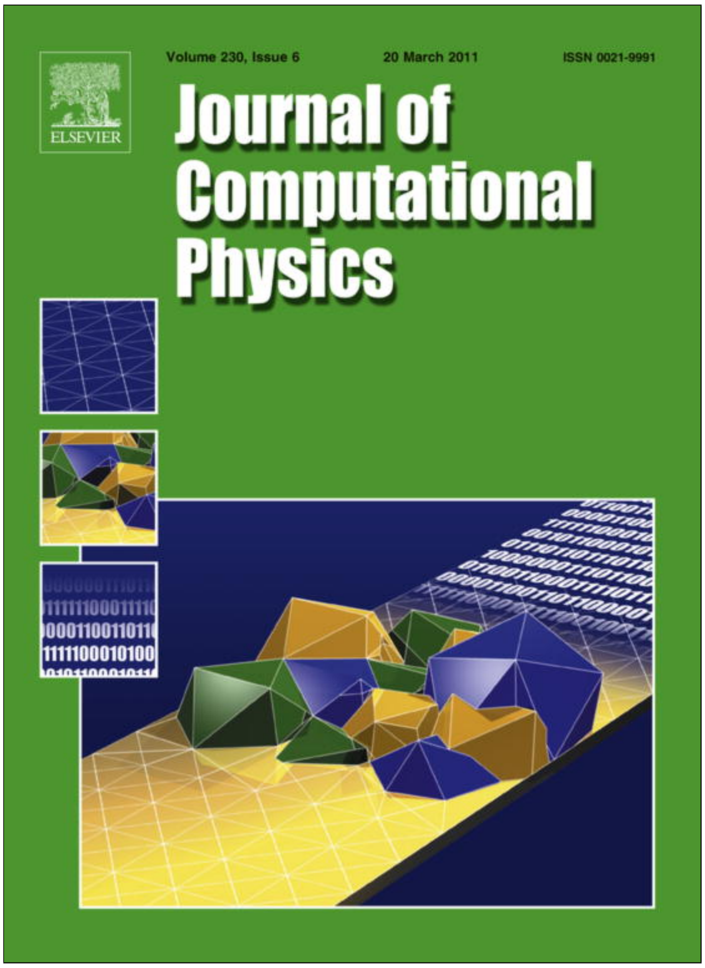

Volume 230, Issue 6
20 March 2011
ISSN 0021-9991
ELSEVIER
Journal of Computational Physics

Journal of Computational Physics 230 (2011) 2270–2285

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Stabilization of RBF-generated finite difference methods for convective PDEs

Bengt Fornberg [a,*], Erik Lehto [b]

[a] University of Colorado, Department of Applied Mathematics, 526 UCB, Boulder, CO 80309, USA
[b] Department of Information Technology, Uppsala University, Box 337, SE-751 05 Uppsala, Sweden

ARTICLE INFO

Article history:
Received 23 June 2010
Received in revised form 7 December 2010
Accepted 9 December 2010
Available online 21 December 2010

Keywords:
Radial basis functions
RBF
Finite differences
RBF-FD
Filter
Stability
Hyperviscosity
Method of lines

ABSTRACT

Radial basis functions (RBFs) are receiving much attention as a tool for solving PDEs because of their ability to achieve spectral accuracy also with unstructured node layouts. Such node sets provide both geometric flexibility and opportunities for local node refinement. In spite of requiring a somewhat larger total number of nodes for the same accuracy, RBF-generated finite difference (RBF-FD) methods can offer significant savings in computer resources (time and memory). This study presents a new filter mechanism, allowing such gains to be realized also for purely convective PDEs that do not naturally feature any stabilizing dissipation.

1. Introduction

Radial basis functions (RBFs) were first applied to the task of solving PDEs about 20 years ago. Only in the last few years has this effort progressed to the point of demonstrating that the approach can compete successfully in terms of cost-effectiveness against the very best existing techniques on several 'global-scale' test problems in the geosciences [5–7,30]; for an overview, see [4]. Even so, calculations using RBFs in full 3D are not likely to scale well for increasing problem sizes, due to the operation count. For time dependent problems with N spatial nodes, it typically costs $O(N^3)$ operations to create the *differentiation matrices* (DMs), which then can be applied at a cost of $O(N^2)$ operations for each time step. With the alternative approach of instead using local RBF-generated finite difference (RBF-FD) formulas, both of these types of costs will be reduced to $O(N)$ operations. Memory requirements will similarly get reduced from $O(N^2)$ to $O(N)$. Although the accuracy will be somewhat less for the same number of nodes, the overall computational cost-effectiveness can nevertheless be significantly higher. Some recent studies on solving elliptic and convection–diffusion type PDEs have shown the RBF-FD approach to be very successful in meshfree multivariate settings [2,22,24–26,31]. Such RBF-FD methods have not yet been described in the literature for purely convection-type PDEs. The reason for this is the common occurrence of spurious eigenvalues in the resulting DMs, together with the fact that the PDE itself in these cases offers no stabilizing viscous damping. As a result, numerical time stepping becomes very prone to instabilities. We introduce here two easy to implement RBF-based types of filters, which damp all spurious modes while leaving the physically relevant modes essentially intact. The preferred ap-

* Corresponding author.
E-mail addresses: fornberg@colorado.edu (B. Fornberg), erik.lehto@it.uu.se (E. Lehto).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.12.014

proach when using RBF-FD formulas involves approximating high powers of the Laplacian operator, and is closely related to the hyperviscosity method, which is commonly used in direct numerical simulations of turbulence (see [1] for references; an early motivation of the concept can be found in [8]). The approach we introduce here for the global RBF case achieves stabilization without involving any approximations of high derivatives. Both approaches make highly accurate simulations possible for a wide range of convection-type PDEs also over very long integration times. We verify this for two standard test problems of purely convective flows over the surface of a sphere.

2. FD-type formulas for derivative approximations

RBF-FD approximations arise naturally as generalizations of regular FD approximations, which we therefore summarize next.

