9-10-2007

Transport Schemes on a Sphere Using Radial Basis Functions

Natasha Flyer
National Center for Atmospheric Research

Grady Wright
Boise State University

Transport Schemes on a Sphere Using Radial Basis Functions

Natasha Flyer [a,*,1], Grady B. Wright [b,2],

[a] *National Center for Atmospheric Research, Institute for Mathematics Applied to the Geosciences, 1850 Table Mesa Dr., Boulder, CO 80305 USA*

[b] *Department of Mathematics, Boise State University, Boise, Idaho 83725-1555*

Abstract

The aim of this work is to introduce the physics community to the high performance of radial basis functions (RBFs) compared to other spectral methods for modeling transport (pure advection) and to provide the first known application of the RBF methodology to hyperbolic partial differential equations on a sphere. First, it is shown that even when the advective operator is posed in spherical coordinates (thus having singularities at the poles), the RBF formulation of it is completely singularity-free. Then, two classical test cases are conducted: 1) linear advection, where the initial condition is simply transported around the sphere and 2) deformational flow (idealized cyclogenesis), where an angular velocity is applied to the initial condition, spinning it up around an axis of rotation. The results show that RBFs allow for a much lower spatial resolution (i.e. lower number of nodes) while being able to take unusually large time-steps to achieve the same accuracy as compared to other commonly used spectral methods on a sphere such as spherical harmonics, double Fourier series, and spectral element methods. Furthermore, RBFs are algorithmically much simpler to program.

Key words: Radial Basis Functions, Hyperbolic PDEs, Spherical Geometry, Mesh-Free
1991 MSC: 58J45, 41A21, 41A15, 41A63, 65D25, 65D30, 65M20, 65M70, 76M22, 86A10

* Corresponding author.
Email addresses: flyer@ucar.edu, wright@boisestate.edu (Grady B. Wright).

[1] The work was supported by ATM-0620100.
[2] The work was supported by NSF VIGRE grant DMS-0091675 and ATM-0620090.

1 Introduction

The purpose of this paper is to introduce the prospect of using radial basis functions (RBFs), a novel numerical methodology that does not require any mesh or grid, for geophysical modeling in spherical domains. It has the advantage of achieving spectral accuracy in multi-dimensions for arbitrary node layouts with extreme algorithmic simplicity. For the purposes of interpolating multi-dimensional surfaces, the methodology has been around for approximately 30 years. However, it is only in the last 15 years that it has been applied to solving mixed partial differential equations (PDEs) containing parabolic and/or elliptic operators (cf. [1–6]). It has furthermore only been considered for PDEs in spherical domains for these same operators in the last 5 years [7,8]. Thus, the aim of this article is two-fold: 1) to present the elegance and power of this methodology to the physics community and 2) to provide the first known application of it to purely hyperbolic PDEs in spherical domains.

Since geophysical fluid motions on all scales are dominated by the advection process, the numerical solution to the advection problem is therefore fundamentally important for the overall accuracy of the flow solver. Thus, in the current paper, we will show striking results for advection tests with respect to accuracy and time stability, requiring only a low number of degrees of freedom for spatial discretization while being able to take much larger time-steps than methods currently employed in geophysical modeling (e.g. spherical harmonics, spectral elements). We consider two well-known test cases that probe the suitability of a new numerical methodology for modeling advection in spherical geometries. The first is the classical advection of a cosine bell with compact support over a sphere at different angles of rotation [9]. The second is a cyclogenesis test problem with a deformational flow that describes the wrap-up of a vortex with increasingly stronger gradients over time, which is a simple model for the observed evolution of cold and warm frontal zones [10]. An overview of the paper is as follows: Section 2 gives an introduction to RBFs; Section 3 discusses node distributions on a sphere and the convergence rates of RBF interpolants; Section 4 derives the RBF formulation of the advection operator; Section 5 and 6 are the numerical tests and results for convergence and time stability.

2 Introduction to Radial Basis Functions

The motivation of the RBF methodology originated with R.L. Hardy [11] asking the question, 'Given a set of sparse *scattered* data, $\{f_j\}_{j=1}^N$, at the node locations $\{\underline{x}_j\}_{j=1}^N$ in multi-dimensions, can an interpolant be constructed that adequately represents the unknown surface?'. It was first shown by Mairhuber

[12] that, in more than one dimension, interpolation by an expansion in basis functions, $\{\psi_j(\underline{x})\}_{j=1}^N$, $\underline{x} \in \mathbb{R}^d$, that are independent of the node locations is not well-posed. That is, there exists an infinite number of node configurations that will yield a singular interpolation problem . Hardy bypassed this singularity problem with a novel approach in which the interpolant is constructed from linear combinations of a *single* basis function that is radially symmetric about its center and whose argument is *dependent* on the node locations. In lieu of giving up orthogonality, well-posedness of the interpolant and its derivatives for *any* set of distinct scattered nodes in *any* dimension is gained.

Commonly used RBFs are given in Figure 1, where $r = \|\underline{x} - \underline{x}_j\|$ is the Euclidean or ℓ_2 norm. The piecewise smooth RBFs feature a jump in some derivative at $\underline{x} = \underline{x}_j$ and thus can only lead to algebraic convergence. For instance, the radial cubic $|r|^3$ has a jump in the third derivative, leading to fourth order convergence in 1-D, with the order of convergence increasing as the dimension increases (c.f.[13]). On the other hand, the evidence strongly suggests that infinitely smooth RBFs will lead to spectral convergence [14,15]. Notice that the infinitely smooth RBFs depend on a shape parameter ε. It was first shown by Driscoll and Fornberg [16] that, in 1-D, in the limit of $\varepsilon \to 0$ (i.e. flat RBFs) the RBF methodology reproduces pseudospectral methods (PS) if the nodes are accordingly placed (i.e. equispaced nodes for Fourier methods, Gauss-Chebyshev nodes for Chebyshev methods, etc.).

A comparison between the concept of PS methods and RBFs with differing values of the shape parameter is given in Figure 2. First, a PS expansion is always along a given coordinate direction, making them inherently 1-D objects (with expansions in higher dimensions represented by a tensor product with the respective 1-D basis expansions). By contrast, the scalar argument r of the RBF does not depend on a coordinate system, but is simply the distance between two nodes that are defined in d-dimensional space. Secondly, PS methods approximate a function by linear combinations of orthogonal functions that become more oscillatory as the degree of the polynomial increases, resulting in very clearly linearly independent functions as can be seen in Figure 2. By contrast, RBFs approximate a function with an expansion of one radially symmetric function whose only variation is the node location at which it is centered. While increasing the order of the polynomial expansion improves accuracy for PS approximations, accuracy of an RBF approximation can be improved by increasing the number of terms in the expansion and/or decreasing the shape parameter ε [4]. In either case, the shifted RBFs in the expansion become indistinguishable from one another as can clearly be seen in the case for $\varepsilon = 0.01$ in Figure 2, which leads to ill-conditioning. However, even for a moderate number of terms in the expansion and values of $\varepsilon \sim O(1)$, the evidence in the paper strongly suggests that RBFs have the potential of outperforming other methods that require much higher spatial and temporal resolution to achieve the same accuracy and which are much more algorithmi-

cally complex.

It should be noted that the Contour-Padé algorithm [17] can be used to bypass the RBF ill-conditioning mentioned above for the case of a fixed (relatively small) number of terms and increasingly small values of ε (even $\varepsilon = 0$). Furthermore, Fornberg and Piret [18] have recently discovered an algorithm for bypassing the ill-conditioning for RBF interpolation on the surface of the sphere both as the number of terms is increased and ε is decreased right to zero. We will, however, not pursue these algorithms in this study.

Although we will be solving hyperbolic PDEs, a good way to introduce the RBF methodology is through interpolation since at each time-step (in the explicit scheme) the exact spatial derivative operator is applied to the RBF interpolant to arrive at the derivative of the function at the node points. As mentioned above, RBFs approximate a function $f(x)$ sampled at some set of N distinct node locations by translates of a single radially symmetric function $\phi(r)$. For example, given the nodes $\{\underline{x}_j\}_{j=1}^N$ and corresponding scalar function values $\{f_j\}_{j=1}^N$, the RBF interpolant $s(\underline{x})$ to the data is defined by

$$s(\underline{x}) = \sum_{j=1}^{N} c_j \, \phi(\|\underline{x} - \underline{x}_j\|), \tag{1}$$

where the expansion coefficients, $\{c_j\}_{j=1}^N$, are found by enforcing the collocation conditions such that the residual is zero at the data locations. This is equivalent to solving the symmetric linear system of equations

$$\underbrace{\begin{bmatrix} \phi(\|\underline{x}_1 - \underline{x}_1\|) & \phi(\|\underline{x}_1 - \underline{x}_2\|) & \cdots & \phi(\|\underline{x}_1 - \underline{x}_N\|) \\ \phi(\|\underline{x}_2 - \underline{x}_1\|) & \phi(\|\underline{x}_2 - \underline{x}_2\|) & \cdots & \phi(\|\underline{x}_2 - \underline{x}_N\|) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\|\underline{x}_N - \underline{x}_1\|) & \phi(\|\underline{x}_N - \underline{x}_2\|) & \cdots & \phi(\|\underline{x}_N - \underline{x}_N\|) \end{bmatrix}}_{A} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}, \tag{2}$$

where A is the interpolation matrix. The concept of RBF collocation is illustrated in Figure 3 and 4 for 1-D and 2-D, respectively. For RBFs such as the GA, IMQ, and IQ, (2) is positive definite regardless of the distinct node locations and the dimension. For complete details on the well-posedness of (2) for all the RBFs listed in Figure 1, see [19, Ch. 12–16], for example.

To obtain a discrete RBF derivative operator, the exact differential operator is applied to the interpolant (1) and then evaluated at the data locations. As a result, we will first comment on the node distribution and convergence of RBF interpolants on the sphere before discussing the derivation of the RBF derivative operator in section 4.

3 Node Distribution and Convergence of RBF Interpolants

Since RBFs only depend on the scalar distance between nodes and not on a grid, the basis functions are not linked to any geometry or dimension. In other words, there is nothing inherently built into the RBFs to shout out "spherical geometry". In fact, they are unaware of the poles inherent in the spherical coordinate system. Studies have shown that if the shape parameter, ε, is kept fixed throughout the domain (as will be done in the current study—variable shape parameter is needed when implementing local mesh refinement [20,6,21]) best results are achieved with roughly evenly distributed nodes [22]. Since only a maximum of 20 nodes can be evenly distributed on a sphere, there are a multitude of algorithms to define "even" distribution for larger numbers of nodes, such as equal partitioned area, convex hull approaches, electrostatic repulsion, etc. [23]. Although any of these will suffice, we have decided to use an electrostatic repulsion approach since the nodes do not line up along any vertices or lines, emphasizing the arbitrary node layout and coordinate-free nature of a RBF methodology. A multitude of different size node sets is readily available at the website [24].

Assuming $\{\underline{x}_j\}_{j=1}^N$ are N nodes on the unit sphere, this approach, also known as the minimum energy (ME) point distribution on the sphere $\mathbb{S}^2$, provides a quasi-uniform distribution on the sphere by maximizing the minimum distance between nodes according to the measure

$$h = \max_{\underline{x}\in\mathbb{S}^2} \min_{1\le i\le N} \text{dist}(\underline{x}, \underline{x}_i), \tag{3}$$

where dist is the geodesic distance from $\underline{x}$ to $\underline{x}_i$. This quantity is referred to as the mesh-norm [24,25] and, geometrically, it represents the radius of the largest cap that covers the area between any subset of nodes on the sphere. The ME node sets have the property that h decays approximately uniformly like the inverse of the square root of the number of nodes N, i.e.

$$h \sim \frac{1}{\sqrt{N}}.$$

Thus, they are similar to a uniform discretization of the unit square. In Figure 5, the distribution for 1849 nodes on the unit sphere is displayed.

The mesh norm is also of practical importance since it appears in many proofs of error bounds for RBF interpolation on the sphere (e.g. [25,26]). Indeed, in the context of infinitely smooth RBFs, it is shown in [25] that, provided the underlying function being interpolated is sufficiently smooth, RBF interpolants converge (in the L^∞ norm) like $h^{-1/2}e^{-c/4h}$, i.e at an exponential rate, for some constant $c > 0$ that depends on the RBF. For the ME node sets, convergence will thus proceed like $N^{1/4}e^{-c\sqrt{N}/4}$. In the experiments that fol-

low we will demonstrate that this error bound seems to also hold for the RBF method-of-lines approximation of the two test cases.

4 RBF formulation of the advection operator

Physical phenomena are naturally not associated with any coordinate system. However, scientists impose coordinate systems to formulate their PDEs. In spherical geometries, this results in the spatial operator being singular at the poles. For example, the gradient on the surface of a unit sphere in spherical coordinates (λ is longitude, θ is latitude and measured from the equator) is given by

$$\nabla = \frac{1}{\cos\theta}\frac{\partial}{\partial\lambda}\hat{\boldsymbol{\lambda}} + \frac{\partial}{\partial\theta}\hat{\boldsymbol{\theta}}, \tag{4}$$

which is singular at $\theta = \pm\frac{\pi}{2}$, the north and south pole, respectively. Since RBFs depend only on the Euclidean distance between nodes, the basis functions are thus not associated with any coordinate system and therefore do not "feel" the effects of the geometry of the domain. As a result, RBFs do not recognize the singularities naturally inherent in the coordinate system and all remnants of such singularities vanish when the methodology is implemented. This allows the scientist to directly connect the physics to the numerics. Since the test cases that are considered both use the advection operator, we will now see how it becomes nonsingular when formulated with RBFs.

