overestimates the rate, whereas the finest interval ($N = 1936$ to $N = 7744$) approaches the expected values: for the gradient operator, the convergence order of the L_2 error norm tends to 2 and 3 for the second- and third-degree augmented polynomials, respectively, for both RBF-FD approaches; whereas for the Laplacian operator, the convergence order tends to be smaller and less regular, reflecting the second-order (and therefore more ill-conditioned) nature of the composed operator (16).

2.4. Hyper-Viscosity

For RBF-FD methods, the eigenvalue spectrum of differentiation matrices, particularly those discretizing advection operators, non-physical “spurious modes” are present in most cases. These modes possess eigenvalues with positive real parts, causing the time integration to grow exponentially and ultimately diverge. To damp these small-scale modes without affecting the well-resolved large-scale dynamics, it is common practice to append a hyper-viscous term to the right-hand side of each prognostic equation:

$$\frac{\partial q}{\partial t} = F(q) + (-1)^{p+1} \nu \Delta_\Gamma^p q, \quad (23)$$

where q denotes the height h or any Cartesian component of the velocity $\vec{V}$, $F(q)$ is the corresponding right-hand side of (4)–(7), Δ_Γ is the Laplace–Beltrami operator, p is the order of the hyper-Laplacian and $\nu > 0$ is the dissipation coefficient. The sign factor $(-1)^{p+1}$ renders the damping operator negative semi-definite so that high-wavenumber modes are strongly damped while the low-wavenumber modes remain nearly unaffected.

According to the empirical formula $p = \text{int}(1.5 \ln n)$, where $\text{int}(\cdot)$ denotes the integer part and n is the stencil size, the order p of the hyper-viscosity term grows logarithmically with the stencil size. This guideline was inferred from the numerical experiments reported in the RBF-FD literature [27] and confirmed by our own tests: as the stencil size (and hence the number of resolved near-grid scales) increases, a higher-order hyper-Laplacian is required to confine the dissipation to the smallest resolved scales, while the logarithmic growth keeps the operator from over-smoothing the intermediate wavenumbers. Because $\text{int}(\cdot)$ returns an integer, p is always a positive integer. In the discrete setting, Δ_Γ is

replaced by the RBF-FD Laplacian matrix defined in (16), Δ_Γ^p is realized by p successive multiplications with Δ_Γ and no fractional power (operator square root) of Δ_Γ is required.

The selection of the coefficient ν for the hyperviscosity term is by no means straightforward, and its appropriate value is typically closely tied to the specific problem being solved. In this work, ν is calibrated [10,14] so that the highest-wavenumber mode is damped by a prescribed fraction within a single time step. To this end, let λ_L denote the largest (in magnitude) eigenvalue of the discrete Laplacian matrix $L = \Delta_\Gamma$; this eigenvalue locates the “grid scale” because the spectral radius of the RBF-FD Laplacian on spherical nodes is strongly correlated with the node distribution and automatically incorporates geometric effects such as node non-uniformity. The eigenvalue of the hyper-viscous operator (23) for this mode is $(-1)^{p+1}\nu\lambda_L^p$ so that, over one time step Δt , the highest-wavenumber mode decays by the fraction $C_{\text{hyp}} = \nu|\lambda_L|^p\Delta t$. Fixing this per-step decay, i.e., prescribing $0.1 \leq C_{\text{hyp}} \leq 0.5$, yields

$$\nu = \frac{C_{\text{hyp}}}{|\lambda_L|^p \Delta t}, \quad (24)$$

so that ν is inversely proportional to the p -th power of the largest eigenvalue of the Laplacian and to the time step. This keeps the hyper-viscosity a selective dissipation rather than a hard cutoff and, since the operator acts at every step of the explicit time integration, makes the per-step damping identical across different resolutions, different orders p and different time steps, thereby guaranteeing physical comparability across resolutions. Alternatively, an eigenvalue-based tuning of the discrete operator can be used [13,18]: the eigenvalue spectrum of the RBF-FD advection operator is inspected, the spurious modes with positive real parts are identified and ν is chosen as the smallest value that moves these modes into the left half-plane, i.e., $\nu|\lambda_L|^p \gtrsim \max_{\text{Re}(\lambda_s) > 0} \text{Re}(\lambda_s)$, where λ_s denotes the spurious eigenvalues. An empirical a posteriori calibration is also common in practice [10,17]: one starts with a small value of ν and increases it until the high-frequency noise is eliminated while the resolved dynamics and the reference error norms remain essentially unchanged. In all cases, ν is chosen small enough to leave the resolved dynamics unchanged; the values used in each numerical experiment are stated in Section 4.