2.1. Polynomial-based FD formulas

Regular FD formulas are usually derived only in 1D. Generalizations to more dimensions are mostly limited to Cartesian type grids, which allow such 1D formulas to be applied separately in each spatial direction. The simplest approach for finding the weights in 1D FD formulas (although not the most effective one, cf. [10]) is to consider the FD weights to be unknowns and then enforce that the resulting FD formula becomes exact for the monomials $1, x, x^2, \ldots$, up to as high degree as possible. For example, in the case of approximating a linear differential operator L (such as $\frac{\partial}{\partial x}, \frac{\partial^2}{\partial x^2}$, etc.) at a location x_c with an n-node stencil, we can determine the weights w_i at node locations x_i, $i = 1, 2, \ldots, n$, by solving the linear system

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & x_n \\ x_1^2 & x_2^2 & x_3^2 & \cdots & x_n^2 \\ \vdots & \vdots & \vdots & & \vdots \\ x_1^{n-1} & x_2^{n-1} & x_3^{n-1} & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} L1|_{x=x_c} \\ Lx|_{x=x_c} \\ Lx^2|_{x=x_c} \\ \vdots \\ Lx^{n-1}|_{x=x_c} \end{bmatrix}. \tag{2.1}$$

As long as the nodes x_i are distinct, this Vandermonde matrix is guaranteed to be nonsingular. This *unisolvency* property is no longer assured if the approach is generalized to scattered nodes and multivariate polynomials in 2D or higher.

2.2. RBF-generated FD formulas

The key idea is to replace the 1D polynomial test functions $\{1, x, x^2, \ldots\}$ with dD RBFs $\phi(\|\underline{x} - \underline{x}_i\|)$ centered at the nodes $\underline{x}_i$ ($i = 1, 2, \ldots, n$). For the RBF types listed in Table 1 (and many others), this restores guaranteed nonsingularity, no matter how the (distinct) nodes $\underline{x}_i$ are scattered. With $\|\cdot\|$ denoting the standard Euclidean 2-norm, the counterpart to (2.1) for approximating a linear operator L at location $\underline{x}_c$ becomes

$$\begin{bmatrix} \phi(\|\underline{x}_1 - \underline{x}_1\|) & \phi(\|\underline{x}_2 - \underline{x}_1\|) & \cdots & \phi(\|\underline{x}_n - \underline{x}_1\|) \\ \phi(\|\underline{x}_1 - \underline{x}_2\|) & \phi(\|\underline{x}_2 - \underline{x}_2\|) & \cdots & \phi(\|\underline{x}_n - \underline{x}_2\|) \\ \vdots & \vdots & & \vdots \\ \phi(\|\underline{x}_1 - \underline{x}_n\|) & \phi(\|\underline{x}_2 - \underline{x}_n\|) & \cdots & \phi(\|\underline{x}_n - \underline{x}_n\|) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} L\phi(\|\underline{x} - \underline{x}_1\|)|_{\underline{x}=\underline{x}_c} \\ L\phi(\|\underline{x} - \underline{x}_2\|)|_{\underline{x}=\underline{x}_c} \\ \vdots \\ L\phi(\|\underline{x} - \underline{x}_n\|)|_{\underline{x}=\underline{x}_c} \end{bmatrix}. \tag{2.2}$$

The matrix A in the linear system above is the same as the matrix that arises in standard RBF interpolation of scattered data: If the data values are f_i at the locations $\underline{x}_i$, $i = 1, 2, \ldots, n$, the RBF interpolant becomes

$$s(\underline{x}) = \sum_{i=1}^{n} \lambda_i \phi(\|\underline{x} - \underline{x}_i\|), \tag{2.3}$$

Table 1
Definitions of some smooth radial functions. The shape parameter ε controls the flatness (width) of the basis functions.

Examples of positive definite radial functions		
GA	Gaussian	$\phi(r) = e^{-(\varepsilon r)^2}$
IMQ	Inverse multiquadric	$\phi(r) = \frac{1}{\sqrt{1+(\varepsilon r)^2}}$
IQ	Inverse quadratic	$\phi(r) = \frac{1}{1+(\varepsilon r)^2}$
Example of a radial function that is not positive definite		
MQ	Multiquadric	$\phi(r) = \sqrt{1+(\varepsilon r)^2}$

where the coefficient vector $\underline{\lambda}$ is obtained by solving $A\underline{\lambda} = \underline{f}$. In this case of RBF interpolation, (2.3) is often replaced by $s(\underline{x}) = (\sum_{i=1}^{n} \lambda_i \phi(||\underline{x} - \underline{x}_i||)) + \lambda_{n+1}$ together with the constraint $\sum_{i=1}^{n} \lambda_i = 0$. When using this alternate procedure, the same type of derivation that led to (2.2) will instead lead to the relation

$$\left[\begin{array}{cccc|c} & & & & 1 \\ & A & & & \vdots \\ & & & & 1 \\ \hline 1 & \cdots & 1 & & 0 \end{array}\right] \left[\begin{array}{c} w_1 \\ \vdots \\ w_n \\ \hline w_{n+1} \end{array}\right] = \left[\begin{array}{c} L\phi(||\underline{x} - \underline{x}_1||)|_{\underline{x}=\underline{x}_c} \\ \vdots \\ L\phi(||\underline{x} - \underline{x}_n||)|_{\underline{x}=\underline{x}_c} \\ \hline L1|_{\underline{x}=\underline{x}_c} \end{array}\right]. \tag{2.4}$$

The A-matrix in (2.4) is the same as the matrix in (2.2), and the last entry w_{n+1} in the solution vector should be ignored.

We use RBFs to calculate derivative approximations in two different modes, *global* and *local*. In both cases, each of the N nodes across the full domain requires its own stencil. In the global mode, the number of nodes/weights n within each stencil will be the same as the total number of nodes N in the full domain. In this case, we will calculate all weights by (2.2). This study focuses on the *local* case (RBF-FD), for which the number of nodes n in each stencil is much less than N. Since numerical experiments show that (2.4) then usually is a better choice than (2.2), this was used for the tests we describe in Sections 5 and 6. All the work in the present study has been carried out using GA RBFs, i.e. $\phi(r) = e^{-(\varepsilon r)^2}$.

2.3. Differentiation matrices (DMs)

2.3.1. Global RBF derivative approximations

In the global case, the nodes within the stencil, $x_1, \underline{x}_2, \ldots, \underline{x}_n$, include all the nodes in a domain, i.e. $n = N$. The task of approximating L requires for each node point a separate set of weights. A DM is a matrix that contains these weight sets in its successive rows. To find this DM, we first use (2.2) to obtain the full weights sets at every one of the node points, giving $A\,W = B$, where A again has the elements $A_{i,j} = \phi(\|\underline{x}_j - \underline{x}_i\|)$, and B is a square matrix with the elements $B_{i,j} = L\phi(||\underline{x} - \underline{x}_i||)|_{\underline{x}=\underline{x}_j}$. The DM D then becomes

$$D = W^T = B^T A^{-1}, \tag{2.5}$$

i.e. we approximate

$$\begin{bmatrix} Lf(\underline{x})|_{x=x_1} \\ \vdots \\ Lf(\underline{x})|_{x=x_N} \end{bmatrix} \approx \begin{bmatrix} D \end{bmatrix} \begin{bmatrix} f(\underline{x}_1) \\ \vdots \\ f(\underline{x}_N) \end{bmatrix}.$$

Both A and B are obtained in $O(N^2)$ operations, but computing D requires $O(N^3)$ operations (assuming the use of direct methods). When solving a time dependent PDE by a Method of Lines (MOL) approach, each time step will require one or more matrix-times-vector multiplications, at a cost of $O(N^2)$ operations.