Let $\underline{x} = (x, y, z)$, $\underline{x}_j = (x_j, y_j, z_j)$ be two points on the surface of the unit sphere and (λ, θ), (λ_j, θ_j) the corresponding spherical coordinates, i.e.

$$\begin{aligned} x &= \cos\lambda\cos\theta, \\ y &= \sin\lambda\cos\theta, \\ z &= \sin\theta. \end{aligned} \tag{5}$$

(This differs from traditional spherical coordinates in that we measure θ from the equator rather than from the north pole). Then, the Euclidean distance from $\underline{x}$ to $\underline{x}_j$ is

$$\begin{aligned} r = \|\underline{x} - \underline{x}_j\| &= \sqrt{(x-x_j)^2 + (y-y_j)^2 + (z-z_j)^2} \\ &= \sqrt{2(1 - \cos\theta\cos\theta_j\cos(\lambda-\lambda_j) - \sin\theta\sin\theta_j)}. \end{aligned}$$

It is important to note that the distances are not great circle arcs measured along the surface but are the Euclidean distance measured straight through the sphere.

Let $\phi_j(r) = \phi(\|\underline{x} - \underline{x}_j\|)$ be an RBF centered at $\underline{x}_j$. Using the chain rule, the

partial derivatives of the RBF $\phi_j(r)$ with respect to λ and θ are given by

$$\frac{\partial}{\partial\lambda}\phi_j(r) = \cos\theta\cos\theta_j\sin(\lambda-\lambda_j)\left(\frac{1}{r}\frac{d\phi_j}{dr}\right), \tag{6}$$

$$\frac{\partial}{\partial\theta}\phi_j(r) = (\sin\theta\cos\theta_j\cos(\lambda-\lambda_j) - \cos\theta\sin\theta_j)\left(\frac{1}{r}\frac{d\phi_j}{dr}\right). \tag{7}$$

Note that (6) and (7) are well defined for $r = 0$ (i.e. $\underline{x} = \underline{x}_j$ since we are assuming $\phi_j(r)$ is C^∞ and radially symmetric about $\underline{x}_j$). Inserting (6) and (7) into (4), we have the action of the gradient operator on the RBF scalar function:

$$\begin{aligned}\nabla\phi_j(r) = \Big[&\cos\theta_j\sin(\lambda-\lambda_j)\hat{\boldsymbol{\lambda}}+ \\ &(\cos\theta_j\sin\theta\cos(\lambda-\lambda_j) - \sin\theta_j\cos\theta)\hat{\boldsymbol{\theta}}\Big]\left(\frac{1}{r}\frac{d\phi_j}{dr}\right).\end{aligned} \tag{8}$$

As mentioned at the beginning of this section, the gradient has a singularity in the λ direction at the poles unless the derivative (with respect to λ) of the underlying function also vanishes at the poles. We see from (8) that this is exactly what happens when using an RBF.

Now, we have all the components that are necessary to build the action of the advection operator on an RBF representation of a geophysical field. Suppose we want to advect some scalar quantity, say a given height field $h(\lambda,\theta)$, where the components of the advecting wind $\mathbf{U}$ are given by $\mathbf{U} = u(\lambda,\theta)\hat{\boldsymbol{\lambda}} + v(\lambda,\theta)\hat{\boldsymbol{\theta}}$. Let $\{\underline{x}_j\}_{j=1}^N = \{(\lambda_j,\theta_j)\}_{j=1}^N$ be the node locations where $h(\lambda,\theta)$ is known. We first represent $h(\lambda,\theta)$ as an RBF expansion given by

$$h(\lambda,\theta) = \sum_{j=1}^{N} c_j\phi_j(r) \tag{9}$$

where $\phi_j(r)$ is again the RBF centered at the node $\underline{x}_j = (\lambda_j,\theta_j)$. We then apply the exact differential operator

$$(\mathbf{U}\cdot\nabla) = \frac{u(\lambda,\theta)}{\cos\theta}\frac{\partial}{\partial\lambda} + v(\lambda,\theta)\frac{\partial}{\partial\theta}$$

to (9) and evaluate it at the node locations:

$$\begin{aligned}(\mathbf{U}\cdot\nabla)h(\lambda_i,\theta_i) &= \sum_{j=1}^{N} c_j \underbrace{\left[(\mathbf{U}\cdot\nabla)\phi_j(r)\right]\big|_{(\lambda,\theta)=(\lambda_i,\theta_i)}}_{\text{Components of } B} \qquad (i=1,\ldots,N)\\ &= B\boldsymbol{c}\\ &= (BA^{-1})\mathbf{h}\\ &= D_N\mathbf{h},\end{aligned} \tag{10}$$

where $\mathbf{h}$ contains the N discrete values of the height field h at the nodes, $\boldsymbol{c}$ contains the N discrete expansion coefficients and is formally given by $\boldsymbol{c} = A^{-1}\mathbf{h}$, where A^{-1} is the inverse of the RBF interpolation matrix defined in (2). The discrete operator $D_N = BA^{-1}$ is referred to as the RBF differentiation matrix, and the components of the matrix B are explicitly given by

$$\begin{aligned} B_{i,j} = &\{u(\lambda_i,\theta_i)\cos\theta_j \sin(\lambda_i-\lambda_j)+ \\ &v(\lambda_i,\theta_i)[\cos\theta_j \sin\theta_i \cos(\lambda_i-\lambda_j) - \sin\theta_j\cos\theta_i]\} \left(\frac{1}{r}\frac{d\phi}{dr}\right)\Bigg|_{r=\|\underline{x}_i-\underline{x}_j\|}, \end{aligned} \tag{11}$$

for $i,j = 1,\ldots,N$. Notice that (11) is nowhere singular on the sphere, remembering of course that the velocity field is completely smooth. Although the computation of D_N requires $O(N^3)$ operations, it is a pre-processing step that needs to be done only once.

5 Numerical Test Case 1: Solid Body Rotation

The first test case (solid body rotation or passive advection), using the setup given in [9], simulates the advection of a height field, $h(\lambda,\theta)$, over the surface of a sphere at an angle α relative to the pole of the standard longitude-latitude (λ-θ) grid (see Figure 6). The PDE to be solved is the advection equation, which in spherical coordinates is given by

$$\frac{\partial h}{\partial t} + \frac{u}{a\cos\theta}\frac{\partial h}{\partial \lambda} + \frac{v}{a}\frac{\partial h}{\partial \theta} = 0, \tag{12}$$

with the advecting wind being

$$u = u_0(\cos\theta\cos\alpha + \sin\theta\cos\lambda\sin\alpha), \tag{13}$$

$$v = -u_0\sin\lambda\sin\alpha, \tag{14}$$

where a is the radius of the earth, $6.37122\cdot 10^6$m and $u_0 = 2\pi a/(12 \text{ days} = 288 \text{ hours})$.

We will consider two initial conditions (which are also the solution for all time) that are to be advected without distortion by the above steady wind (13-14). As illustrated in Figure 7(a), the first is the classical test case in the literature [9], a cosine bell profile that is C^1 and centered at (λ_c,θ_c):

$$h(\lambda,\theta) = \begin{cases} \frac{h_0}{2}\left[1+\cos\left(\pi\frac{\rho}{R}\right)\right] & \rho < R, \\ 0 & \rho \geq R, \end{cases} \tag{15}$$

where $h_0 = 1000$m, $R = a/3$ and $\rho = a \arccos(\sin\theta_c \sin\theta + \cos\theta_c \cos\theta \cos(\lambda - \lambda_c))$. The second—illustrated in Figure 7(b)—is an exceptionally steep Gaussian profile that is C^∞:

$$h(\lambda, \theta) = h_0 e^{-(2.25\rho/R)^2}, \tag{16}$$

where h_0, ρ, and R are the same as (a). This profile is used to demonstrate that the RBF method is indeed spectral.

The center of the bell is initially taken to be at the equator, $(\lambda_c, \theta_c) = (0, 0)$. In testing previously used methods such as spherical harmonics, double Fourier series, or spectral element methods, the object is rotated at various angles α with regard to the polar axis of the spherical coordinate system, with rotation over the poles being the most severe test case. This is to see how the methodology handles the "pole singularity" inherent in a spherical coordinate system. Since RBFs and the node layout are free of any coordinate system, the error is invariant to the angle of rotation (see the Appendix for a rigorous proof). As a result, the choice of α is irrelevant. We choose $\alpha = \pi/2$ (i.e. flow right over the poles) only for comparison reasons to other methods, since it is the angle for which the error is most quoted in the literature.

The method-of-lines RBF formulation for (12) is given by

$$\frac{\partial \mathbf{h}}{\partial t} + D_N \mathbf{h} = 0, \tag{17}$$

where the differentiation matrix D_N represents the discretized advection operator derived in (10) of the previous section. The standard fourth-order Runge-Kutta scheme (RK4) is used to advance the solution in time and no filtering is applied. The code is given in Appendix B with several lines added for easier readability.

5.1 Comparative Results for Solid Body Rotation

We first consider the cosine bell test case using GA RBFs (c.f. Figure 1) with $\varepsilon = 8.2$, $N = 4096$ ME nodes, and a time-step of $\Delta t = 30$ minutes. The choice for these values is to obtain error norm results that are comparable to other methods used in numerical climate modeling, and yet point out the strength of the RBF method in terms of time stability and spatial resolution requirements.

Figure 8(a) displays a surface plot of the numerical solution after one full revolution around the sphere ($t = 12$ days). Comparing this to the true solution (also the initial condition) in Figure 7(a), we see that the numerical RBF

solution is visibly identical to the exact solution. Figure 8(b) displays an orthographic projection of the error in the RBF solution, showing all deviations in the solution of less than 1 meter (or an error of < 0.001) in white. The figure shows that the dominant error is a ring at the base of the bell where the function is only C^1. Furthermore, there is no evidence of any trailing dispersive wave trains. At the conclusion of this revolution, the ℓ_2 error is $6.18 \cdot 10^{-3}$ and the ℓ_∞ error is $2.27 \cdot 10^{-3}$. Figure 9 shows the error as a function of time for the 12 day simulation.

Table 1 compares the performance of different spectral methods used on the sphere: spherical harmonics (SH), double Fourier (DF), and a discontinuous Galerkin method (DG), a hybrid approach combining spectral elements and finite volume methods [27], developed for spherical geometry by [28]. This last method was chosen over a direct spectral element approach (as [29] or [30]) since it produced better results for this test case. A common basis of comparison for all the methods considered that could be found in the literature was the spatial and temporal specifications needed for each method to achieve an ℓ_2 error of approximately 0.005 [31,32,28]. Studying Table 1 there are five points that clearly stand out:

(1) The number of nodes needed for the RBF method is approximately half that needed for the DG method and 8 times less than needed for both SH and DF.

(2) The time-step taken for the RBF method is 5 times larger than the DG and 20 times larger than that taken for both SH and DF.

(3) The algorithmic complexity of implementing an RBF method is essentially trivial in comparison to other methods. Our code from start to finish is less than 40 lines of MATLAB, using *NO* MATLAB routines that need to be compiled (i.e we use only built-in MATLAB routines that are coded at the machine level as *, ', and /). The complexity of the code would not change if the dimension of the problem to be solved is increased, which can not be said for any other method.

(4) Only the DG and RBF method allow for local mesh refinement [20,6,21].

(5) The RBF method has the highest computational cost, requiring a matrix-vector multiply per time-step, where the matrix is full. However, fast algorithms are available that have the potential for reducing this cost to $O(N \log N)$ or possibly $O(N)$ (c.f. [33–35]).

In order to better understand these results, we will first perform a convergence study with regard to: (a) h-refinement, the number of nodes used N (remembering that $1/\sqrt{N}$ is proportional to the spacing of the node distributions)

and (b) ε-refinement, the shape parameter of the RBF. Then, we will do a stability and eigenvalue study with regard to these parameters, illustrating why such large time-steps can be taken. Our objective is to show trends in convergence and stability which can be illustrated with a much lower number of nodes. We choose $N = 1849$, the same number of degrees of freedom as the National Center for Atmospheric Research (NCAR) T42 spherical harmonic community climate model (i.e. spherical harmonics up to order 42), which is much quoted in the literature for comparison purposes [31,29,32].

5.2 Convergence Study

5.2.1 h-refinement

Figure 10(a) shows the convergence rates in both the ℓ_∞ and ℓ_2 norm for the cosine bell test case on a log-log plot. Although a time-step of $\Delta t = 30$ minutes was used for each node set in the RK4 integration, as the size of the node set decreases the maximum time step so that spatial errors dominate will increase, e.g. for $N = 1849$, a time step of 50 minutes could be used. Similar to all spectral methods, the RBF methodology results in spectral convergence if the initial condition is C^∞. However, in this case the cosine bell is only a C^1 function, which results in low-order algebraic convergence as illustrated in the figure by the dashed lines.

To demonstrate that the RBF method is indeed spectral, we instead advected the steep Gaussian bell (16) (c.f. Figure 7(b)) that is similar to the cosine bell, but is C^∞. Figure 10(b) (a log-linear plot) shows that indeed the convergence rates are spectral in both the ℓ_∞ and ℓ_2 norm with the error being $2(10)^{-7}$ for $N = 4096$.

It should be noted that there is nothing special about the node layout used. Similar error norms would be achieved if we were using equal-area node distributions or those laid out according to the golden ratio, as in a sunflower pattern. The only requirement is that the nodes are roughly equally distributed in some sense on the surface of the sphere, as close clustering can lead to ill-conditioning. In a similar vein, for the number of nodes used in the paper, $N \leq 4096$, ill-conditioning is not a problem and MATLAB's mrdivide command can be used to calculate D_N without hesitation for the range of ε used (see next subsection).

5.2.2 ε-refinement

While refining h (i.e. increasing N) results in more terms in the RBF expansion and a better resolution in the approximation, refining the shape param-

eter ε, as illustrated in the right column of Figure 2, results in a smoother basis for the RBF expansion. (We use the term smoother here not to mean the number of bounded derivatives, but that the growth of the derivatives is smaller.) Qualitatively, this means that refining ε allows the RBF expansion to better approximate smoother functions, however the phenomenon is not fully understood. It has generally been reported in the literature that there is typically an optimal value of ε and this value tends to decrease with increasing smoothness of the underlying function being approximated (cf. [4,36]). While a few algorithms have been developed for trying to determine the "optimal ε" (e.g. [36,37]), they are primarily based on heuristic arguments and are not robust. We therefore do not employ them in this study.