3. Time Integration

In order to be compatible with the high-order RBF-FD spatial discretization, high-order time integration methods are employed accordingly [28]. In this work, we use two classes of time integrators. The first class consists of the classical explicit Runge–Kutta (RK) methods, including the fourth-order scheme (RK4) and John Butcher’s fifth-order scheme (RK5) [29]; since both are standard, their implementation is omitted here.

The second class consists of exponential Rosenbrock integrators, which are well suited to stiff problems and provide an efficient alternative to fully implicit methods. Writing the semi-discrete system (4)–(7) in the compact form $\frac{\partial q}{\partial t} = F(q)$, let $J_n = \frac{\partial F(q_n)}{\partial q}$ denote the Jacobian at the current state q_n , and split $F(q) = J_n q + g_n(q)$ with the residual $g_n(q) = F(q) - J_n q$. A general s -stage exponential Rosenbrock method then reads

$$\begin{aligned} Q_{ni} &= q_n + c_i \Delta t \varphi_1(c_i \Delta t J_n) F(q_n) + \Delta t \sum_{j=2}^{i-1} a_{ij} (\Delta t J_n) D_{nj}, \quad i = 2, \dots, s, \\ q_{n+1} &= q_n + \Delta t \varphi_1(\Delta t J_n) F(q_n) + \Delta t \sum_{i=2}^s b_i (\Delta t J_n) D_{ni}, \end{aligned} \quad (25)$$

where $D_{ni} = g_n(Q_{ni}) - g_n(q_n)$, and the φ -functions are defined recursively by

$$\varphi_0(z) = e^z, \quad \varphi_k(z) = \frac{1}{(k-1)!} \int_0^1 e^{(1-\theta)z} \theta^{k-1} d\theta, \quad k \geq 1. \quad (26)$$

In this paper, we adopt the parallel fourth-order 3-stage integrator Pexprb43 and the fifth-order 5-stage integrator Pexprb54s5 proposed by Luan and Ostermann [30]; the corresponding coefficients c_i , $a_{ij}(z)$ and $b_i(z)$ can be found therein. In the implementation, the Jacobian J_n is formed analytically from the right-hand sides of (4)–(7), and the products $\varphi_k(\Delta t J_n)v$ are approximated within a Krylov subspace.

Unlike the explicit RK schemes, the exponential Rosenbrock integrators are not severely constrained by the CFL condition and therefore allow substantially larger time steps. However, this advantage comes at a higher computational cost per step, mainly due to the evaluation of the Jacobian, the solution of large-scale linear systems and the construction of Krylov subspaces.

4. Numerical Experiments

In this section, we report numerical results obtained for SWEs (1) and (2) by RBF-FD approaches coupled with high-order time integrators. Four test cases are considered, including three benchmarks in [31]: the time-independent geostrophic flow, zonal flow over an isolated mountain and the Rossby–Haurwitz wave propagation, followed by the barotropic instability in [32].

The physical constants about the Earth are $R_e = 6.37122 \times 10^6$ m (the mean radius of Earth), $\Omega = 7.292 \times 10^{-5} \text{ s}^{-1}$ (Earth angular velocity) and $g = 9.80616 \text{ ms}^{-2}$ (gravity constant). In order to facilitate comparison to previously published error norms of shallow water test case simulations, we specify the normalized L_∞ and L_2 norms of the approximation error for the height field as

$$L_\infty = \frac{\max |\tilde{h} - h_{ref}|}{\max |h_{ref}|}, \quad (27)$$

$$L_2 = \sqrt{\frac{\sum (\tilde{h} - h_{ref})^2}{\sum (h_{ref})^2}},$$

where $\tilde{h}$ denotes the numerical solution and h_{ref} true solution or reference solution.

Before presenting the individual test cases, we comment on the widely different values of the dissipation coefficient ν quoted below. According to the calibration (24), ν is inversely proportional to the p -th power of the largest eigenvalue $|\lambda_L|$ of the discrete Laplacian matrix and to the time step. For spherical-harmonic modes, $|\lambda_L| \approx \ell_{\max}(\ell_{\max} + 1)$; with $\ell_{\max} \sim 10^2$, i.e., $|\lambda_L| \sim 10^4$, the value required for $p = 5$ is about eight orders of magnitude smaller than the corresponding value for $p = 3$, which explains the main difference between the fifth-order and third-order values below. The remaining spread between the two third-order cases reflects the different small-scale energy of the test flows: the mountain test is a nearly balanced, weakly nonlinear flow with very little grid-scale noise, whereas the Rossby–Haurwitz wave and, above all, the barotropically unstable jet generate substantially stronger small-scale activity and therefore require a larger (but still small) ν . In all cases, ν was calibrated with the strategies described in Section 2.4 so that the spurious modes are damped while the resolved dynamics is left essentially unchanged. Halving or doubling the quoted values only rescales the damping time of the near-cutoff modes; in our tests, this changed the reported error norms by a few percent and left the flow patterns visually

unchanged, whereas values larger by one or two orders of magnitude over-smooth the solution and degrade the physical fields.