It can be noted that *compactly supported* RBFs (such as *Wendland functions* and related similar classes [27,33]) will make both A and B sparse, but this property does not carry over to D. The purpose of the RBF-FD approach is to speed up calculations by achieving matrices D that are both accurate and very sparse.

Another significant issue relates to numerical conditioning, especially in cases when good accuracy requires small values of the shape parameter ε. The matrices A and B remain finite even in the $\varepsilon \to 0$ limit, whereas the elements of A^{-1} then diverge rapidly to infinity. Nevertheless, the entries of $D = B^T A^{-1}$ typically converge (always in the case of GA RBFs [3,17,23]). Their stable numerical calculation for all values of ε was first achieved by the Contour–Padé algorithm [16], although with quite severe limitations on N ($N \lesssim 25$ in 1D and $N \lesssim 80$ in 2D; significantly higher in 3D). These limitations are restrictive for global RBF approximations, but can be acceptable in RBF-FD contexts. Significant efforts have since been devoted to finding stable algorithms without any limits on the number of nodes. The original version of the RBF-QR algorithm [14] applies to any number of nodes scattered over the full surface of a sphere, and a recent variation applies to scattered nodes in finite domains in 1D, 2D and 3D [12].

2.3.2. RBF-FD derivative approximations

For each of the N node points in a computational domain, the RBF-FD approach requires a separate stencil, each based on n node points. The resulting RBF-FD DMs are best represented as sparse matrices, with n entries in each of their N rows. Each application of a spatial operator will then only require one sparse matrix–vector multiplication, i.e. about $2Nn$ arithmetic operations (where $n \ll N$). For simplicity, we include in the RBF-FD stencil with each node only its $n - 1$ nearest neighbors. Finding these neighbors is a computationally fast task even for very large N, when using *kd-trees* (for which effective implementations are available, e.g. in Matlab Central). However, it is possible that other strategies (not tested in the present work) might be advantageous. For example, in the vicinity of boundaries, it may become necessary to use partly one-sided approximations. Also, for convective operators L, it might be preferable to use more nodes along rather than across the direction of the flow [11], or to incorporate some type of upwind concepts.

3. Three related strategies for high frequency damping

The first three subsections below each describe a high mode filtering strategy. We illustrate each in the case of a simple 1D equispaced test case. The two RBF-based methods are novel, and they apply immediately also to scattered multidimensional node sets.

3.1. Regular FD-based hyperviscosity

This procedure is easiest to describe on an infinite unit-spaced grid in 1D, i.e. $x_k = k$, k integer. The Fourier modes that are distinguishable from each other on this grid are then $e^{i\omega x}$, $|\omega| \leqslant \pi$. Choosing for example $n = 7$, the main possibilities for diffusive FD operators that can be implemented with n node wide stencils, become

$$D = \begin{cases} \frac{\partial^2 u}{\partial x^2}, & \text{approximated to 2nd, 4th, and 6th order,} \\ -\frac{\partial^4 u}{\partial x^4}, & \text{approximated to 2nd, and 4th order,} \\ \frac{\partial^6 u}{\partial x^6}, & \text{approximated to 2nd order,} \end{cases}$$

Fig. 3.1(a) and (b) show the rate by which the PDE $\frac{\partial u}{\partial t} = \gamma D u$ will alter the amplitude of these modes, for the different cases above. In case of stencils with weights ν_k at nodes k, $k = -3, -2, -1, 0, 1, 2, 3$, the curves in Fig. 3.1 are calculated as $\nu(\omega) = \sum_{k=-3}^{3} \nu_k e^{ik\omega}$, followed by a scaling so that $\nu(\pm\pi) = 1$. It is clear from parts a,b of the figure that the last case ($\frac{\partial^6 u}{\partial x^6}$, approximated to 2nd order) is optimal in the sense that it leaves the widest range of modes around $\omega = 0$ intact while still damping all high ones.