As noted in Section 2, as ε is refined more and more, the shifted RBFs in the expansion become less and less distinguishable from one another, leading to ill-conditioning of the linear system (2). While it has been shown that RBF interpolants are overall well-conditioned even in the limit of $\varepsilon \to 0$ refinement [16,38], special algorithms like Contour-Padé [17] and RBF-QR [18] are needed for these smaller values. However, as discussed next, we are able to obtain very good results for our test cases even with moderate values of ε that are well outside the ill-conditioning range.

To test the accepted results on ε-refinement we fixed the number of nodes in the RBF expansion at $N = 1849$ and performed a test on the error as a function of ε for the cosine and Gaussian bell initial conditions with the GA RBF and a time step $\Delta t = 50$ minutes. The results are shown in Figures 11(a) and 11(b), respectively. We can clearly see from the figures that there is an optimal value of ε for which the error is the smallest, and that, for the rougher C^1 cosine bell initial condition, the optimal ε is larger than the C^∞ Gaussian bell ($\varepsilon \approx 6$ and $\varepsilon \approx 3$, respectively). The increase in error after the optimal ε is not due to ill-conditioning but is a property of the function being approximated (see [20]).

5.3 Eigenvalue Stability

Since the RBF differentiation matrices are not normal, classical eigenvalue stability theory is theoretically insufficient. However, in practice, as with many pseudospectral methods, it is still a very good predictor for the maximum stable time-step as we shall see below.

Let us first examine why such high accuracy can be achieved with a large time step compared to other methods as reported in the previous section. If the eigenvalues of the differentiation matrix D_N for $N = 1849$ are plotted as in Figure 12, we see that they lie exactly on the imaginary axis. Not one

eigenvalue lies in the right half plane, which would eventually lead to numerical instability, nor in the left half plane, leading to dissipation. This result follows from the fact that $D_N = BA^{-1}$, the product of an anti-symmteric matrix B given by (11) and a positive definite matrix A given by (2). Although the product of the two is not antisymmetric, it preserves the property of antisymmetry that all eigenvalues lie on the imaginary axis [39].

The size of the maximum time step to maintain stability depends on the eigenvalues of D_N fitting within the stability domain of the RK4 method. The maximum eigenvalue of D_N, in turn, depends on the two parameters N and ε as shown in Figure 13. As the accuracy of the method increases, either by increasing N (h-refinement) or decreasing ε (ε-refinement), the maximum eigenvalue will increase, implying that the maximum time step that can be taken must decrease. Note, however, that changes in these two variables have different impacts on the maximum eigenvalue (or allowable time step). The maximum eigenvalue linearly increases with spatial resolution up/down the imaginary axis, which is the classical result for linear hyperbolic PDEs. In contrast, decreasing ε changes the maximum eigenvalue in a manner that is reminiscent of what happens to finite difference approximations to derivative operators as the order increases for fixed N. For example, increasing the accuracy of the first derivative operator from second order centered finite differences to Fourier PS, increases the maximum eigenvalue by π ([40, pp. 41–42]). In a similar manner, as shown in Figure 13(b), the maximum eigenvalue increases by 4 as ε varies from 9 to zero. However, the reader should be reminded that the RBF method is spectral for all values of ε. It is just that as ε decreases to some (typically non-zero) optimal value, the accuracy improves (e.g. in the case of the cosine bell it is ε=6, for the Gaussian bell it is ε=3). Furthermore, even though RBFs reproduce classical PS methods in limit of $\varepsilon = 0$ [16,18,41], it is for nonzero values of the parameter that RBFs outperform PS as was shown in the comparative study of the previous section.

The results shown graphically in Figure 13(a) and (b) are also given numerically in Table 2 and Table 3, where the maximum eigenvalue has also been translated into the maximum allowable time step for the RK4 integrator (i.e such that the eigenvalue falls with the RK4 stability domain). For the case $N = 1849$ or $\varepsilon = 6$, the theory predicts that the maximum time step is 208 minutes (3 hours 28 minutes). To test this, we plot in Figure 14 the time evolution of the ℓ_∞ error for the cosine bell test case using $\varepsilon = 6$ and $N = 1849$ with a time-step of 208 minutes and 209 minutes in the RK4 integrator. After 36 days of integration (3 full revolutions around the sphere), the test run using a 208 minute time step is completely stable while the use of a 209 minute time step has caused numerical instability to set in at approximately 20 days, verifying the predicted results from classical eigenvalue stability theory. However, for best computational efficiency one should use time steps so that temporal and spatial errors match. Such is illustrated in Figure 15 which shows that for

$\varepsilon = 6$ and $N = 1849$ the ℓ_∞ error of the solution after a 12 day single revolution around the earth is steady up to a 50 minute time step. Larger time steps, although stable as just demonstrated, will cause discretization errors in time to dominate over those in space. For the comparative study of the previous section with $N = 4096$, this breaking point was a 30 minute time step.

6 Numerical Test Case 2: Deformational flow

The second test case involves no translational motion, but instead an angular velocity field spins up the initial condition, resulting in two diametrically opposed vortices. This test for idealized cyclogenesis on the sphere was first described by Nair et al. [10]. Let (λ', θ') be the rotated coordinate system with north pole at (λ_p, θ_p) with respect to the regular spherical coordinate system (λ, θ). In these rotated coordinates, the PDE to simulate is given by

$$\frac{\partial h}{\partial t} + \frac{u'}{\cos\theta'}\frac{\partial h}{\partial \lambda'} = 0 \tag{18}$$

where u' is the tangential velocity field in the rotated coordinates and is given by

$$u' = \omega(\theta')\cos\theta'.$$

Note that since the vortices are steady there is no velocity in the normal direction v' of the rotated coordinate system. The angular velocity ω of the vortex field is given by

$$\omega(\theta') = \begin{cases} \frac{3\sqrt{3}}{2\rho(\theta')}\mathrm{sech}^2(\rho(\theta'))\tanh(\rho(\theta')) & \text{if} \quad \rho(\theta') \neq 0 \\ 0 & \text{if} \quad \rho(\theta') = 0, \end{cases}$$

where $\rho(\theta') = \rho_0 \cos\theta'$ is the radial distance of the vortex. The exact solution in non-dimensional units at time t is given by

$$h(\lambda', \theta', t) = 1 - \tanh\left(\frac{\rho(\theta')}{\gamma}\sin\left(\lambda' - \omega(\theta')t\right)\right), \tag{19}$$

where γ is a parameter defining the characteristic width of the frontal zone. To be consistent with [28], we set $\rho_0 = 3$ and $\gamma = 5$. The initial condition, $h(\lambda', \theta', 0)$, with these parameters is displayed on an unrolled sphere in Figure 16.

Upon transformation to an unrotated (λ, θ) spherical coordinate system (18) becomes [10]

$$\frac{\partial h}{\partial t} + \frac{u}{\cos\theta}\frac{\partial h}{\partial \lambda} + v\frac{\partial h}{\partial \theta} = 0,$$

where

$$\begin{aligned} u =& \omega(\theta')(\sin\theta_p\cos\theta - \cos\theta_p\cos(\lambda-\lambda_p)\sin\theta), \\ v =& \omega(\theta')\cos\theta_p\sin(\lambda-\lambda_p). \end{aligned}$$

This form is much more complicated than the original since we have now introduced flow in the normal direction to the (λ,θ) grid.

However, since the RBF method is completely independent of how the underlying coordinate system is oriented, we can simply simulate (18) in its rotated (λ',θ') form. To this end, let $\underline{x}'_i = (\lambda'_i,\theta'_i)$, $i = 1,2,\ldots,N$, be the original nodes on the standard (λ,θ) grid rotated to the new coordinate system (λ',θ'), i.e.

$$\begin{aligned} \lambda'_i &= \arctan\left(\frac{\sin(\lambda_i-\lambda_p)}{\sin\lambda_p\cos(\lambda_i-\lambda_p) - \cos\theta_p\tan\theta_i}\right), \\ \theta'_i &= \arcsin\left(\sin\theta_i\sin\theta_p + \cos\theta\cos\theta_p\cos(\lambda_i-\lambda_p)\right). \end{aligned}$$

The method-of-lines RBF formulation of (18) is given by

$$\frac{\partial \mathbf{h}}{\partial t} = -WD_N\mathbf{h}, \tag{20}$$

where $\mathbf{h}$ contains the N discrete value of h at the rotated nodes, $W = \mathrm{diag}(\omega(\theta'_i))$, $i = 1,\ldots,N$, and the differentiation matrix $D_N = BA^{-1}$, where A is given by (2) with $\underline{x}_i, \underline{x}_j$ replaced with $\underline{x}'_i$, $\underline{x}'_j$ and

$$B_{i,j} = \cos\theta'_i\cos\theta'_j\sin(\lambda'_i-\lambda'_j)\left(\frac{1}{r}\frac{d\phi}{dr}\right)\Big|_{r=\|\underline{x}'_i-\underline{x}'_j\|}, \; i,j = 1,\ldots,N.$$

As in the solid body rotation problem, the RBF formulation is completely free of any coordinate singularities.

6.1 Results for Deformational Flow

The analysis for deformational flow will be similar to that of solid body rotation. We will perform a convergence and eigenvalue/stability study to understand the high performance of the RBF method. However, since this is a new test that is performed in the numerical climate modeling community, the only results in the spectral methods literature that have been published is for the DG method [28] and thus comparison is limited to this method.

6.1.1 Convergence study

Figure 17 shows the initial condition, exact solution and GA RBF solution for $N = 3136$ and $\varepsilon = 6.45$ at the final time $t = 3$ for two different grey scale

mappings: the left column is simply the initial condition displayed in Figure 16 projected onto the surface of the sphere, the right column highlights the band corresponding to the plane $h = 1$ in Figure 16 about which the vortex wrap-up occurs. As can be seen, the exact and RBF solutions are indistinguishable to the eye. Figure 18 displays the error (exact - numerical). All errors that are less than 10^{-4} in magnitude are displayed in white. The maximum errors are located near the center of the vortex wrap-up where the solution is the most highly oscillatory.

Figure 19 plots the ℓ_1 and ℓ_2 error as a function of the grid spacing $\sqrt{N}$ on log-linear plot, showing that the method converges exponentially (i.e. spectral). The sets of nodes used is the same as in the cosine bell test case, ranging from 526 to 4096. For $N = 3136$, the ℓ_1 and ℓ_2 errors are $1 \cdot 10^{-5}$ and $5 \cdot 10^{-5}$, respectively. These results are an order of magnitude lower than those reported in [28] for the DG method using about the same number of nodes ($N = 3456$) with ℓ_1 and ℓ_2 errors being approximately $1 \cdot 10^{-4}$ and $6 \cdot 10^{-4}$, respectively. Furthermore, if we increase the number of nodes for the RBF method by 930 to $N = 4096$, the ℓ_1 and ℓ_2 errors drop to approximately $3 \cdot 10^{-6}$ and $1.5 \cdot 10^{-5}$, respectively. To achieve the same accuracy in the ℓ_1 error, the DG method requires almost double the number of nodes at $N = 7776$ and in the ℓ_2 error it requires 14 times the number of nodes at $N = 55296$ (here, the number of elements has been fixed while increasing the order of the polynomial expansion on each element, p refinement, which is a better way of increasing accuracy than h refinement). In other words, while the RBF method is showing spectral convergence, the DG method is showing high-order algebraic convergence in the ℓ_1 norm and very low-order algebraic convergence in the ℓ_2 norm. More surprisingly, however, is the incredible time stability that the RBF method exhibits compared to the DG method as discussed next.

6.1.2 Eigenvalue stability

In the above convergence study 30 time steps were taken to go from $t = 0$ to the final time $t = 3$, i.e. $\Delta t = 1/10$. This is a factor of 48 less than reported for the DG method which used 1440 time steps due to its CFL condition. In fact for this test case, as few as 13 time steps ($\Delta t = 3/13$) could be taken with the ℓ_1 and ℓ_2 error norms remaining unchanged as seen in Figure 20. This is a reduction of *110 times* compared to what is needed for the DG method. However, it is noted in [27] that the DG method has a very restrictive time step. To understand these results we need to take a look at the eigenvalues of the RBF differentiation matrix $-WD_N$ that will govern the stability of the semi-discrete RBF approximation (20).

Figures 21(a)–(c) display the eigenvalues of $-WD_N$ for the GA RBF with $N = 3136$ and $\varepsilon = 6.45$ superimposed on the stability domain of RK4 [40,

pp.197–200] and scaled by the time steps $\Delta t = 3/14, 3/15, 3/18$, respectively. Since the spectrum is no longer purely imaginary but tightly packed about the imaginary axis, instability can be manifested in different manners. Hence, each of these diagrams shows a distinct feature. The test case ($t_{final} = 3$) can easily be performed with a time step of $3/14$ even though there are eigenvalues on the imaginary axis that lie just outside the RK4 stability domain as shown in Figure 21(a). However, this will soon lead to temporal instability as seen in Figure 22(a) where the ℓ_1 and ℓ_2 error are plotted as a function of time for these time steps. In the case of $\Delta t = 3/14$, instability occurs at about t=24, a time integration 8 times longer than the test case stated in the literature [28]. In Figure 21(c) ($\Delta t = 3/18$), no eigenvalues on the imaginary axis are outside the RK4 stability region but the two eigenvalues with the largest positive real parts lie just outside this domain. However, these eigenvalues are associated with highly oscillatory eigenmodes and will not come into play until the solution exhibits such behavior. Figure 22(b) displays the exact solution at $t = 35$, the time at which the RBF approximation with $N = 3136$ and $\Delta t = 3/18$ breaks down both in time and space. From a temporal standpoint, the eigenmodes with the largest positive real parts are now triggered since the solution is very highly oscillatory, leading to exponential blow-up. From a viewpoint of spatial resolution, there really are no adverse effects of the positive real parts of the eigenvalues until the solution features have become too fine to be resolvable by the current number of nodes (near the center of the vortex wrap-up there is only one node per wavelength which is less than the needed theoretical limit of 2 nodes per wave length). As shown in Figure 21(b), with $\Delta t = 3/15$, the two eigenvalues with the largest real parts now lie just inside the RK4 stability domain. This results in a longer stable time integration as seen in Figure 22(a).