4.1. Time-Independent Geostrophic Flow

This test is a steady-state solution to nonlinear SWEs. Similar to the test-case 2 of Williamson et al. [31], the initial height fields are as follows:

$$gh = gh_0 - \left(R\Omega u_0 + \frac{u_0^2}{2} \right) (\sin(\theta) \cos(\alpha) - \cos(\lambda) \cos(\theta) \sin(\alpha))^2 \quad (28)$$

where $gh_0 = 2.94 \times 10^4 \text{ m}^2\text{s}^{-2}$, and α denotes the wind direction angle. The initial wind field in longitude–latitude coordinates is defined as

$$\begin{aligned} u &= u_0(\cos(\theta) \cos(\alpha) + \cos(\lambda) \sin(\theta) \sin(\alpha)), \\ v &= -u_0 \sin(\lambda) \sin(\alpha). \end{aligned} \quad (29)$$

The Coriolis parameter f depends on the wind direction α and is defined as

$$f = 2\Omega(\sin(\theta) \cos(\alpha) - \cos(\lambda) \cos(\theta) \sin(\alpha)) \quad (30)$$

We solve (1) and (2) by two RBF-FD approaches mentioned above based on PHS r^5 + poly4 on the Ico nodes with refinement level 5, where the stencil size is 101, the wind direction angle $\alpha = 0$ and no hyper-viscous term (23) is added. Additionally, two Runge–Kutta methods and exponential Rosenbrock schemes are employed for time integration. The normalized L_∞ and L_2 norms of the approximation error for the height field at day 15 are evaluated for the two RBF-FD approaches and different time integrators.

Figure 16 shows the evolution of normalized L_2 error for the height field between the exact solution and numerical solutions by two RBF-FD approaches and different time integration schemes. For RK4 and RK5 schemes, the time step is set to 5 min; for exponential Rosenbrock schemes, the time step is set to 60 min. Compared with the local mapping approach, the normalized L_2 error of height field between the exact solution and numerical solutions by the RBF-FD CAN approach is about 10^{-6} , which demonstrates better mass conservation. In addition, the fifth-order time integration methods, especially the Pexprb54s5 scheme, presents most accurate results. It is also noted that the error keeps growing mildly in the first several days; however, it grows more rapidly after day 13.

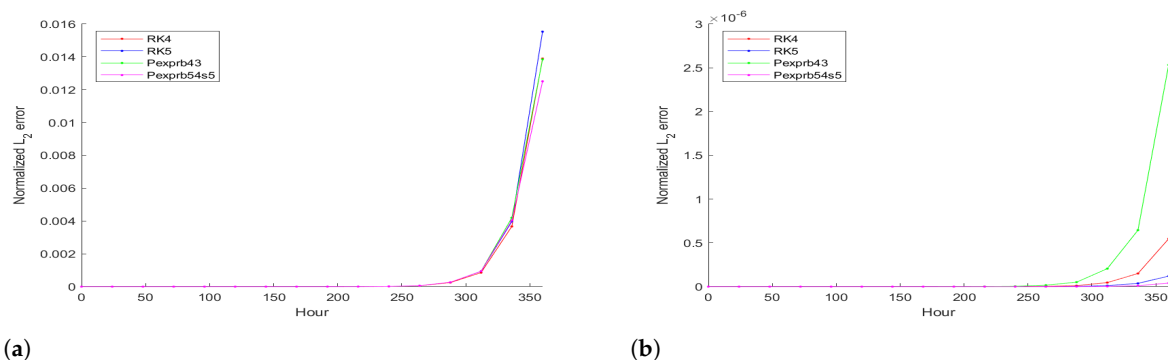


Figure 16. The normalized L_2 error of height field for steady geostrophic flow at day 15 on the Ico nodes with refinement level 5. (a) RBF-FD local mapping approach; (b) RBF-FD CAN approach.

We solve (1) and (2) by the RBF-FD CAN approach coupled with Pexprb54s5 time integrator again. First, without considering hyper-viscous terms, the simulated height field at day 15 and its error field relative to the exact solution are shown in Figure 17. On this

basis, a sixth-order viscous term is added, and the simulation is repeated; the corresponding height field at day 15 and the error field are shown in Figure 18. In addition, we separately tested the results obtained by adding third-, fourth-, and fifth-order hyperviscosity terms. The errors between the computed height fields and initial height fields all fall on the order of 10^{-7} , indicating that the third-order hyperviscosity term is already sufficient in most cases.