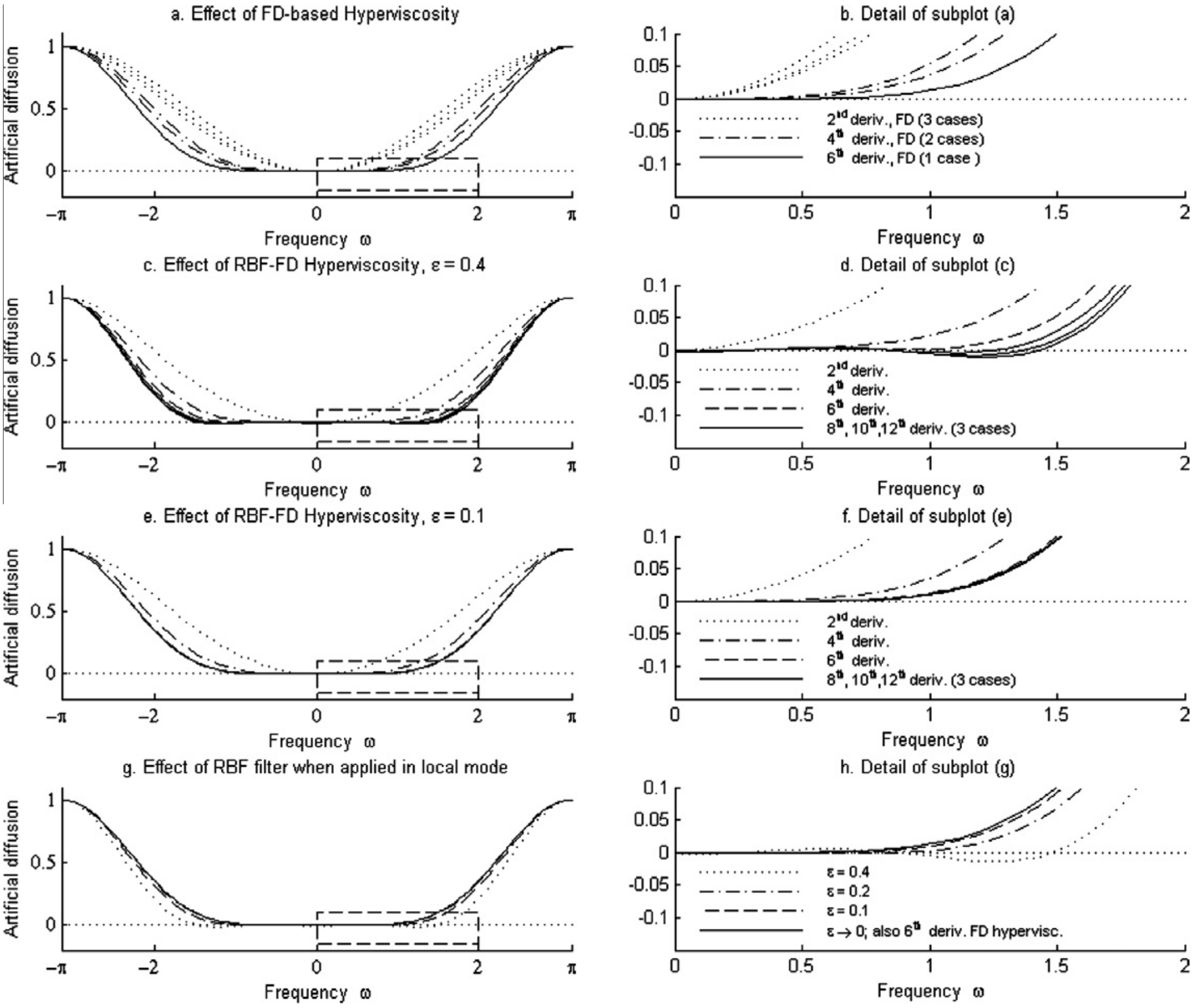


Fig. 3.1. The amount to which different 7-node based dissipative procedures leaves low modes intact. (a) and (b) FD-based hyperviscosity, (c) and (d) RBF-based hyperviscosity ε = 0.4, (e) and (f) same as (c) and (d) but using ε = 0.1, (g) and (h) global RBF filter when applied in local mode.

3.2. RBF-FD operators for hyperviscosity

We consider at first scattered nodes in dD. Let the operator L be a power of the Laplace operator $\Delta = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} + \cdots + \frac{\partial^2}{\partial x_d^2}$. To obtain the weights $\underline{w}$ by means of (2.2) or (2.4), with $\phi(r) = e^{-(\varepsilon r)^2}$, requires us to evaluate

$$L\phi(r) = \Delta^k \phi(r) = \varepsilon^{2k} p_k(r)\phi(r) \tag{3.1}$$

for $r = \|\underline{x}_c - \underline{x}_i\|$, $i = 1,2,\ldots,n$. In this relation, the polynomials $p_k(r)$ are multiples of generalized Laguerre polynomials (with the property that $P_\nu^\mu(x)$ are orthogonal over $x \in [0,\infty]$ with respect to the weight function $e^{-x}x^\mu$, $\nu = 0, 1, \ldots$):

$$p_k(r) = (-4)^k\; k!\; P_k^{d/2-1}((\varepsilon r)^2), \quad k = 0,1,2,\ldots$$

They can readily be calculated recursively

$$\begin{cases} p_0(r) & = 1, \\ p_1(r) & = 4(\varepsilon r)^2 - 2d, \\ p_{k+1}(r) & = 4\left((\varepsilon r)^2 - 2k - \frac{d}{2}\right)p_k(r) - 8k(2k-2+d)p_{k-1}(r), \quad k = 1,2,\ldots \end{cases}$$

In the special case of $d = 2$, the generalized Laguerre polynomials reduce to the regular ones:

$$\begin{aligned} \Delta^0\phi(r) &= \quad \{1\}\phi(r), \\ \Delta^1\phi(r) &= \varepsilon^2\left\{4(\varepsilon r)^2 - 4\right\}\phi(r), \\ \Delta^2\phi(r) &= \varepsilon^4\{16(\varepsilon r)^4 - 64(\varepsilon r)^2 + 32\}\phi(r), \\ \Delta^3\phi(r) &= \varepsilon^6\{64(\varepsilon r)^6 - 576(\varepsilon r)^4 + 1152(\varepsilon r)^2 - 384\}\phi(r), \\ &\ldots \end{aligned} \tag{3.2}$$

The second and third rows of subplots in Fig. 3.1 illustrate the effect of these operators (3.1) in the 1D case when using the formulation (2.2), again with $n = 7$ and with $\varepsilon = 0.4$ and $\varepsilon = 0.1$, respectively. For $k = 1, 2, 3$ (i.e. using $L = \Delta^1, -\Delta^2, \Delta^3$, resp.), we see a behavior similar to that for the FD approximations to $\frac{\partial^2 u}{\partial x^2}, -\frac{\partial^4 u}{\partial x^4}, \frac{\partial^6 u}{\partial x^6}$ but, in sharp contrast to the FD case, it is now perfectly possible to continue to still higher order derivatives with $k = 4, 5, 6, \ldots$ This does not suggest that we are able to actually approximate $\frac{\partial^8 u}{\partial x^8}, \frac{\partial^{10} u}{\partial x^{10}}, \frac{\partial^{12} u}{\partial x^{12}}, \ldots$ using only seven nodes – it just shows that implementing hyperviscosity by means of RBFs settles into a limit as the power of the Laplacian is increased (cf. Fig. 3.1(d) and (f)). This property offers a significant advantage in scattered node cases in showing that it is not particularly critical to decide on just what power of the Laplacian operator to use. For $\varepsilon \to 0$, the limit will agree with the optimal FD-based case.