7 Summary

The main goal of this paper is to illustrate the effectiveness and performance of the RBF methodology for solving purely hyperbolic PDEs on a sphere, using test cases in the numerical climate modeling community given by solid body rotation and deformational flow. The detailed results of the previous sections can be simply summed up as follows:

(1) For these test cases, the RBF methodology outperforms all currently used spectral methods in terms of the number of nodes and time-step needed as well as algorithmic simplicity to achieve a given accuracy.
   (a) In either test case, the code is very short and simple - less than 40 lines of MATLAB, using only built-in MATLAB routines that are coded at the machine level such as *, ', and /.
   (b) Eigenvalues of the differentiation matrices are either purely imaginary (solid body) or have very small real parts (deformational flow).

The absence of spurious eigenvalues of large amplitude give the ability to take exceptionally large time-steps.

(c) The method is spectrally accurate for smooth initial conditions, requiring a much lower number of nodes compared to other commonly used spectral methods on a sphere.

(2) The differentiation matrices are entirely free of any "pole singularities" and invariant of the orientation of the original coordinate system.

(3) The RBF methodology connects the mathematics directly to the physics without the interference of a surface-based grid.

Furthermore, algorithmic complexity does not increase with the dimension of the problem, since RBFs only depend on the Euclidean distance between nodes which is always a scalar independent of dimension (e.g. if linear advection were posed in 3D, the code used in 2D for the solid body rotation test case would not change in length or complexity). These results illustrate that RBFs can provide a promising new approach to modeling in spherical geometries.

Appendix A: Rotational invariance of D_N for solid body rotation

Although we used a spherical coordinate system which was measured from the equator, we will now show how the RBF derivative matrix D_N for the spatial derivatives of (12) is rotationally invariant, i.e. completely independent of how the original spherical coordinate system was oriented in space. Thus, when implementing the advective operator with the wind specified by (13)–(14) using RBFs in spherical coordinates, the result is not only singularity free, but is also completely independent of the spherical coordinate orientation.

An equivalent demonstration is to keep the spherical coordinate system fixed, and instead turn (in unison) both the axis of solid body rotation and all the point locations by an angle α. Let $C_{i,j}$ be the expression inside the curly brackets for $B_{i,j}$ in (11) with u and v given by (13) and (14), respectively, and $u_0 = 1$ for simplicity. If $C_{i,j}$ remains invariant with α, then we are finished.

Each entry $C_{i,j}$ involves only two node points, located at (λ_i, θ_i) and (λ_j, θ_j). After the axis of solid body rotation is turned by an angle α, we obtain the new coordinates (λ', θ') for a point (λ, θ) by the transformation

$$\cos\lambda' \cos\theta' = \cos\lambda \cos\theta \cos\alpha - \sin\theta \sin\alpha \,, \tag{21}$$

$$\sin\theta' = \cos\lambda \cos\theta \sin\alpha + \sin\theta \cos\alpha \,. \tag{22}$$

The goal is to show that

$$\begin{aligned} C_{i,j} = &\cos\alpha \cos\theta_i' \cos\theta_j' \sin(\lambda_i' - \lambda_j') + \\ &\sin\alpha[\sin\lambda_i' \cos\theta_i' \sin\theta_j' - \sin\lambda_j' \cos\theta_j' \sin\theta_i'] \end{aligned} \tag{23}$$

depends only on (λ_i, θ_i) and (λ_j, θ_j) (i.e. that it is independent of α).

The analysis turns out to be fairly simple if we consider the node points in Cartesian coordinates. So, we let the points (λ_i', θ_i') and (λ_j', θ_j') corresponds to the points $\underline{x}_i' = (x_i', y_i', z_i')$ and $\underline{x}_j' = (x_j', y_j', z_j')$, respectively. Using the relationship between Cartesian and spherical coordinates (5), we can rewrite (23) as follows:

$$C_{i,j} = (x_j' y_i' - x_i' y_j') \cos\alpha + (y_i' z_j' - y_j' z_i') \sin\alpha \tag{24}$$

Now, the counterpart to (21) and (22) in Cartesian coordinates is given by the transformation

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} .$$

Using this transformation we can relate the rotated points in (24) to the original points. After some algebra, we find the simple expression

$$\begin{aligned} C_{i,j} &= x_j y_i - x_i y_j \\ &= \cos\theta_i \cos\theta_j \sin(\lambda_i - \lambda_j) . \end{aligned}$$

Thus, $C_{i,j}$ is completely independent of the angle α, which shows the RBF methodology for (12) is completely independent of how the original spherical coordinate system was oriented in space.

Appendix B: Code for solid body rotation with cosine bell IC

```
    ep=6;                % Value of epsilon to used
    alpha = pi/2;        % Angle of rotation measured from the equator
    a = 6.37122e6;       % Mean radius of the earth (meters)
    u0 = 2*pi*a/12;      % Speed of rotation(m/day)-one full revolution in 12 days
    R = a/3;             % Width of bell

%%% Load Nodes: http://web.maths.unsw.edu.au/~rsw/Sphere/Energy/index.html %%%
    load('me1849.dat'); x = me1849(:,1); y = me1849(:,2); z = me1849(:,3);

%%% Compute r^2 = (x_j - x_k)^2 + (y_j - y_k)^2 + (z_j - z_k)^2 %%%
    nodes=[x,y,z];
    rd2 = zeros(length(nodes),length(nodes));
    for j=1:3
         xd1 = nodes(:,j);  xd1 = xd1(:,ones(length(xd1), 1));
         xd2 = xd1';
         rd2 = rd2 + (xd1-xd2).^2;
    end

%%% Set-up 2D surface grids in (theta,phi) for computing B (eqn.(11)) %%%
    theta = atan2(z,sqrt(x.^2+y.^2));  phi = atan2(y,x); % phi=lambda in paper
    tn = theta; tn = tn(:,ones(length(xd1), 1)); tc = tn';
    pn = phi;   pn = pn(:,ones(length(phi), 1)); pc = pn';

%%% Compute differentiation matrix D %%%%
    B = 2*(cos(alpha).*cos(tn).*cos(tc).*sin(pn-pc) + ...
          sin(alpha).*(cos(tn).*cos(pn).*sin(tc) - cos(tc).*cos(pc).*sin(tn)));
    B = (u0/a)*B.*(-ep^2*exp(-ep^2.*rd2));
    A = exp(-ep^2.*rd2);
    D = B/A;

%%% Initial Condition Cosine Bell %%%
    r = a*acos(cos(theta).*cos(phi)); % intially located at equator, (0,0)
    h = 1000/2*(1+cos(pi*r/R));       % height of bell is 1000m
    h(r >= R) = 0;

%%% Time-Stepping - 4th Order RK %%%
    dt=12/288*5/6;         % Time step for 12 day revolution
    for nt=2:(1*288*6/5)
          d1 = dt*D*h;
          d2 = dt*D*(h + 0.5*d1);
          d3 = dt*D*(h + 0.5*d2);
          d4 = dt*D*(h + d3);
          h = h + 1/6*(d1 + 2*d2 + 2*d3 + d4);
    end
```

References

[1] E. J. Kansa, Multiquadrics – a scattered data approximation scheme with applications to computational fluid-dynamics – II: Solutions to parabolic, hyperbolic and elliptic partial differential equations, Comput. Math. Appl. 19 (1990) 147–161.

[2] Y. C. Hon, X. Z. Mao, An efficient numerical scheme for Burgers' equation, Appl. Math. Comput. 95 (1998) 37–50.

[3] Y. C. Hon, K. F. Cheung, X. Z. Mao, E. J. Kansa, Multiquadric solution for shallow water equations, ASCE J. Hydr. Engrg. 125 (5) (1999) 524–533.

[4] E. Larsson, B. Fornberg, A numerical study of some radial basis function based solution methods for elliptic PDEs, Comput. Math. Appl. 46 (2003) 891–902.

[5] G. B. Wright, B. Fornberg, Scattered node compact finite difference-type formulas generated from radial basis functions, J. Comput. Phys. 212 (2006) 99–123.

[6] T. A. Driscoll, A. Heryundono, Adaptive residual subsampling methods for radial basis function interpolation and collocation problems, Comput. Math. Appl. Submitted (2006).

[7] Q. T. L. Gia, Galerkin approximation of elliptic PDEs on spheres, J. Approx. Theory 130 (2004) 125–149.

[8] Q. T. L. Gia, Approximation of parabolic PDEs on spheres using spherical basis functions, Adv. Comput. Math.

[9] D. L. Williamson, J. B. Drake, J. J. Hack, R. Jakob, P. N. Swarztrauber, A standard test set for numerical approximations to the shallow water equations in spherical geometry, J. Comput. Phys. 102 (1992) 211–224.

[10] R. D. Nair, J. Côté, A. Staniforth, Cascade interpolation for semi-Lagrangian advection over the sphere, Quart. J. Roy. Meteor. Soc. 125 (1999) 1445–1468.

[11] R. L. Hardy, Multiquadric equations of topograpy and other irregular surfaces, J. Geophy. Res. 76 (1971) 1905–1915.

[12] J. C. Mairhuber, On Haar's theorem concerning Chebychev approximation problems having unique solutions, Proc. Amer. Math. Soc. 7 (1956) 609–615.

[13] M. J. D. Powell, The theory of radial basis function approximation in 1990, in: W. Light (Ed.), Advances in Numerical Analysis, Vol. II: Wavelets, Subdivision Algorithms and Radial Functions, Oxford University Press, Oxford, UK, 1992, pp. 105–210.

[14] W. R. Madych, S. A. Nelson, Bounds on multivariate polynomials and exponential error estimates for multiquadric interpolation, J. Approx. Theory 70 (1992) 94–114.

[15] J. Yoon, Spectral approximation orders of radial basis function interpolation on the Sobolev space, SIAM J. Math. Anal. 23 (4) (2001) 946–958.

[16] T. A. Driscoll, B. Fornberg, Interpolation in the limit of increasingly flat radial basis functions, Comput. Math. Appl. 43 (2002) 413–422.

[17] B. Fornberg, G. Wright, Stable computation of multiquadric interpolants for all values of the shape parameter, Comput. Math. Appl. 48 (2004) 853–867.

[18] B. Fornberg, C. Piret, A stable algorithm for flat radial basis functions on a sphere To be submitted (2006).

[19] E. W. Cheney, W. A. Light, A Course in Approximation Theory, Brooks/Cole, New York, 2000.

[20] B. Fornberg, J. Zuev, The Runge phenomenon and spatially variable shape parameters in RBF interpolation, Comput. Math. Appl. Submitted (2006).

[21] J. Wertz, E. J. Kansa, L. Ling, The role of multiquadric shape parameters in solving elliptic partial differential equations, Comput. Math. Appl. Submitted (2006).

[22] A. Iske, Multiresolution Methods in Scattered Data Modelling, Vol. 37 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, Heidelberg, 2004.

[23] A. Sherwood, How can I arrange N points evenly on a sphere?, `http://www.ogre.nu/sphere.htm`.

[24] R. S. Womersley, I. H. Sloan, Interpolation and cubature on the sphere, `http://web.maths.unsw.edu.au/~rsw/Sphere/`.

[25] K. Jetter, J. Stöckler, J. D. Ward, Error estimates for scattered data interpolation on spheres, Math. Comput. 68 (1999) 733–747.

[26] S. Hubbert, T. Morton, L^p-error estimates for radial basis function interpolation on the sphere, J. Approx. Theory 129 (2004) 58–77.

[27] B. Cockburn, C. Shu, Tvb runge-kutta local projection discontinuous galerkin method for conservation laws ii: General framework, Math. Comput. 52 (1989) 411–435.

[28] R. D. Nair, S. J. Thomas, R. D. Loft, A discontinuous Galerkin transport scheme on the cubed-sphere, Mon. Wea. Rev. 133 (2005) 814–828.

[29] M. Taylor, J. Tribbia, M. Iskandarani, The spectral element method for the shallow water equations on the sphere, J. Comp. Phys. 130 (1997) 92–108.

[30] S. Thomas, R. D. Loft, Semi-implicit spectral element model, J. Sci. Comput. 17 (2002) 339–350.

[31] R. Jakob-Chien, J. J. Hack, D. L. Williamson, Spectral transform solutions to the shallow water test set, J. Comp. Phys. 119 (1995) 164–187.

[32] W. F. Spotz, M. A. Taylor, P. N. Swarztrauber, Fast shallow water equation solvers in latitude-longitude coordinates, J. Comp. Phys. 145 (1998) 432–444.

[33] R. K. Beatson, J. B. Cherrie, C. T. Mouat, Fast fitting of radial basis functions: Methods based on preconditioned GMRES iteration, Adv. Comput. Math. 11 (1999) 253–270.

[34] J. B. Cherrie, R. K. Beatson, G. N. Newsam, Fast evaluation of radial basis functions: methods for generalized multiquadrics $\mathbb{R}^n$, SIAM J. Sci. Comput. 23 (5) (2002) 1549–1571.

[35] G. Roussos, B. J. C. Baxter, Rapid evaluation of radial basis functions, J. Comput. Appl. Math. 180 (2005) 51–70.

[36] S. Rippa, An algorithm for selecting a good value for the parameter c in radial basis function interpolation, Adv. Comput. Math. 11 (1999) 193–210.

[37] R. E. Carlson, T. A. Foley, The parameter R^2 in multiquadric interpolation, Comput. Math. Appl. 21 (1991) 29–42.

[38] E. Larsson, B. Fornberg, Theoretical and computational aspects of multivariate interpolation with increasingly flat radial basis functions, Comput. Math. Appl. 49 (2005) 103–130.

[39] R. Platte, T. A. Driscoll, Eigenvalue stability of radial basis function discretizations for time-dependent problems, Comp. Math. Appl. to appear.

[40] B. Fornberg, A Practical Guide to Pseudospectral Methods, Cambridge University Press, Cambridge, 1996.

[41] B. Fornberg, G. Wright, E. Larsson, Some observations regarding interpolants in the limit of flat radial basis functions, Comput. Math. Appl. 47 (2004) 37–55.