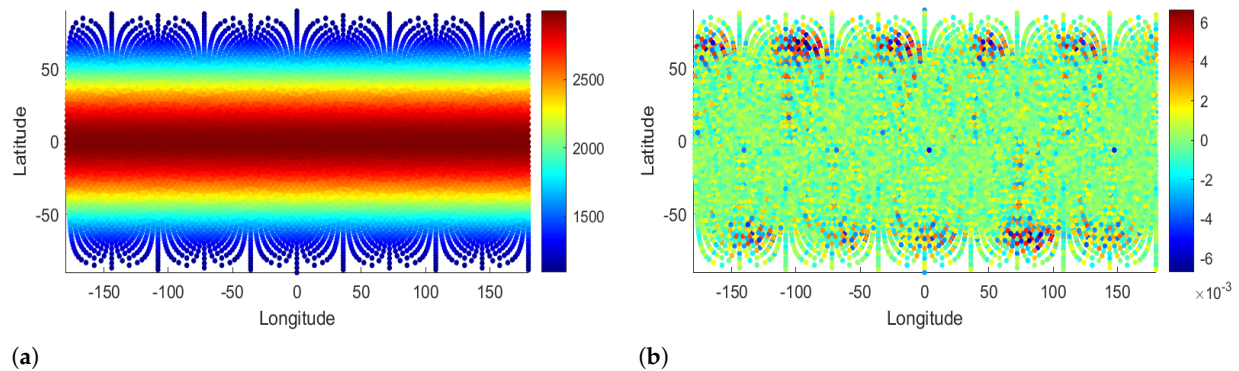


Figure 17. Height field by RBF-FD CAN approach for steady geostrophic flow at day 15 on the Ico nodes with refinement level 5, no hyper-viscous term. (a) Height field; (b) the pointwise error of height field.

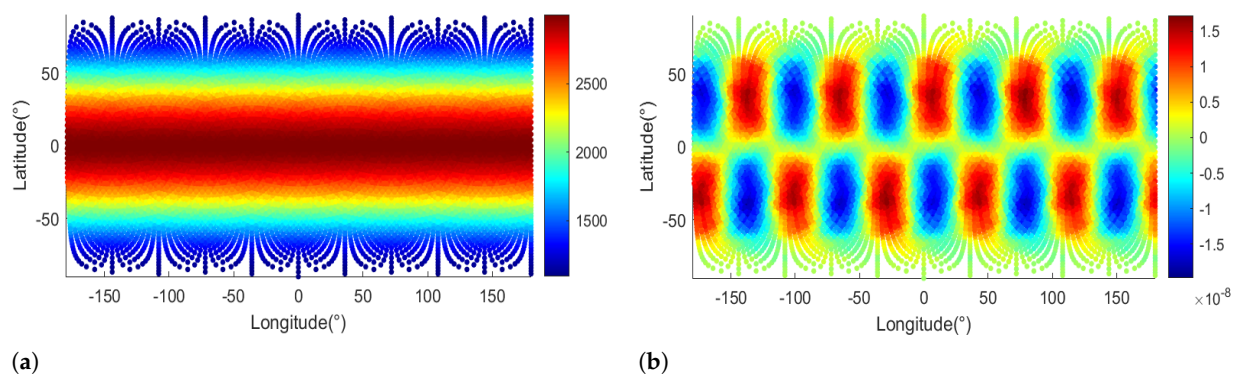


Figure 18. Height field by RBF-FD CAN approach for steady geostrophic flow at day 15 on the Ico nodes with refinement level 5. Sixth-order hyper-viscous term is added. (a) Height field; (b) the pointwise error of height field.

It can be seen that when the relevant parameters are chosen appropriately, adding hyper-viscous terms can not only enhance the numerical stability of the scheme but also effectively improve the accuracy of the computational results. The RBF-FD CAN approach coupled with the Pexprb54s5 time integrator is adopted in the following numerical experiments unless otherwise specified.

4.2. Zonal Flow over an Isolated Mountain

This is the test 5 in [31]. It is a time-dependent case without an analytical solution. The initial condition is similar to that in test case 2, that is, the velocity field is the solid body rotation, and the height field is in the geostrophic balance. The initial data is given by (28) and (29) with the parameters $h_0 = 5960$ m, $u_0 = 20$ m $\cdot$ s $^{-1}$, and the angle of flow direction is $\alpha = 0$. The initial data is perturbed by a conic mountain located at the point $(\lambda_c, \theta_c) = (-\pi/2, \pi/6)$ with a width of $r_0 = \pi/9$. The bottom topography $h_s(\vec{x})$ is

$$h_s(\vec{x}) = 2000 \left(1 - \frac{r(\vec{x})}{r_0} \right),$$

$$r(\vec{x}) = \min \left(r_0, \sqrt{(\lambda - \lambda_c)^2 + (\theta - \theta_c)^2} \right).$$
(31)

We solve (1) and (2) by RBF-FD CAN approaches based on PHS $r^5 + \text{poly4}$ on the Ico nodes with a refinement level 5, Pexprb54s5 scheme, for time integration. The stencil size is 101, and the time interval is set to 1200s. The third-order hyper-viscosity term (23) is added to the height and velocity fields to damp the numerical noise.