3.3. Global RBF filter $-A^{-1}$

For finite domains in dD, numerical evidence strongly suggest that the A matrices will have $\binom{d-1}{0} = 1$ eigenvalue of size $O(\varepsilon^0)$, $\binom{d}{1} = d$ of size $O(\varepsilon^2)$, $\binom{d+1}{2}$ of size $O(\varepsilon^4)$, $\binom{d+2}{3}$ of size $O(\varepsilon^6)$, etc. [18] (assuming that the nodes are not located on some type of regular grid, in which case there will be fewer eigenvalues of each order, i.e. the conditioning of the A-matrix has been worsened). If the nodes are scattered along the periphery of a circle or on the surface of a sphere, the number of eigenvalues of sizes $O(\varepsilon^0)$, $O(\varepsilon^2)$, $O(\varepsilon^4)$, ... become instead $\{1,2,2,2,2,\ldots\}$ and $\{1,3,5,7,9,\ldots\}$, respectively. It can readily be observed that the corresponding eigenvectors (which we can think of as discrete representations of eigenfunctions) at the same time tend to become increasingly oscillatory in character. The matrix A^{-1} has exactly the same eigenvectors as A, but the eigenvalues are the inverses of those for A. In the case of positive definite radial functions, the eigenvalues of A^{-1} are therefore again all positive. However, assuming that ε is relatively small, successive ones will not decay towards zero but will instead grow increasingly rapidly in value. The ODE system $\frac{d\underline{u}}{dt} = -A^{-1}\underline{u}$ will therefore damp low order eigenfunctions slowly but high ones extremely fast. If one scales the right hand side by some small positive factor γ so that the ODE system

$$\frac{d\underline{u}}{dt} = -\gamma A^{-1}\underline{u} \tag{3.3}$$

only gently damps the high modes, the damping of a wide range of low modes will become barely perceptible. The effect will be very much like using a global approximation to a very high power of the Laplacian, but with advantages such as:

- Great simplicity of the damping operator; just apply the matrix A^{-1}, irrespective of node distribution, number of dimensions, type of (positive definite) radial function, etc.
- No decision is called for with regard to what power of the Laplacian Δ one should consider (as no Laplacian enters in the method).
- Guarantee that no mode can grow, no matter what ε-value is used.

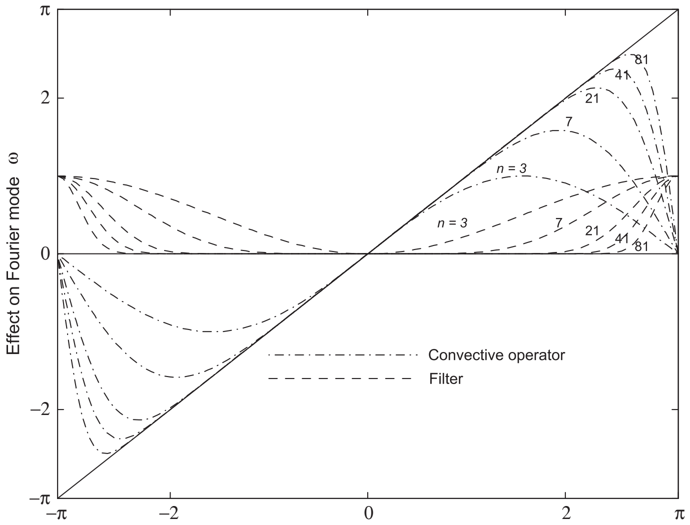


Fig. 3.2. Effect on different Fourier modes by centered FD approximations to $\partial/\partial x$ (dash-dot curves) and by the filter (dashed curves) in cases of stencils with $n = 3, 7, 21, 41, 81$ nodes.

It is possible to use this Global RBF filter approach in a local RBF-FD mode as well. If so, we apply (3.3) separately for each of the N nodes in the domain, and ignore all entries in the left hand side apart from the one corresponding to the local RBF-FD stencil's center point while we, in the right hand side, base A and u on all the n entries of the local stencil. Fig. 3.1 (g) and (h) show how this approach works in the same 7-node 1D test case studied for the two hyperviscosity approaches. In particular we see that, as $\varepsilon \to 0$, the effect will again approach that of the hyperviscosity methods. Although RBF expansion coefficients, eigenvalues of A^{-1}, etc. all diverge in this limit, the result nevertheless follows from the fact that RBF interpolants then converge to Lagrange's interpolation polynomial [3]. When ε is not pushed quite all the way to zero, this filter is seen to leave an even larger range of modes nearly intact, but the displayed curves then dip ever so slightly below the zero line, hinting at a possibility of undesired growth. When this Global RBF filter is used as first described (applied simultaneously to all N nodes as indicated in (3.3)), this cannot happen. This filter type is our preferred one in global RBF settings, whereas we will use the hyperviscosity approach (as described in Section 3.2) in RBF-FD settings.

3.4. Comparison between number of modes convected vs. number of modes damped when using FD methods

When using RBF-FD for a convective equation, we need to separately create two sets of stencils: (i) to approximate the convective operator, and (ii) to implement a matching hyperviscosity filter. The goal of the present section is to offer evidence that it is natural to use the same stencil sizes (values of n) for these two tasks.

With larger (i.e. more accurate) FD stencils for the convective operator, a wider range of modes will be treated accurately. The hyperviscosity approach can then also leave more modes intact before the damping of the high modes sets in. We compare here, again for simplicity on unit spaced grids in 1D, how the extent of 'well treated' modes grows with stencil size for convection vs. for filtering.

Regarding convection in 1D, we note that the mode $u(x) = e^{i\omega x}$ has the analytic derivative $u'(x) = i\omega e^{i\omega x}$. Disregarding the leading "i", the factor ω that appears in front of the exponential is displayed as the solid diagonal straight line in Fig. 3.2. Using standard FD approximations instead of analytic differentiation leads to the five dash-dot curves (cf. Fig. 4.1-3 in [9], and the discussion there). We see that the range of accurately treated modes grows only quite slowly with n (number of nodes in the stencil). However, several studies (surveyed in [9]) have nevertheless shown increases in n to be very cost-effective, in particular when increasing from $n = 3$ (second order accuracy) to the next several orders.

We contrast these curves with the corresponding ones for the optimal hyperviscosity (which is the same as for either of the two RBF-based filtering procedures in their $\varepsilon \to 0$ limit), here shown dashed. The second case $n = 7$ in Fig. 3.2 matches the 7-node case considered in Fig. 3.1. For the same stencil sizes, we see a good match between the range of modes that are convected accurately and the ones that the filter leaves intact. This suggests, as noted above, that one routinely can base the RBF-FD filter on the same local node set as what is used for the RBF-FD approximation to the convective operator.

The two sets of curves in Fig. 3.2 can be calculated either from the weights of their corresponding FD stencils, or via closed form expressions. Regarding the approximations to $\partial/\partial x$, see [19] or [9] (Eq. 4.1-4). The curves for the filter are given by $(\sin\frac{\omega}{2})^{n-1}$ (where n is assumed to be odd).

4. Global RBFs and RBF-FD on a spherical surface

In order to test numerical stability properties over long time integrations in cases for which no regular node layouts are available, we have chosen as our test problems two often used ones for convective flows over the surface of a sphere. For creating the RBF-FD filters, we use in all cases (3.2). This is a natural choice since, sufficiently locally, any smooth surface appears as nearly flat.