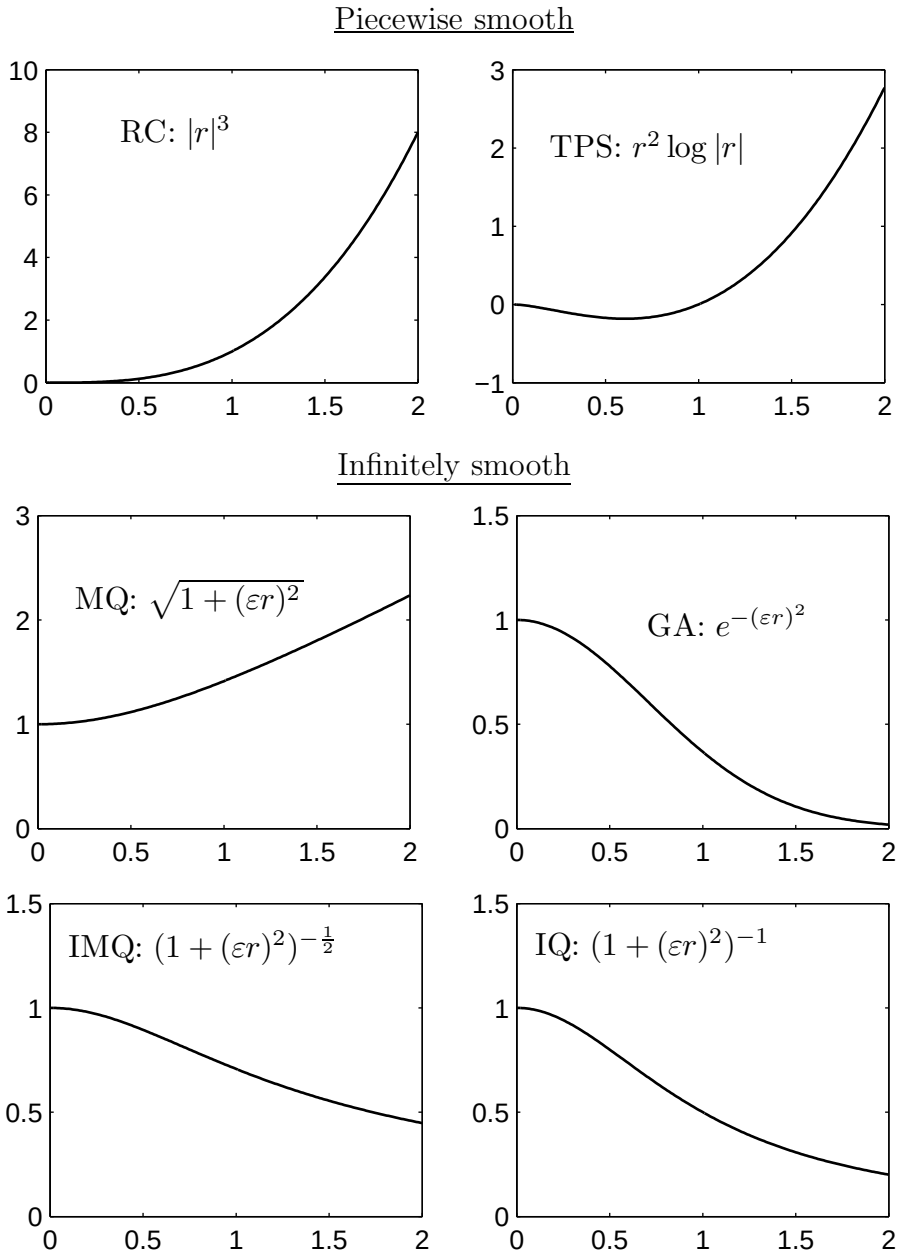


Fig. 1. Examples of commonly used RBFs $\phi(r)$: RC = radial cubic, TPS = thin plate spline, MQ = multiquadric, GA = Gaussian, IMQ = inverse multiquadric, IQ = inverse quadratic. The variable ε in the infinitely smooth RBFs is known as the shape parameter.

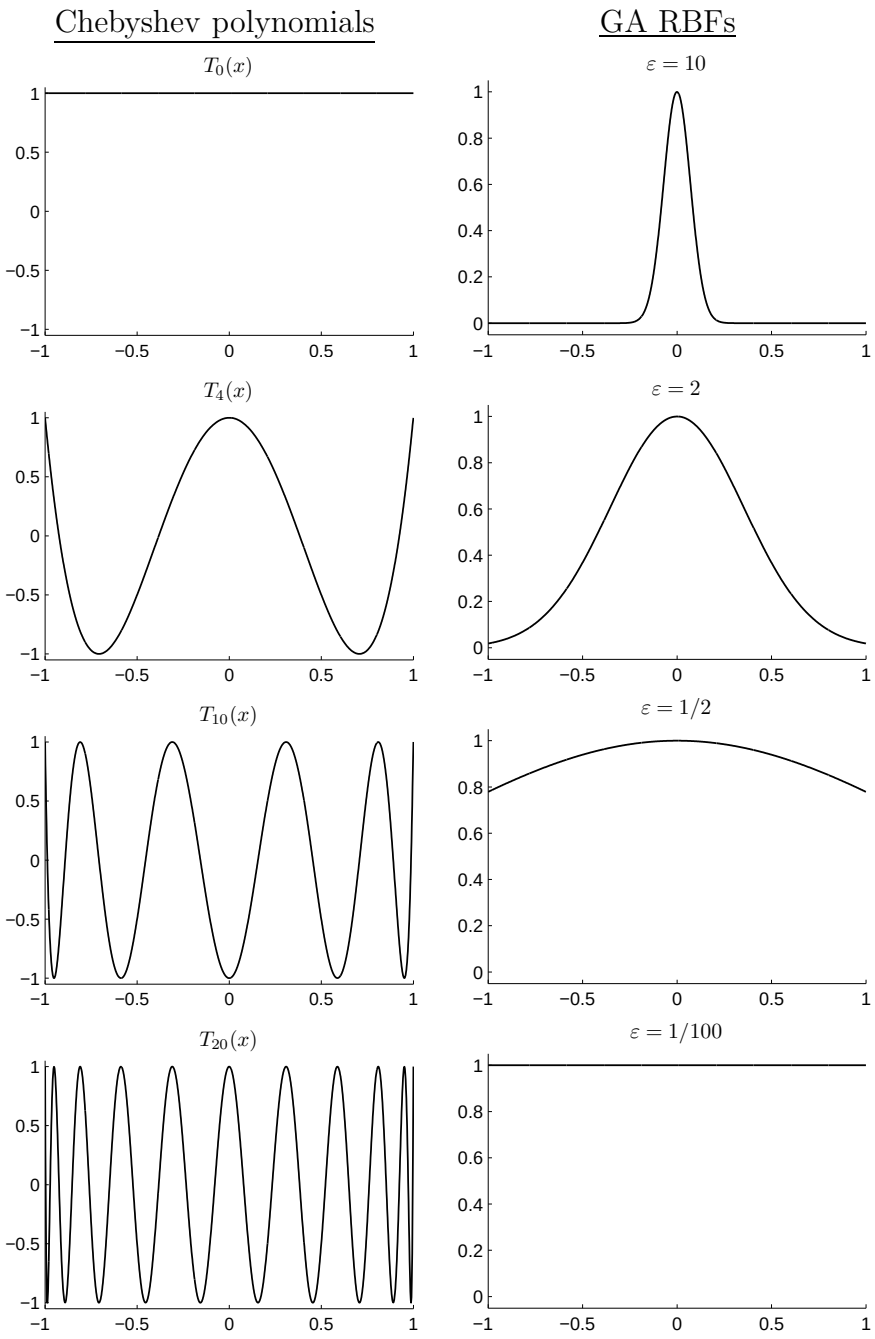


Fig. 2. Comparison of 1-D Chebyshev PS basis, $T_k(x)$, and Gaussian (GA) RBF basis, $\phi(r) = e^{-(\varepsilon r)^2}$ (for this case $r = x$).

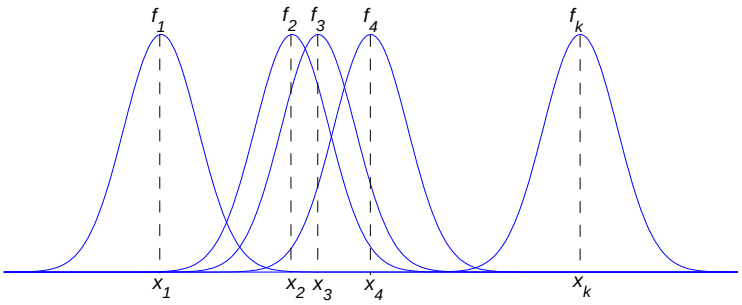


Fig. 3. Example of RBF collocation in 1-D.

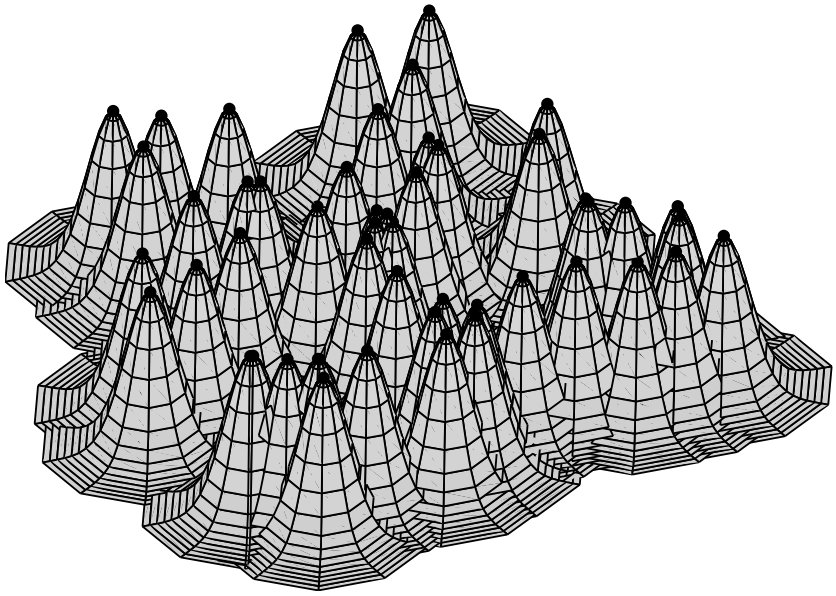


Fig. 4. Example of RBF collocation in 2-D (here, shown for larger ε - i.e. more peaked - than would typically be used).

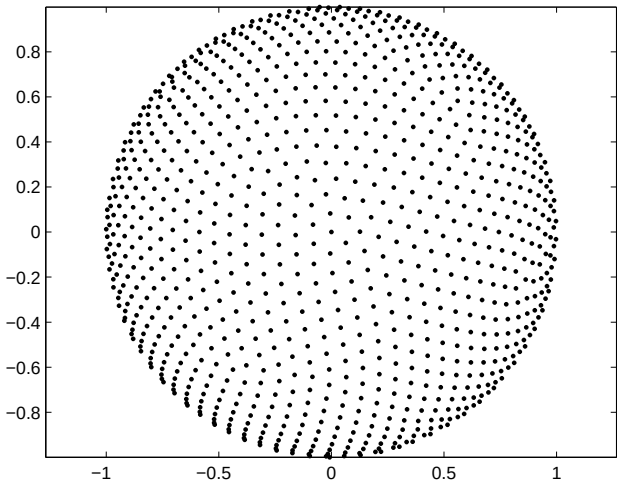


Fig. 5. Minimum energy (ME) node distribution on the sphere, $N = 1849$.

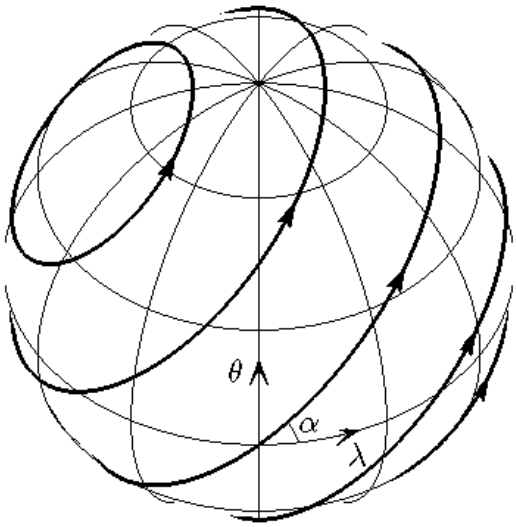
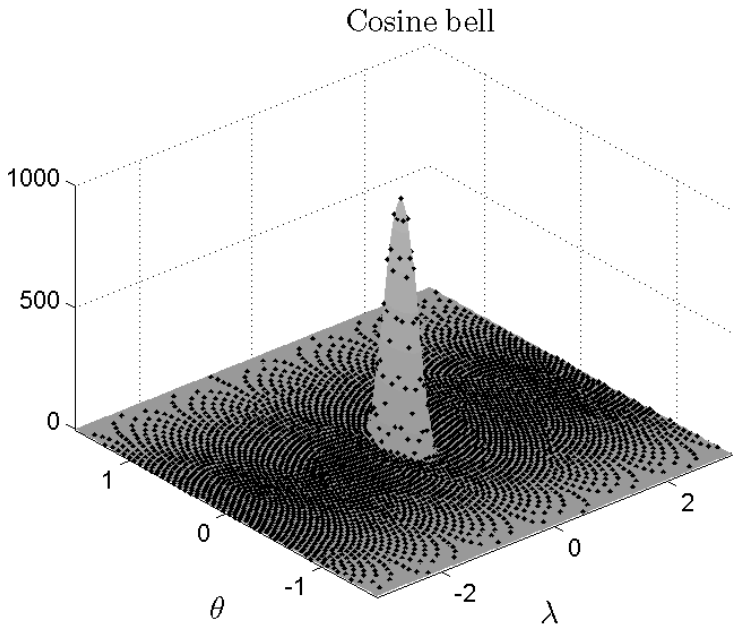
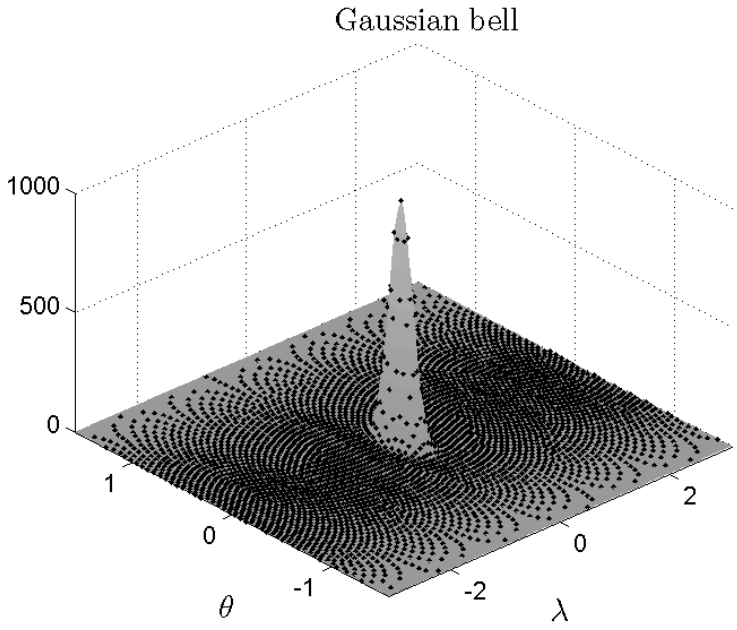


Fig. 6. Solid body rotation over the surface of a sphere at an angle α (darker solid lines) relative to the standard longitude-latitude grid (lighter solid lines).