The snapshot of the height field at day 15 is shown in Figure 19. It can be seen that the proposed method could accurately model the meander of tracer field around the cone-shaped hill. The results are in good agreement with those obtained by the conservative schemes in [33].

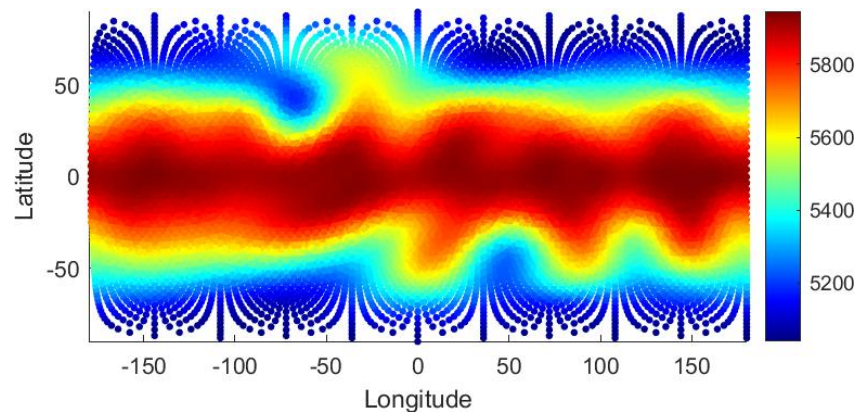


Figure 19. Height field of the zonal flow over an isolated mountain at day 15 on the Ico nodes with refinement level 5.

4.3. Rossby–Haurwitz Wave

The propagation of the Rossby–Haurwitz wave is test 6 in [31]. It is used to assess the qualitative behavior of numerical schemes for (1) and (2). The initial velocity in longitude–latitude coordinates is expressed as

$$\begin{aligned} u(\vec{x}) &= a\omega \cos(\theta) + aK \cos^{R-1}(\theta)(R \sin^2(\theta) - \cos^2(\theta)) \cos(R\lambda), \\ v(\vec{x}) &= -aK \cos^{R-1}(\theta) \sin(\theta) \sin(R\lambda). \end{aligned} \quad (32)$$

The initial height is

$$gh(\vec{x}, t)|_{t=0} = gh_0 + a^2 A(\theta) + a^2 B(\theta) \cos(R\lambda) + a^2 C(\theta) \cos(2R\lambda) \quad (33)$$

where the functions A , B and C are latitude dependent:

$$\begin{aligned} A(\theta) &= \frac{\omega}{2}(2\Omega + \omega) \cos^2(\theta) + \frac{K^2}{4} \cos^{2R}(\theta) \left((R+1) \cos^2(\theta) + (2R^2 - R - 2) - 2R^2 \cos^{-2}(\theta) \right), \\ B(\theta) &= \frac{2K(\Omega + \omega)}{(R+1)(R+2)} \cos^R(\theta) \left((R^2 + 2R + 2) - (R+1)^2 \cos^2(\theta) \right), \\ C(\theta) &= \frac{1}{4} K^2 \cos^{2R}(\theta) \left((R+1) \cos^2(\theta) - (R+2) \right). \end{aligned} \quad (34)$$

The constants are set as $\omega = K = 7.848 \times 10^{-6} \text{ s}^{-1}$, $h_0 = 8000 \text{ m}$ and wave-number $R = 4$. Equations (1) and (2) are solved by the RBF-FD constant along the normal method based on PHS $r^5 + \text{poly4}$, and the Ico node set with refinement level 5 is used. The time interval is set to 1200s, and the stencil size is 101. The third-order hyper-viscosity term (23) is added to the height and velocity fields to damp the numerical noise. The snapshot of the height field at day 14 is shown in Figure 20. The results are in good agreement with those obtained by the conservative schemes in [34].

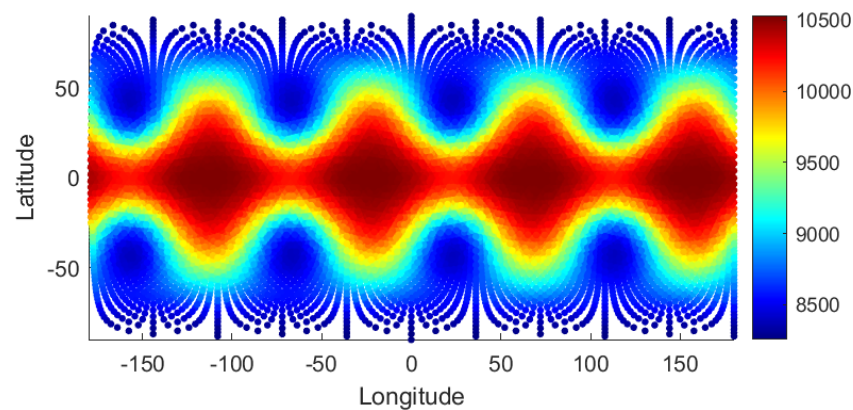


Figure 20. Height field for Rossby–Haurwitz wave with zonal wave-number 4 at day 14 on the Ico nodes with refinement level 5.