4.1. Node sets

We use in this study only MD (maximal determinant) node sets, as described in [29] and available for downloading from these authors' web site. In particular, we do not utilize local node refinement in order to meet spatially varying resolution requirements [5,18]. In case of interpolation with spherical harmonics, the MD node sets are much more stable numerically than the ME (minimal energy) type sets [29]. This is due to the MD nodes' subtle location irregularities, which can be seen in Fig. 4.1. This difference in stability is therefore also present for global RBFs in their $\varepsilon \to 0$ limit [14], but it becomes insignificant for larger ε-values. Tests that we have carried out show that such node location details are irrelevant also for the accuracy and robustness of the RBF-FD approach.

4.2. Eigenvalues of the filters

We consider next the filters on their own, without any associated convective operator. In Sections 5 and 6, we then test the stabilizing filters in combination with such operators.

4.2.1. Global RBF case

Fig. 4.2(a) illustrates the eigenvalues of the global filter DM, as obtained from the eigenvalues of $-A^{-1}$(based on all N nodes) in the case of $N = 900$, $\varepsilon = 2.5$. As implied by standard RBF theory, all the eigenvalues will be located on the negative real axis. The ones corresponding to low modes remain very close to the origin, while high ones are shifted far to the left (i.e. these highly oscillatory modes will be very effectively damped during time stepping).

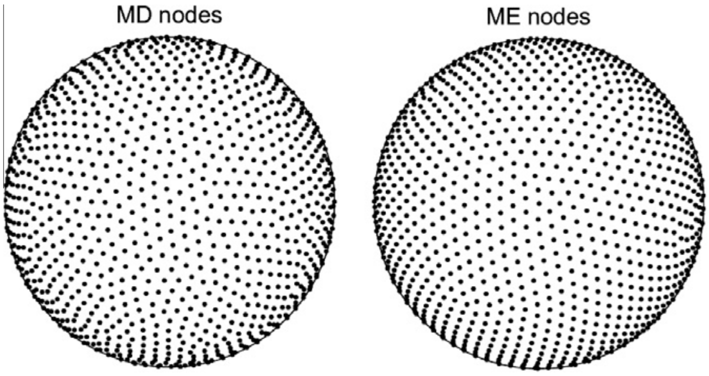


Fig. 4.1. Illustration of maximal determinant (MD) and minimal energy (ME) node sets in the case of $N = 1849$ nodes.

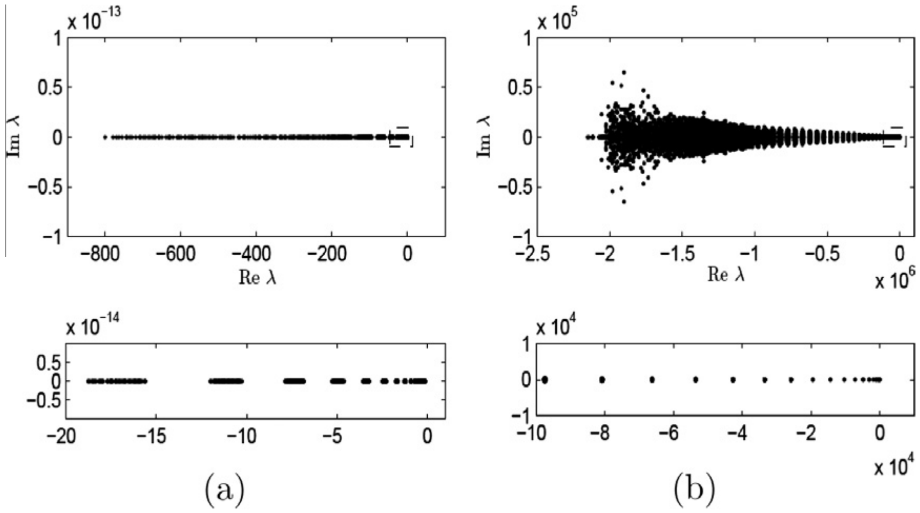


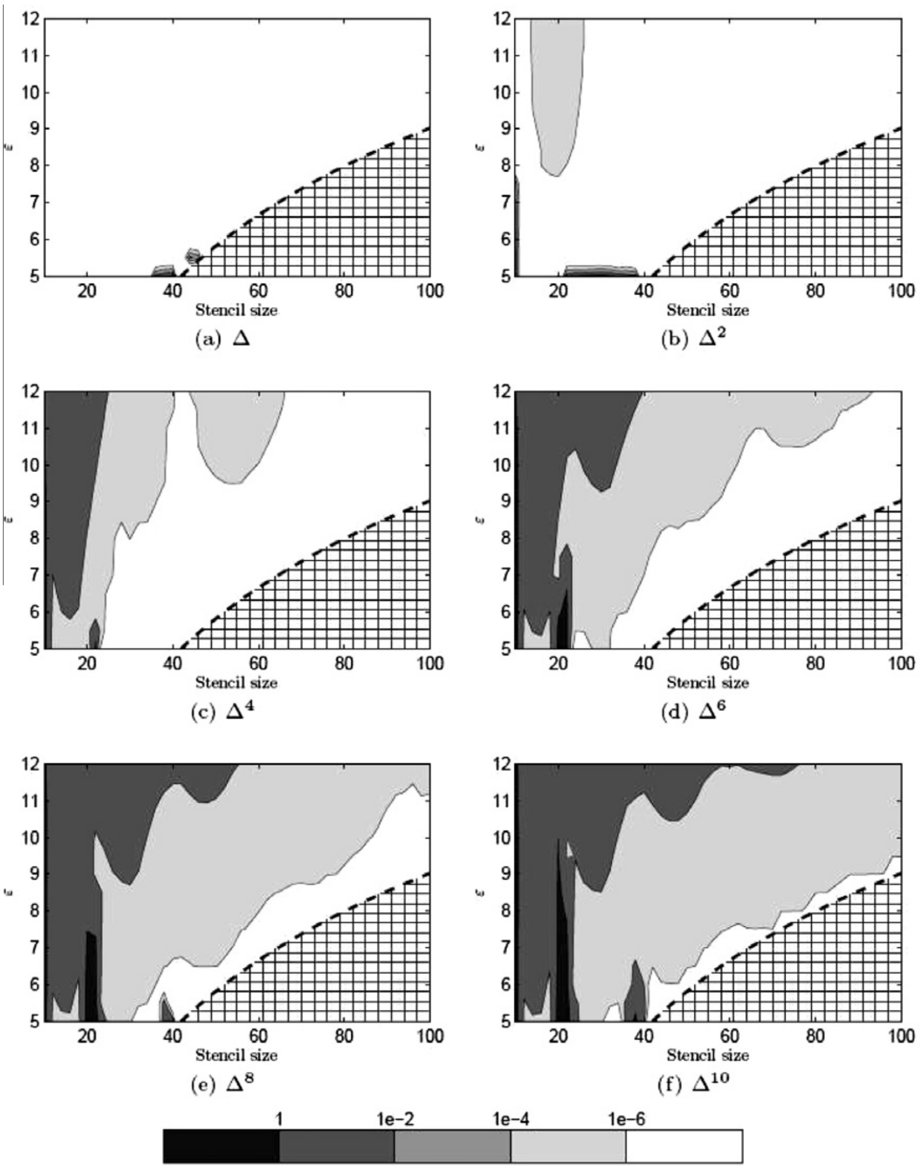
Fig. 4.2. (a) Eigenvalues of the DM corresponding to the global RBF filter ($N = 900$, $\varepsilon = 2.5$). (b) Eigenvalues of the DM corresponding to a local RBF-FD filter ($N = 3600$, $n = 17$, $\varepsilon = 2.5$). In both cases, the bottom subplot enlarges the area around the origin that is marked by a small rectangle in the upper plot.