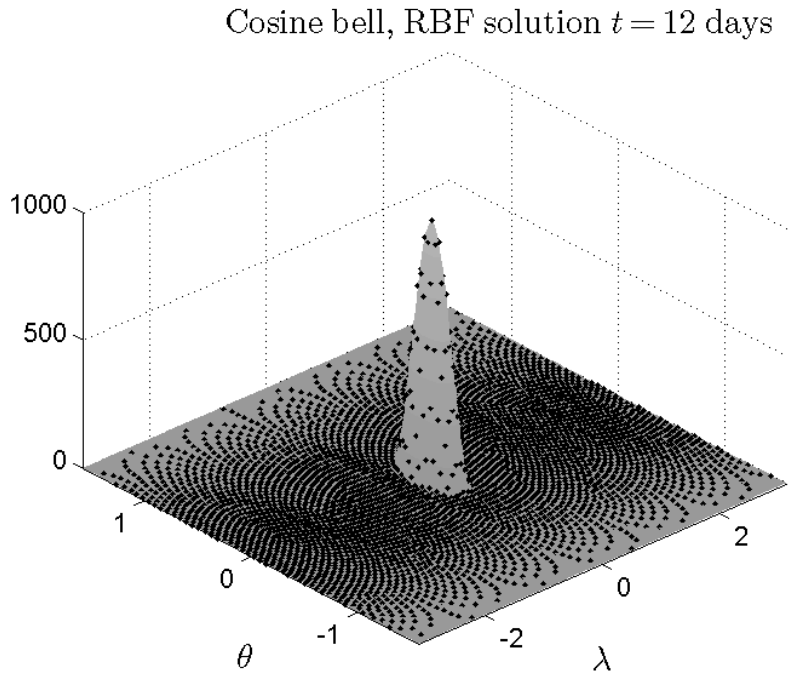


(a)

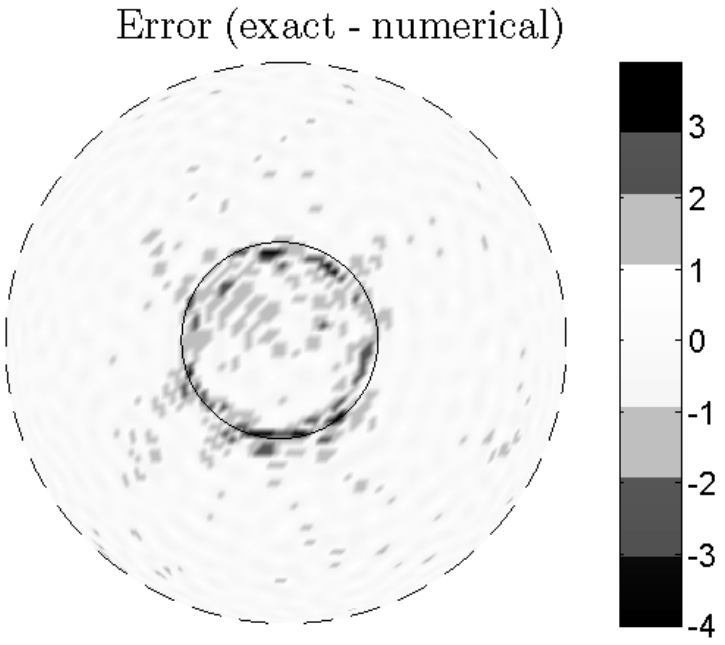


(b)

Fig. 7. (a) Cosine bell and (b) Gaussian bell initial conditions and exact solution after 1 revolution ($t = 12$ days) on an 'unrolled' sphere. Solid black circles mark the $N = 4096$ ME node points.



(a)



(b)

Fig. 8. (a) Numerical solution of the cosine bell test case after 1 revolution ($t = 12$ days) on an 'unrolled' sphere (c.f. Figure 7(a)). Solid black circles mark the $N = 4096$ ME node points. (b) Orthographic projection of the error (exact - numerical) for part (a). Solid line marks the base of the cosine bell where the function is C^1, while the dashed line outlines the sphere. Results are for the GA RBF and RK4 for the time integration with a 30 minute time-step (spatial errors dominate).

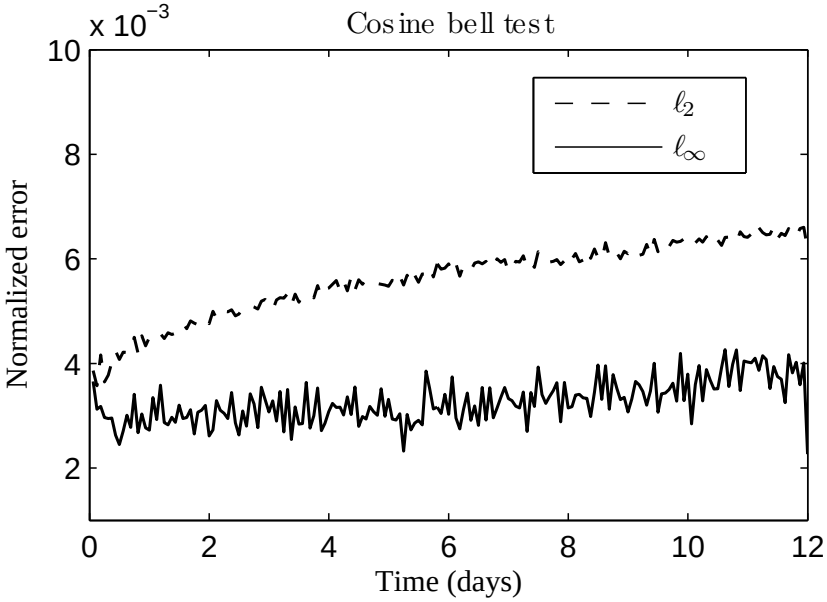
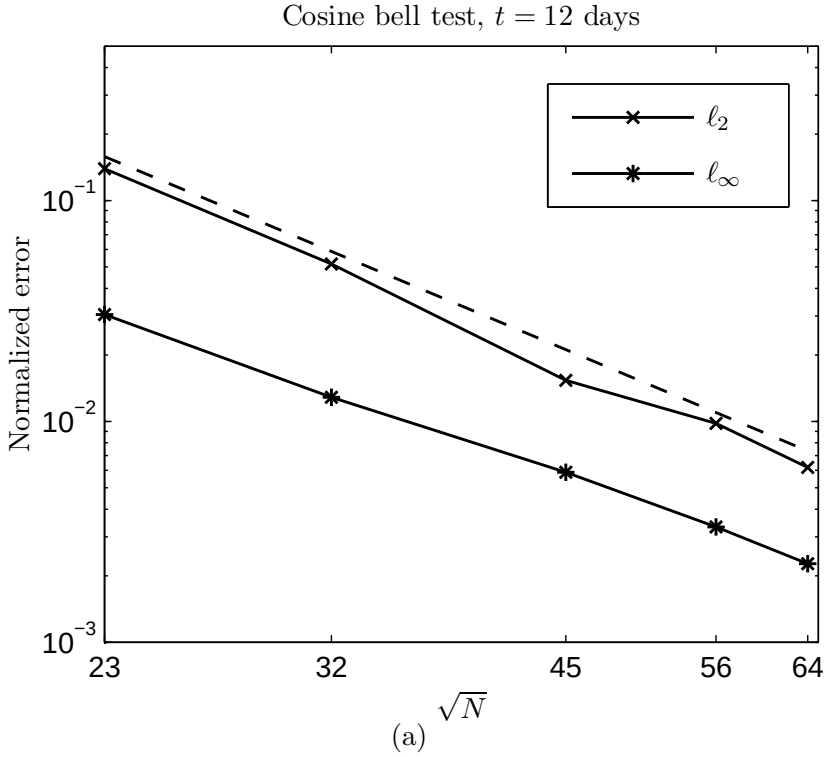
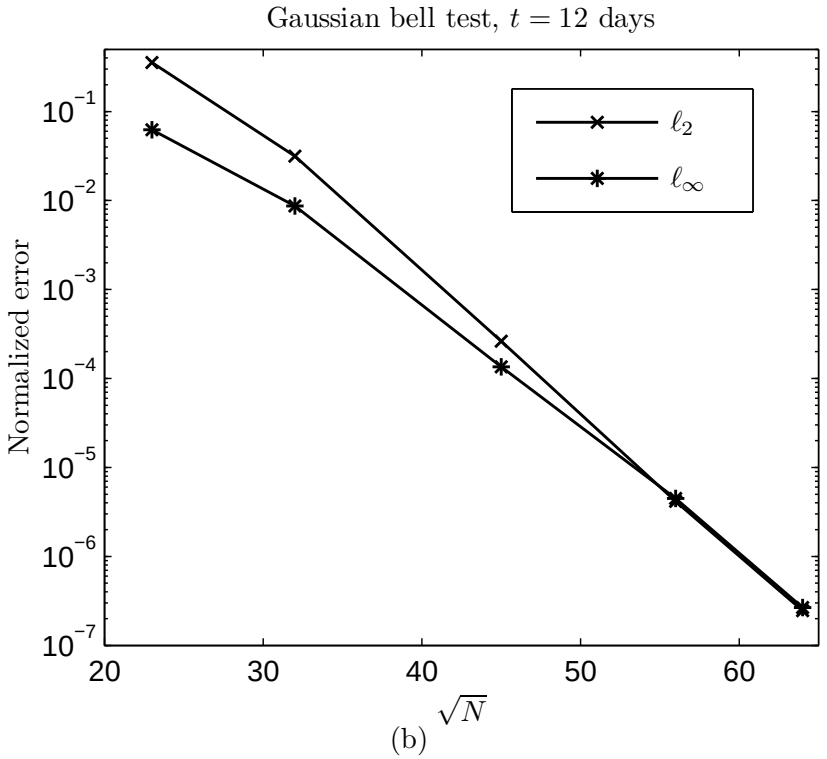


Fig. 9. The ℓ_∞ and ℓ_2 error for the cosine bell test case as a function of time for the $N = 4096$ ME node set. Advection of the height field is directly over the poles. Results are for the GA RBF, and $\Delta t = 30$ minutes in the RK4 integration.

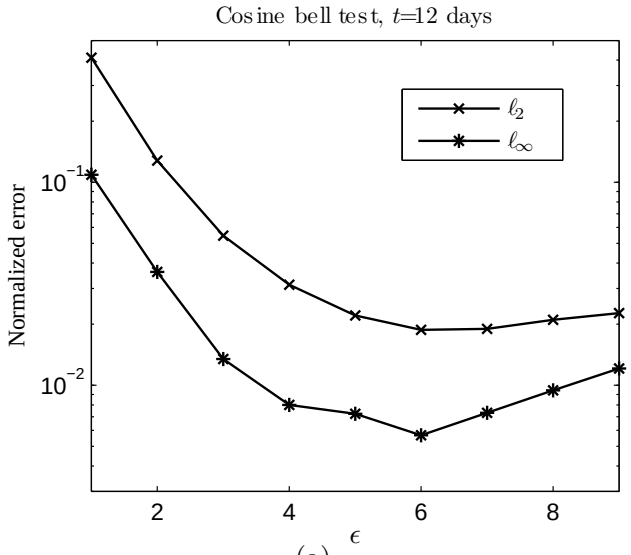


(a)

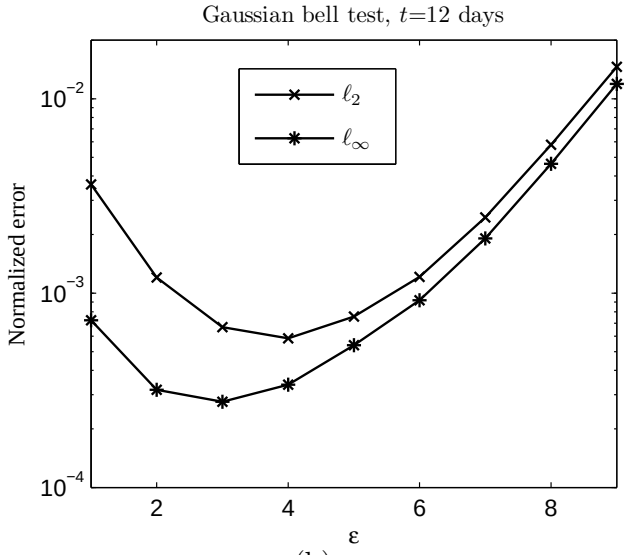


(b)

Fig. 10. (a) The ℓ_∞ and ℓ_2 error at $t = 12$ days for the cosine bell test case as a function of the spacing of the ME node sets ($h \sim N^{-1/2}$) on a log-log plot. The dashed line near the ℓ_2 error is a plot of $N^{-3/2}$, showing that the convergence is cubic. (b) Same as (a) but for the Gaussian test case on a log-linear plot. Results are for the GA RBF, RK4 for the time integration, and the node sets N =529, 1024, 1849, 3136, 4096.