4.4. Perturbed Zonal Jet Flow

The perturbed jet flow was introduced by Galewsky et al. [32]. This experiment was designed to test the model ability of simulating the fast and slow motion. The initial flow-vector field is only a function of latitude and is defined as

$$u(\vec{x}, t)|_{t=0} = \begin{cases} \frac{u_{\max}}{e_n} \exp\left(\frac{1}{(\theta - \theta_0)(\theta - \theta_1)}\right), & \theta \in (\theta_0, \theta_1), \\ 0, & \text{otherwise,} \end{cases} \quad (35)$$

where $u_{\max} = 80 \text{ ms}^{-1}$, $\theta_0 = \frac{\pi}{7}$, $\theta_1 = \frac{\pi}{2} - \theta_0$ and $e_n = \exp\left(\frac{-4}{(\theta_1 - \theta_0)^2}\right)$. The initial height field is

$$h(\vec{x}, t)|_{t=0} = h_0 + h'(\lambda, \theta) - \frac{1}{8} \int_{-\frac{\pi}{2}}^{\theta} u(\theta') (af + \tan(\theta') u(\theta')) d\theta' \quad (36)$$

where $h_0 = 10,158.186 \text{ m}$ (reference depth), and a denotes the radius of the earth. The perturbation $h'(\lambda, \theta)$ is expressed as

$$h'(\vec{x}, t) = 120 \cos(\theta) \exp\left(-\left(\frac{\lambda}{\alpha}\right)^2 - \left(\frac{\theta_2 - \theta}{\beta}\right)^2\right), \quad \lambda \in (-\pi, \pi), \quad (37)$$

where $\vec{x}$ denotes the position vector on the sphere, (λ, θ) denotes its longitude and latitude, $\theta_2 = \frac{\pi}{4}$, $\alpha = \frac{1}{3}$ and $\beta = \frac{1}{15}$.

The SWEs (1) and (2) are solved by the RBF-FD CAN approach based on PHS $r^5 + \text{poly4}$, and the Ico node set with refinement level 5 is used. The stencil size is 101, and the time interval is set to 1200s. The fifth-order hyper-viscosity term (23) is added to the height and velocity fields to control the small-scale numerical noise. The zonal jet is barotropically unstable, leading to the growth of perturbations. Driven by barotropic instability, the zonal jet excites Rossby waves. Figure 21 shows the relative vorticity of the flow field at day 6; a series of vortex pairs can be observed alternately distributed along the zonal direction, with approximately seven distinct vortex pairs clearly visible. The positive vorticity cores (shown in red) are located at higher latitudes, while the negative vorticity cores (shown in blue) are located at lower latitudes, with a meridional extent roughly spanning from 40° to 60° . The vorticity pattern is close to those obtained by the conservative schemes in [35].

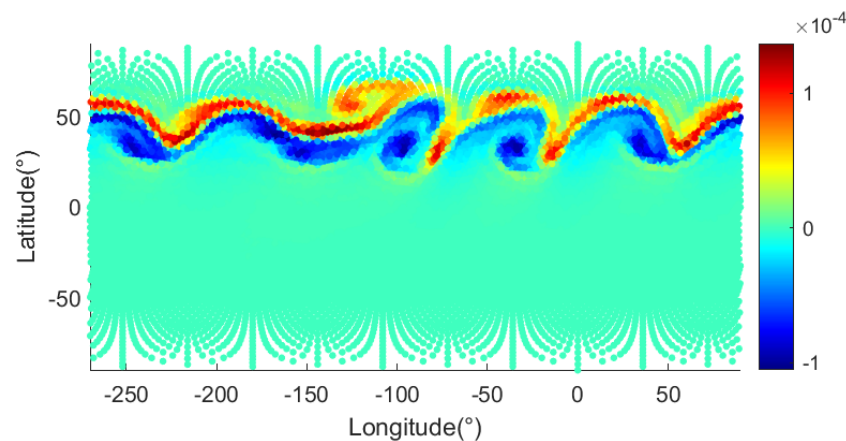


Figure 21. Relative vorticity field of the perturbed zonal jet at day 6 on the Ico node set with refinement level 5.