4.2.2. RBF-FD case

From the perspective of computational cost (time and memory), the RBF-FD case is the one of most interest. The RBF-FD filter procedure is now applied to each of the N different n-node stencils separately, modifying only the data value at the stencil's center point. It was found by numerical tests (as noted at the end of Section 3.3) that the hyperviscosity approach then worked better than using a multiple of $-A^{-1}$ (based on the n nodes within each RBF-FD stencil). As Fig. 4.2(b) shows ($N = 3600$, $n = 17$, $\varepsilon = 2.5$, Δ^2-type hyperviscosity), the DM eigenvalue distribution again resembles the global case in that almost all modes are either very close to zero or well into the left half plane. However, in contrast to the global case, some of the eigenvalues far out in the left half plane have now also noticeable imaginary parts.

Fig. 4.3 illustrates for a relatively large node set on the sphere ($N = 25600$) how the maximum real part for the DM eigenvalues (obtained readily with Matlab's *eigs* routine) varies with n and ε when choosing different orders of hyperviscosity ($L = \Delta$, Δ^2, ..., Δ^{10}). Numerical ill-conditioning, when calculating the RBF-FD filter stencils, prevented reliable results in the cross-hatched areas in the bottom right corners of each subplot. The white areas show that, for lower powers of Δ, it is very easy to find combinations of n (stencil size) and ε such that no detrimental growth will occur. When higher powers of Δ are desired, an easy alternative to finding a good combination of ε and n is to apply a lower power of Δ twice in succession (however doubling the computational cost). Fig. 4.4 shows that filter stability then becomes nearly perfect.

The results in Fig. 4.3 suggest that the use of a numerically stable RBF algorithm (giving numerical access throughout the cross-hatched areas) may lead to excellent filters. It is likely that the RBF-RA algorithm [32], presently under development, will offer this capability. For general 1D, 2D and 3D domains, the RBF-QR algorithm [12] will resolve the conditioning issue.



(a) Δ

(b) Δ^2

(c) Δ^4

(d) Δ^6

(e) Δ^8

(f) Δ^{10}

Fig. 4.3. The maximum real part of the eigenvalues of the hyperviscosity DMs for different powers of Δ, displayed over an (n,ε)-plane. The eigenvalues are normalized such that $\min Re\lambda = -1$. In this figure, $N = 25600$ and the 2D formula is used. The strip at the bottom explains the different grey levels. White areas are entirely satisfactory for numerical use.

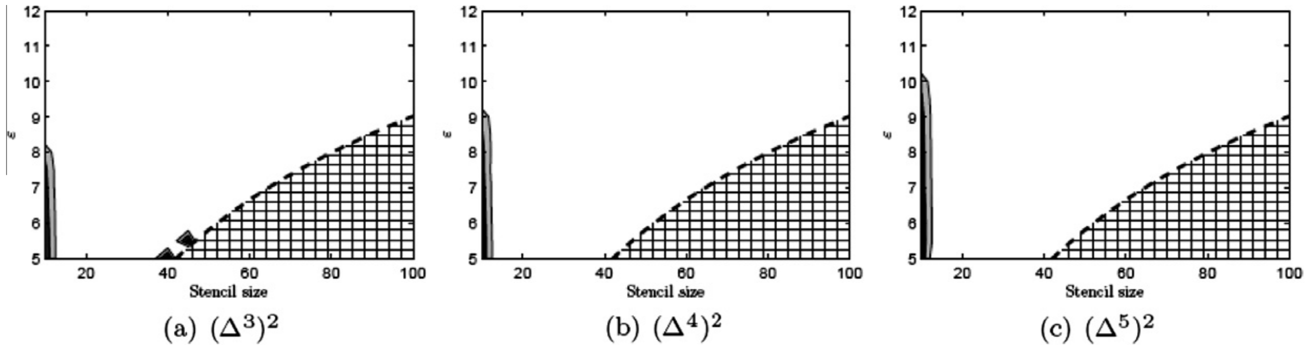


Fig. 4.4. The maximum real parts of the eigenvalues of the hyperviscosity DMs when the powers Δ^6, Δ^8, Δ^{10} are implemented as $(\Delta^3)^2$, $(\Delta^4)^2$, $(\Delta^5)^2$, respectively (i.e. at twice the cost by applying the FD stencils corresponding to the lower powers twice). The numerical values for the different grey levels are the same as in Fig. 4.3, i.e. the white areas are entirely satisfactory for numerical use.

It should be noted that the ε-value and the RBF type used for the filter need not have any relation to what is used in the corresponding RBF-FD stencil that approximates the convective operator. Although we only used GA in the present work, MQ (which is not positive definite) would most certainly have worked equally well for the convective part. If the PDE is linear and its coefficients are not time dependent, one can save operations by simply adding together the stencil weights for the convective operator and the filter operators before applying them. Otherwise, the RBF-FD filter would be applied as an additional right hand side term.