(a)



(b)

Fig. 11. Normalized ℓ_∞ and ℓ_2 error measured after one full revolution ($t = 12$ days) as a function of ε for the (a) cosine bell and (b) Gaussian bell initial conditions. Results are for $N = 1849$ ME node set, the GA RBF, and $\Delta t = 50$ minutes in the RK4 integration.

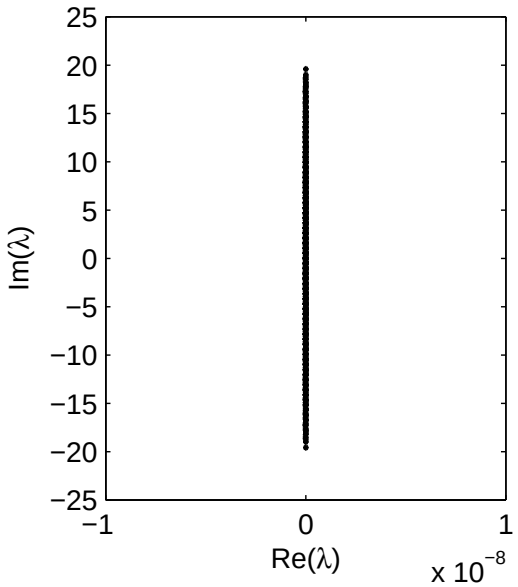
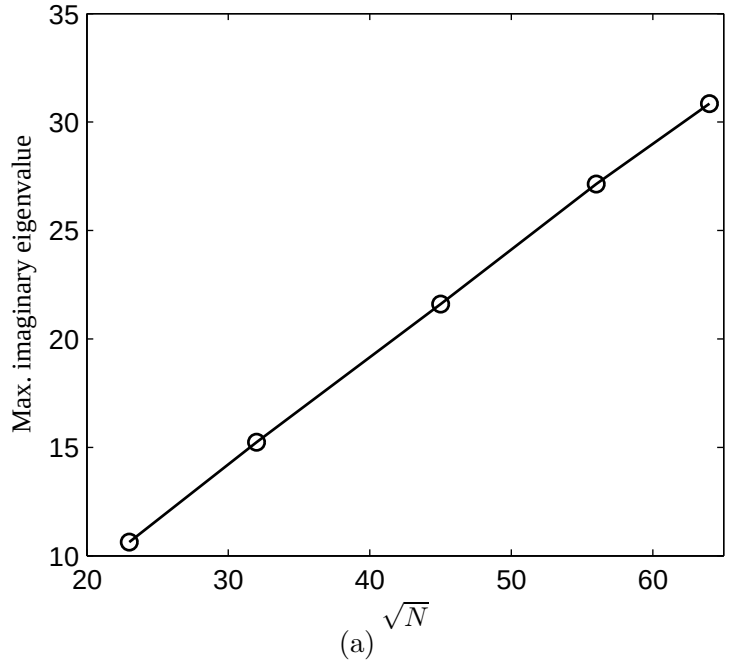
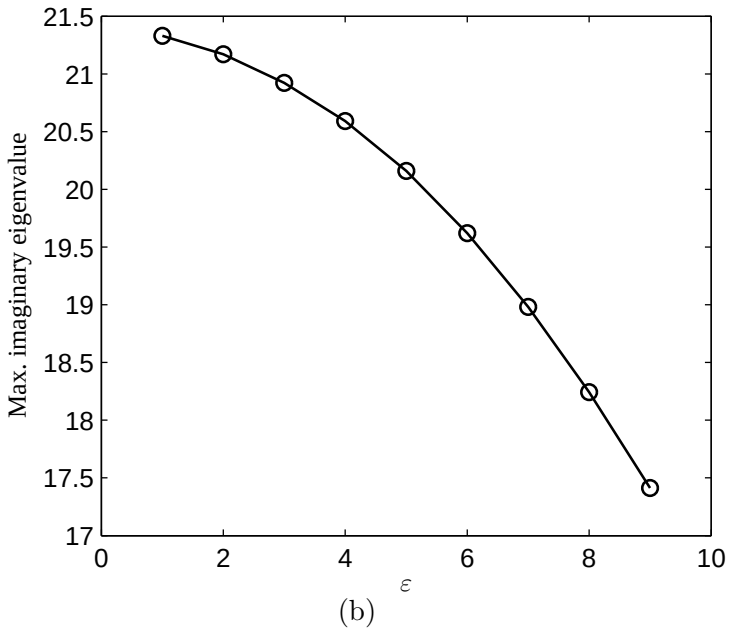


Fig. 12. Eigenvalues of the RBF differentation matrix for the solid body rotation test case. Results are for $N = 1849$ ME node set and the GA RBF with $\varepsilon = 6$.



(a)



(b)

Fig. 13. Maximum eigenvalue of the differentiation matrix for the solid body rotation test case as a function of (a) the spacing of the ME node sets ($h \sim N^{-1/2}$) and (b) ε where $N = 1849$.

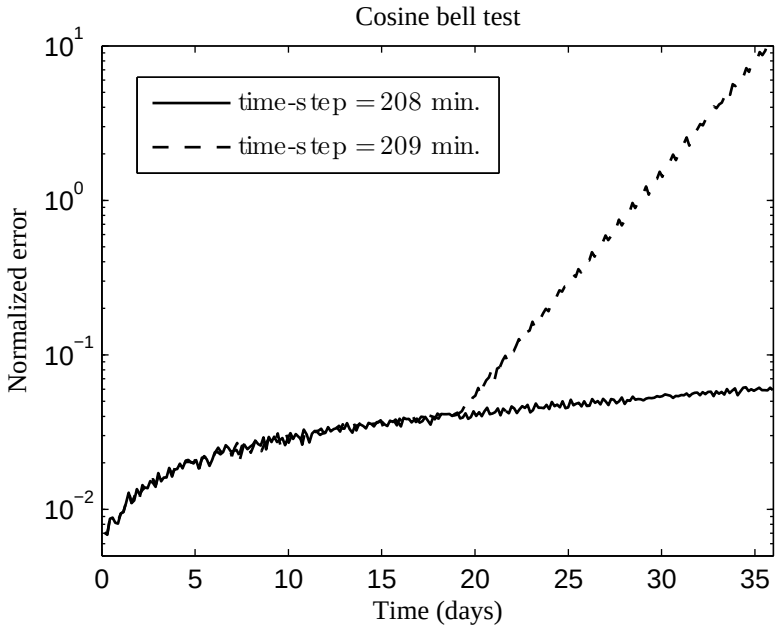


Fig. 14. The ℓ_∞ error for the cosine bell test for $0 \leq t \leq 36$ days (3 full revolutions) using a stable and unstable time-step with RK4 (see Table 2). Results are for the $N = 1849$ ME node set and the GA RBF with $\varepsilon = 6$.

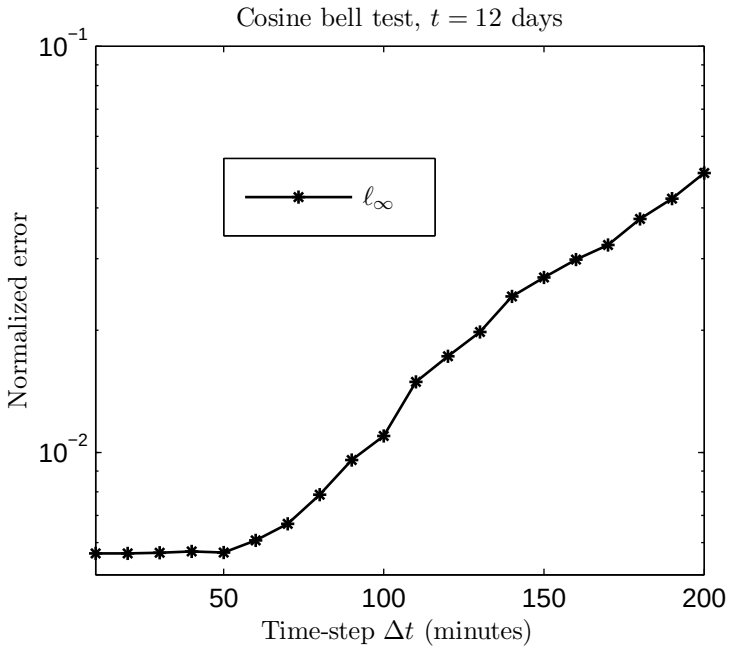


Fig. 15. The ℓ_∞ error at $t = 12$ days for the cosine bell test as a function of the time-step used in the RK4 integration. Results are for the $N = 1849$ ME node set and the GA RBF with $\varepsilon = 6$.

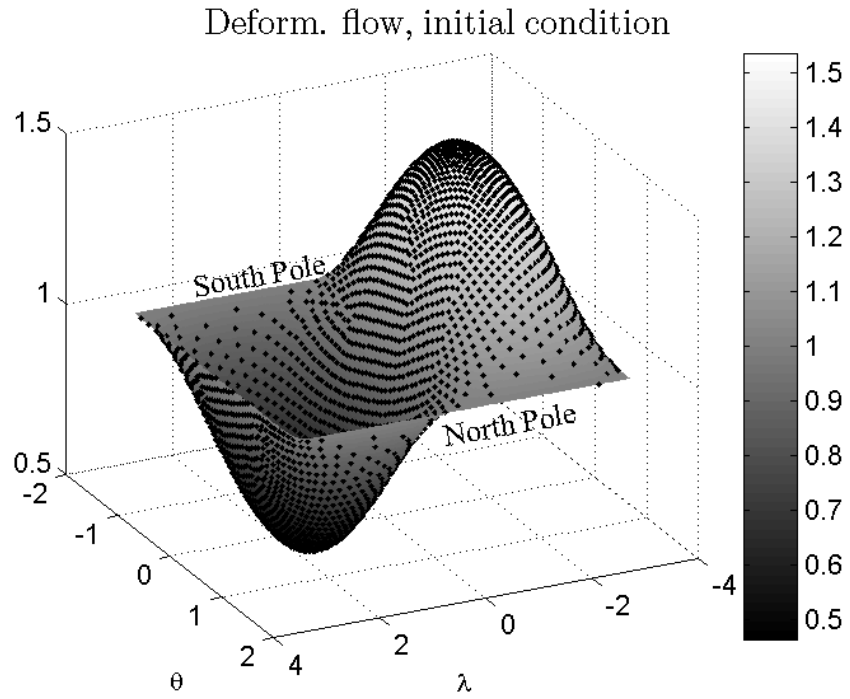


Fig. 16. The initial condition for the deformational test case on an unrolled sphere. Solid black circles mark the $N = 3136$ ME node points.

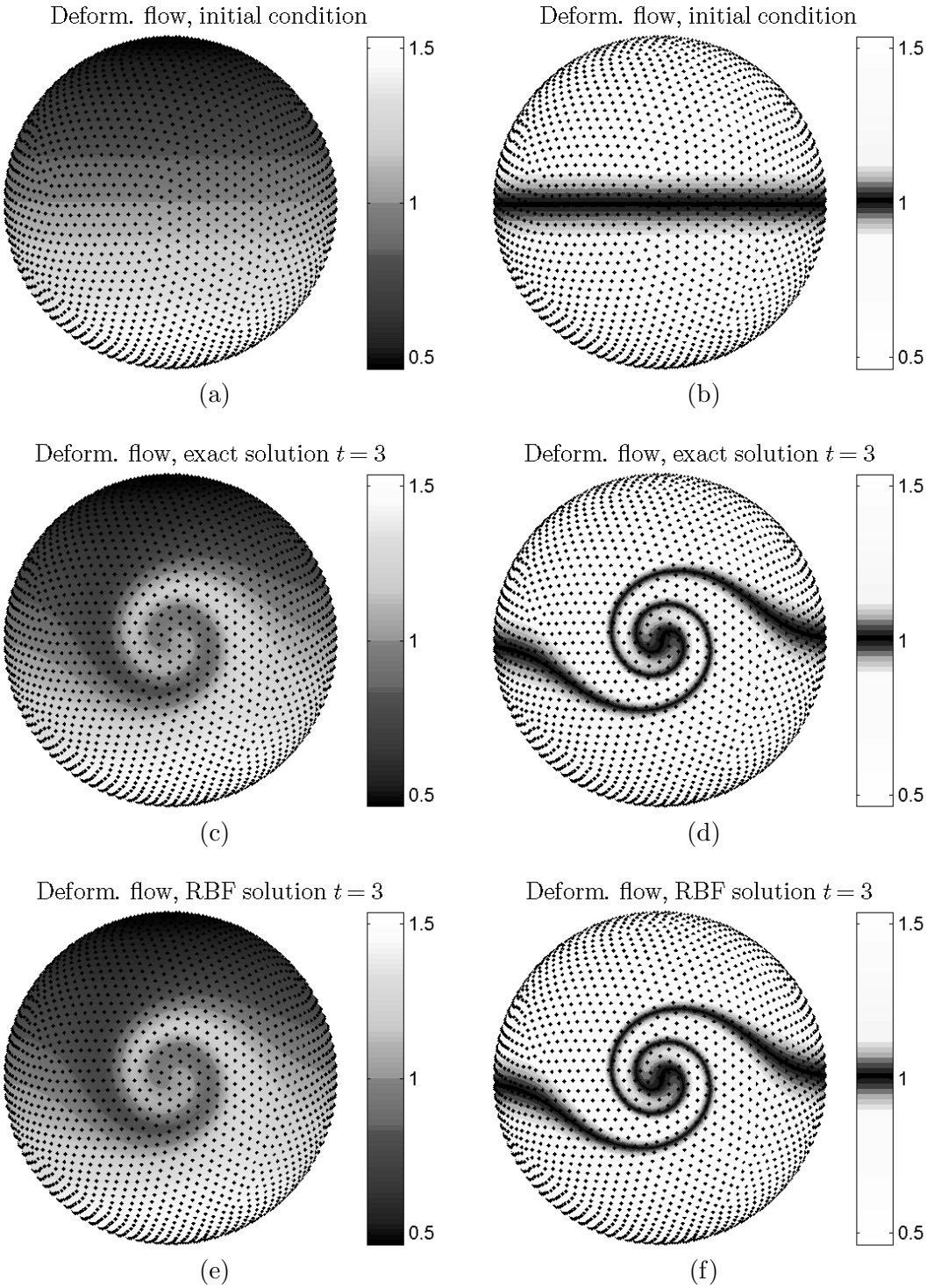


Fig. 17. (a) Initial condition displayed in Figure 16 projected onto the surface of the sphere (view is from the north pole), (b) a band marking the plane $h = 1$ on the sphere at $t = 0$, (c) and (d) are the exact solution at $t = 3$ with (e) and (f) being the RBF solution at $t = 3$ (refer to Section 6.1).