5. Conclusions

We present a systematic comparison of two RBF-FD approaches based on PHS with augmented polynomials for approximating the gradient operator and Laplacian operator on the sphere. The results show that the RBF-FD CAN approach achieves a higher accuracy in most cases. In terms of time integration, although the introduction of a hyper-viscosity term can significantly enhance the stability of the traditional explicit Runge–Kutta method, the exponential Rosenbrock time integration method exhibits even greater stability, thereby allowing for a much larger time step size than the former. This advantage effectively reduces the number of time steps while maintaining numerical stability. Numerical experiments on the spherical shallow water equations demonstrate that the RBF-FD CAN method, combined with the exponential Rosenbrock time integration method, is promising and can be readily extended to solving partial differential equations on other smooth surfaces.

Author Contributions: Conceptualization, F.L. and K.D.; methodology, F.L. and Y.Z.; software, F.L.; validation, F.L. and K.D.; formal analysis, F.L.; investigation, F.L.; resources, F.L.; data curation, F.L.; writing—original draft preparation, F.L.; writing—review and editing, F.L., K.D. and Y.Z.; visualization, F.L. and Z.Y.; supervision, K.D.; project administration, K.D.; funding acquisition, F.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research is partially supported by the Natural Science Fund for Colleges and Universities in Jiangsu Province (Grant No. 25KJA420001), Jiangsu Young Talents Support Program (Grant No. JSTJ2025796) and Changzhou Young Talents Support Project (No. CZTJ202529).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be made available on request.

Acknowledgments: This research is partially supported by the Jiangsu Provincial University Key Laboratory of Intelligent Multi-source Information Processing and Security, Jiangsu Provincial High-level Talent Cultivation Program (“333 Project”), and the authors thank the anonymous reviewers for carefully reading this paper and their constructive comments and suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RBF	Radial basis function
RBF-FD	Radial basis function-generated finite difference

PDE	Partial differential equation
SWEs	Shallow water equations
PHS	Polyharmonic spline
CAN	Constant along normal
RK4	Fourth-order Runge–Kutta method
RK5	Fifth-order Runge–Kutta method
Pexprb43	Parallel fourth-order exponential Rosenbrock integrator
Pexprb54s5	Parallel fifth-order 5-stage exponential Rosenbrock integrator

References

1. Fasshauer, G.E. *Meshfree Approximation Methods with MATLAB*; World Scientific Publishing Co Pte. Ltd.: Singapore, 2007.
2. Wright, G.B.; Fornberg, B. Scattered node compact finite difference-type formulas generated from radial basis functions. *J. Comput. Phys.* **2006**, *212*, 99–123.
3. Yang, Y.; Soleymani, F.; Barfeie, M.; Tohidi, E. A radial basis function-Hermite finite difference approach to tackle cash-or-nothing and asset-or-nothing options. *J. Comput. Appl. Math.* **2020**, *368*, 112523.
4. Shankar, V. The overlapped radial basis function-finite difference (RBF-FD) method: A generalization of RBF-FD. *J. Comput. Phys.* **2017**, *342*, 211–228.
5. Tominec, I.; Nazarov, M. Residual viscosity stabilized RBF-FD methods for solving nonlinear conservation laws. *J. Sci. Comput.* **2023**, *94*, 1–31.
6. Shahane, S.; Radhakrishnan, A.; Vanka, S.P. A high-order accurate meshless method for solution of incompressible fluid flow problems. *J. Comput. Phys.* **2021**, *445*, 110623.
7. Santos, L.G.C.; Manzanares-Filho, N.; Menon, G.J.; Abreu, E. Comparing RBF-FD approximations based on stabilized Gaussians and on polyharmonic splines with polynomials. *Int. J. Numer. Methods Eng.* **2018**, *115*, 462–500.
8. Flyer, N.; Barnett, G.A.; Wicker, L.J. Enhancing finite differences with radial basis functions: Experiments on the Navier–Stokes equations. *J. Comput. Phys.* **2016**, *316*, 39–62.
9. Hosseinzadeh, N.; Shivanian, E.; Abbasbandy, S. Mesh-free RBF-FD method with polyharmonic splines and polynomials for high-dimensional PDEs and financial option pricing. *Control Optim. Appl. Math.* **2025**, *10*, 193–215.
10. Flyer, N.; Lehto, E.; Blaise, S.; Wright, G.B.; St-Cyr, A. A guide to RBF-generated finite differences for nonlinear transport: Shallow water simulations on a sphere. *J. Comput. Phys.* **2012**, *231*, 4078–4095.
11. Liu, Y.; Qiao, Y.; Feng, X. A stable radial basis function partition of unity method for solving convection–diffusion equations on surfaces. *Eng. Anal. Bound. Elem.* **2023**, *155*, 148–159.
12. Gunderman, D.; Flyer, N.; Fornberg, B. Transport schemes in spherical geometries using spline-based RBF-FD with polynomials. *J. Comput. Phys.* **2020**, *408*, 109256.
13. Bayona, V.; Petras, A.; Piret, C.; Ruuth, S.J. A meshfree RBF-FD constant along normal method for solving PDEs on surfaces. *SIAM J. Sci. Comput.* **2024**, *46*, A3897–A3921.
14. Fornberg, B.; Flyer, N. *A Primer on Radial Basis Functions with Applications to the Geosciences*; SIAM: Philadelphia, PA, USA, 2015.
15. Liu, Z.; Barfeie, M.; Soleymani, F. Weight calculation and convergence analysis of polyharmonic spline (PHS) with polynomials for different stencils. *Calcolo* **2024**, *61*, 22.
16. Tomita, H.; Tsugawa, M.; Satoh, M.; Goto, K. Shallow water model on a modified icosahedral geodesic grid by using spring dynamics. *J. Comput. Phys.* **2001**, *174*, 579–613.
17. Kolar-Poun, A.; Jančič, M.; Rot, M.; Kosec, G. Some observations regarding the RBF-FD approximation accuracy dependence on stencil size. *J. Comput. Sci.* **2024**, *79*, 102284.
18. Borne, S.L.; Leinen, W. Guidelines for RBF-FD discretization: Numerical experiments on the interplay of a multitude of parameter choices. *J. Sci. Comput.* **2023**, *95*, 8.
19. Flyer, N.; Fornberg, B.; Bayona, V.; Barnett, G.A. On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy. *J. Comput. Phys.* **2016**, *321*, 21–38.
20. Bayona, V. An insight into RBF-FD approximations augmented with polynomials. *Comput. Math. Appl.* **2019**, *77*, 2337–2353.
21. Subich, C.J. Higher-order finite volume differential operators with selective upwinding on the icosahedral spherical grid. *J. Comput. Phys.* **2018**, *368*, 21–46.
22. Swinbank, R.; Purser, R.J. Fibonacci grids: A novel approach to global modelling. *Q. J. R. Meteorol. Soc.* **2010**, *132*, 1769–1793.
23. Hardin, D.P.; Saff, E.B. Discretizing manifolds via minimum energy points. *Not. Am. Math. Soc.* **2004**, *51*, 1186–1194.
24. Available online: <http://web.maths.unsw.edu.au/~rsw/Sphere/Extremal/New/index.html> (accessed on 8 September 2026).
25. Available online: <http://web.maths.unsw.edu.au/~rsw/Sphere/EffSphDes/> (accessed on 8 September 2026).

26. Gorski, K.M.; Hivon, E.; Banday, A.J.; Wandelt, B.D.; Hansen, F.K.; Reinecke, M.; Bartelmann, M. HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *Astrophys. J.* **2004**, *622*, 759–771.
27. Shankar, V.; Wright, G.B.; Kirby, R.M.; Fogelson, A.L. Hyperviscosity-Based Stabilization for Radial Basis Function-Finite Difference (RBF-FD) Discretizations of Advection–Diffusion Equations. *J. Comput. Phys.* **2018**, *372*, 616–639.
28. Luan, V.T.; Pudykiewicz, J.; Reynolds, D.R. Further development of efficient and accurate time integration schemes for meteorological models. *J. Comput. Phys.* **2019**, *376*, 817–837.
29. Butcher, J.C. *Numerical Methods for Ordinary Differential Equations*, 2nd ed.; John Wiley & Sons: Chichester, UK, 2008.
30. Luan, V.T.; Ostermann, A. Parallel exponential Rosenbrock methods. *Comput. Math. Appl.* **2016**, *71*, 1137–1150.
31. Williamson, D.L.; Drake, J.B.; Hack, J.J.; Jacob, R.; Swarztrauber, P.N. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *J. Comput. Phys.* **1992**, *101*, 227–228.
32. Galewsky, J.; Scott, R.K.; Polvani, L.M. An initial-value problem for testing numerical models of the global shallow-water equations. *Tellus A* **2004**, *56*, 429–440.
33. Shashkin, V.; Goyman, G.; Tolstykh, M. Summation-by-parts finite-difference shallow water model on the cubed-sphere grid. Part I: Non-staggered grid. *J. Comput. Phys.* **2023**, *474*, 111797.
34. Gaudreault, S.; Charron, M.; Dallerit, V.; Tokman, M. High-order numerical solutions to the shallow-water equations on the rotated cubed-sphere grid. *J. Comput. Phys.* **2022**, *449*, 110792.
35. Zhou, L.; Xue, W.; Shen, X. HOPE: An arbitrary-order non-oscillatory finite-volume shallow water dynamical core with automatic differentiation. *Geosci. Model Dev.* **2025**, *18*, 8175–8201.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.