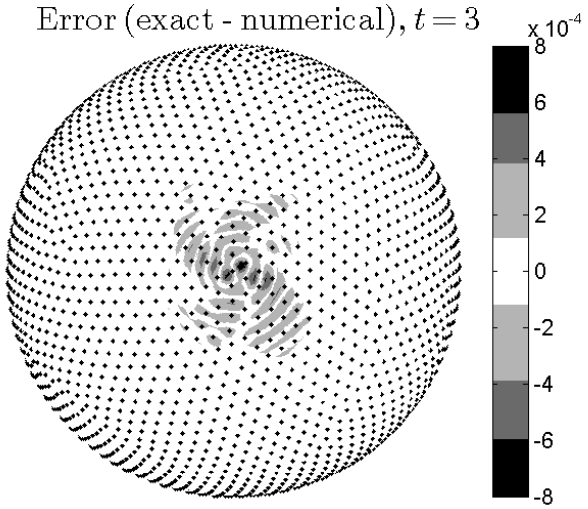


Fig. 18. The error in the solution displayed in Figure 17(e) and (f) at the final time $t = 3$.

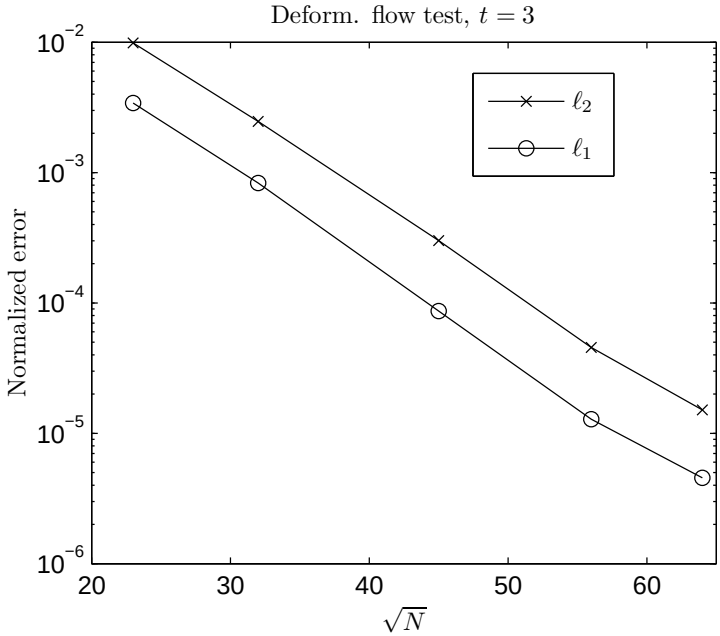


Fig. 19. The ℓ_1 and ℓ_2 error at $t = 3$ time units for the deformational flow test as a function of the spacing of the ME node sets ($h \sim N^{-1/2}$) on a log-linear plot. Results are for the GA RBF and RK4 for the time integration with $\Delta t = 1/10$ (spatial errors dominate).

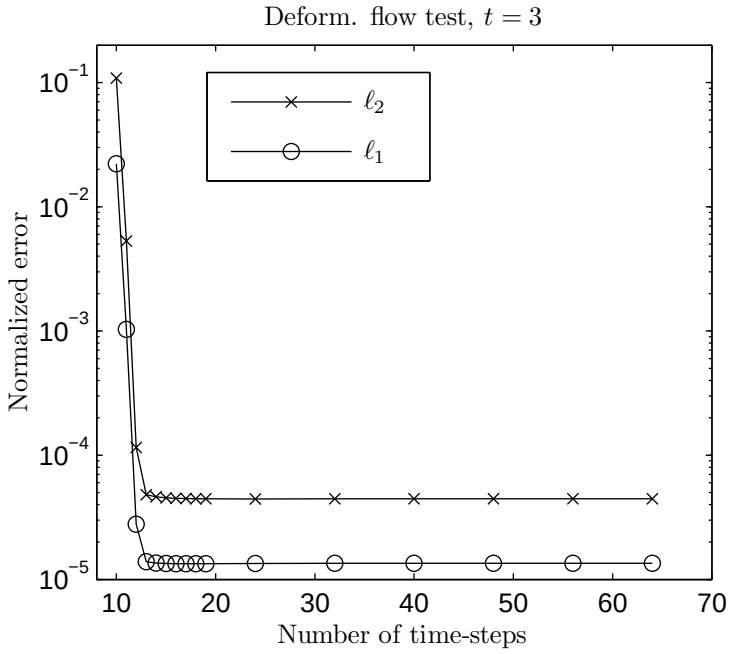


Fig. 20. The ℓ_1 and ℓ_2 error at $t = 3$ for the deformational flow test as a function of the number of time-steps used in the RK4 integration. Results are for the $N = 3136$ ME node set and the GA RBF with $\varepsilon = 6.45$.

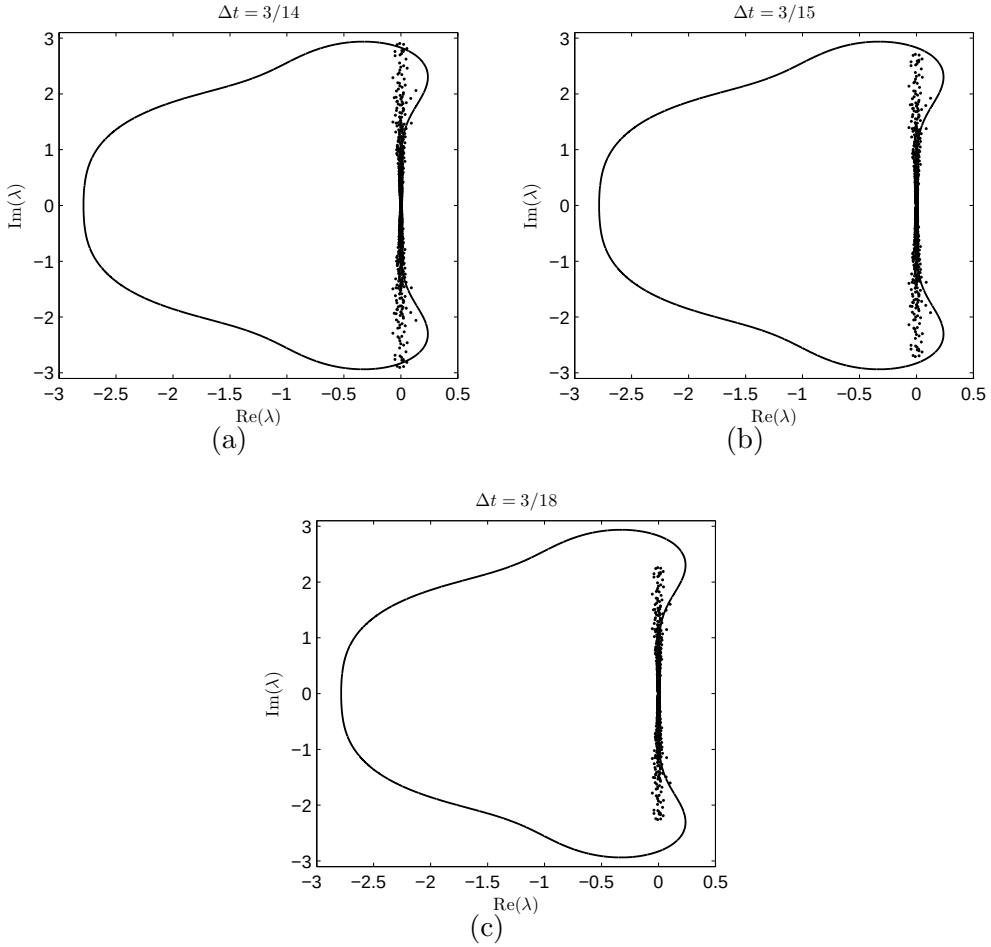
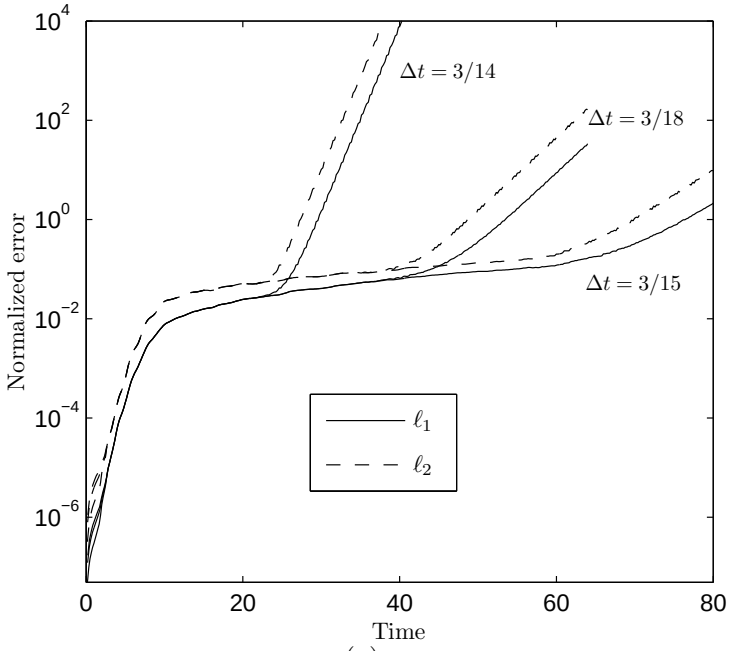
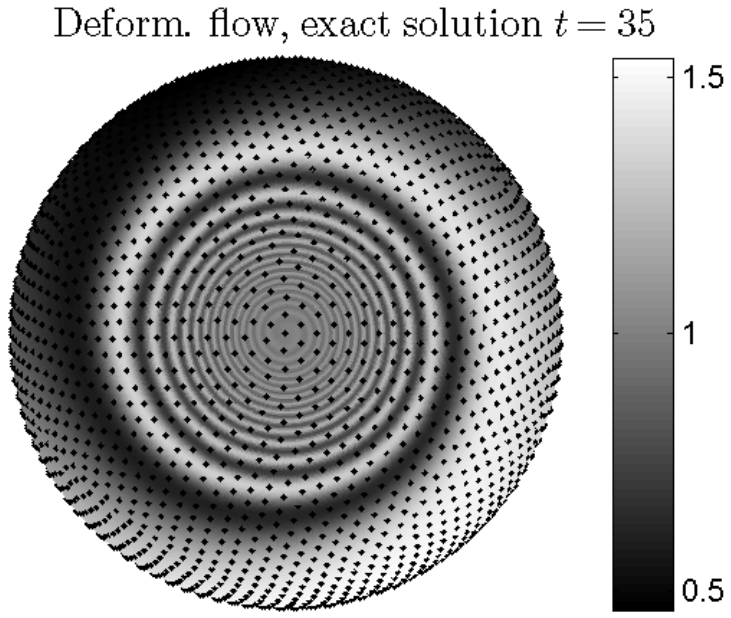


Fig. 21. Stability domain of RK4 together with the eigenvalues of the differentiation matrix for the deformational flow test, $-WD_N$, scaled by (a) $\Delta t = 3/14$, (b) $\Delta t = 3/18$, and (c) $\Delta t = 3/15$.



(a)



(b)

Fig. 22. (a) ℓ_1 and ℓ_2 error for the RBF approximation of the deformational flow test as a function of time for different Δt. (b) Exact solution of the deformational flow test at $t = 35$, the time at which the RBF approximation becomes unstable using a time step of $\Delta t = 3/18$. Results are for $N = 3136$ ME node set and the GA RBF.

Method	Cost per time-step	ℓ_2 error	Time-step	Number of grid points	Code length (# of lines)	Local mesh refinement
RBF	$O(N^2)$	0.006	1/2 hour	4096	< 40	yes
SH	$O(M^{3/2})$	0.005	90 seconds	32768	> 500	no
DF	$O(N \log N)$	0.005	90 seconds	32768	> 100	no
DG	$O(kN_e)$	0.005	6 minutes	7776	> 1000	yes

Table 1
Performance comparsion between commonly used spectral methods in order to achievce a ℓ_2 error of approximately 0.005. M is the number of spherical harmonics (in this case $M = 7396$). N_e is the total number of nodes per element and k is the number of elements.

ε	Max. eigenvalue	$\Delta t_{\max}$ (minutes)
1	21.3	191
2	21.2	192
3	20.9	195
4	20.6	198
5	20.2	202
6	19.6	208
7	19.0	215
8	18.2	223
9	17.4	234

Table 2
A comparison between ε and the corresponding maximum eigenvalue of the differentiation matrix for the cosine bell test together with the maximum allowable time-step for RK4 according to eigenvalue stability theory. Results are for the $N = 1849$ ME node set and the GA RBF.

N	$\sqrt{N}$	Max. eigenvalue	$\Delta t_{\max}$ (minutes)
529	23	10.6	383
1024	32	15.2	267
1849	43	19.6	208
3136	56	27.1	150
4096	64	30.8	132

Table 3
A comparison between $\sqrt{N}$, N = number of nodes, and the maximum eigenvalue of the differentiation matrix for the cosine bell test together with the maximum allowable time-step for the RK4 integrator according to eigenvalue stability theory. $\sqrt{N}$ is inversely proportional to the quasi-uniform spacing of nodes h. Results are for the GA RBFs.