5. First test case: solid body rotation on a spherical surface

This standard PDE test problem describes convection around an axis that is inclined by the angle α relative to the polar axis (cf. [6,13,15,28]). Using spherical coordinates in terms of longitude φ and latitude θ

$$\begin{cases} x = \rho\cos\varphi\cos\theta, \\ y = \rho\sin\varphi\cos\theta, \\ z = \rho\sin\theta, \end{cases}$$

the governing PDE becomes

$$\frac{\partial h}{\partial t} + \frac{u}{\cos\theta}\frac{\partial h}{\partial\varphi} + v\frac{\partial h}{\partial\theta} = 0, \tag{5.1}$$

with

$$\begin{cases} u = \cos\theta\cos\alpha - \sin\theta\sin\varphi\sin\alpha, \\ v = -\sin\varphi\sin\alpha \end{cases} \tag{5.2}$$

describing the flow velocities. We have here assumed that the sphere has radius one, and a rotation rate for the convection such that one full revolution corresponds to the time $t = 2\pi$. As noted in [6], the singularity in (5.1) at the poles ($1/\cos\theta \to \infty$ as $\theta \to \pm\frac{\pi}{2}$) is purely an artifact of the singularity of the spherical coordinate system at these locations, and not of the convective process. Hence, these 'pole singularities' will automatically disappear *analytically* when the spatial operator is applied in the right hand sides of (2.2) and (2.4), i.e. the RBF-FD weights will not feature any pole-related irregularities. This is in sharp contrast to the situation when the spatial derivatives in (5.1) are approximated numerically. The singularity of $1/\cos\theta$ will then, at the poles, vastly amplify any errors in the $\frac{\partial h}{\partial\varphi}$ approximations. An additional disadvantage with (φ, θ)-based coordinates is that their (wastefully) high resolution in the φ-direction near the poles leads to a more severe CFL restriction.

We use for this test problem a cosine bell as the initial condition. When centered at (θ_c, φ_c), it is defined by

$$h = \begin{cases} \frac{1}{2}\left(1 + \cos\frac{\pi\rho}{R}\right) & \rho < R, \\ 0 & \rho \geqslant R, \end{cases} \tag{5.3}$$

where $R = 1/3$ and $\rho = \arccos(\sin\theta_c\sin\theta + \cos\theta_c\cos\theta\cos(\varphi - \varphi_c))$. This is a C^1-function, featuring a jump in the second derivative around its base. It is commonly preferred as a test function over for example Gaussian bells since (i) it is compact in extent, and (ii) its does not give very high order/spectral methods potentially misleading advantages (that are dependent on extreme test function smoothness).

5.1. Eigenvalues of the convection operator

5.1.1. Global RBF case

We consider first the global case, in order to provide a reference to compare the RBF-FD approach against. Fig. 5.1(a) shows the eigenvalues (EVs) in the case of $\varepsilon \to 0$. In the present case of $N = 900$, i.e. $N = m^2$ nodes with $m = 30$, there are

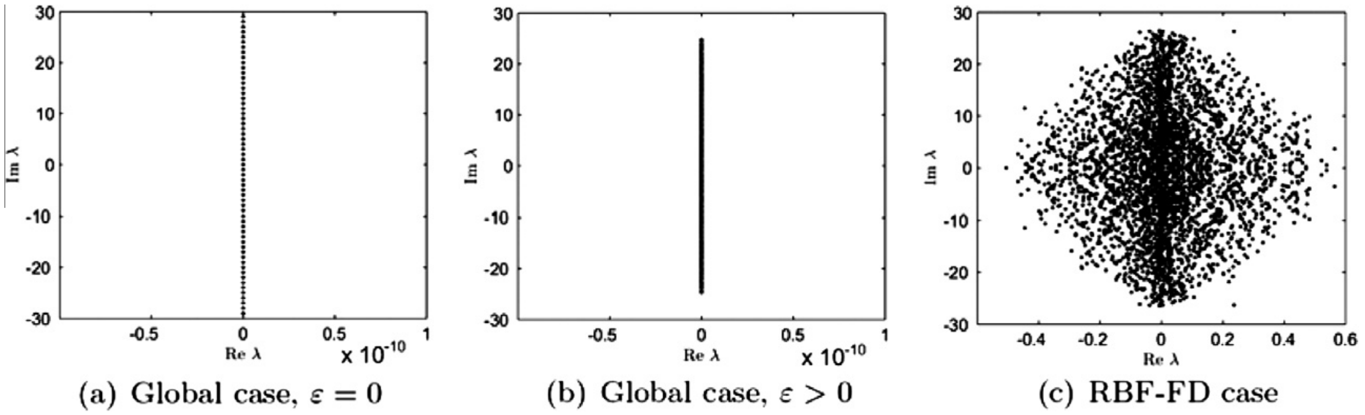


Fig. 5.1. DM eigenvalues for the convective operator. (a) and (b) Global RBF approximation with $N = 900$, $\varepsilon = 0$ and $\varepsilon = 5$ resp., (c) RBF-FD case, $N = 3600$, $n = 17$, $\varepsilon = 2.5$.

m EVs at 0, $m - 1$ at each of $\pm i$, $m - 2$ at each of $\pm 2i$, etc. until 1 at each of $\pm(m - 1)i$ [15]. For positive ε, these EVs get perturbed but, in this solid body convection problem, they remain exactly on the imaginary axis although no longer only at integer multiples of i [6,21]; cf. Fig. 5.1(b). Low modes remain very accurate, but high modes feature errors that will become purely dispersive during time stepping.

5.1.2. RBF-FD case

The DM will again be accurate for low modes, but errors for higher modes will no longer be purely dispersive. Fig. 5.1(c) illustrates a typical case for which many high modes have been scattered quite far into the two half planes. The eigenvalues in the left half plane will be naturally decaying when the PDE is advanced in time, but the ones in the right half plane will need to be suppressed by the RBF-FD filter. Fig. 5.2 shows how the convection operator's spectrum has changed after the RBF-FD filter has been applied (by adding its DM to that of the convective operator, scaled with the magnitude of the local velocity; Figs. 5.1(c) and 5.2 were obtained with $N = 3600$, $n = 17$, $\varepsilon = 2.5$ and Δ^2-type hyperviscosity). We recognize here a large number of low modes in quite accurate positions along and just to the left of the imaginary axis. All the offending spurious modes, previously in the right half plane, have now moved to the left half plane where most standard numerical time stepping methods will naturally damp them out. In both this and the next test case, we used the most standard fourth order Runge–Kutta (RK4) algorithm, choosing the time steps sufficiently small that the temporal errors would not exceed the spatial ones.

5.2. Computations with the filtered RBF-FD approach

Fig. 5.3 shows the numerical solution to the solid body rotation test case and the magnitude of the error after 1, 10 and 1000 revolutions, with the RBF-FD approach using $N = 25600$ MD nodes, $n = 74$, $\varepsilon = 8$ and Δ^8-type hyperviscosity (implemented

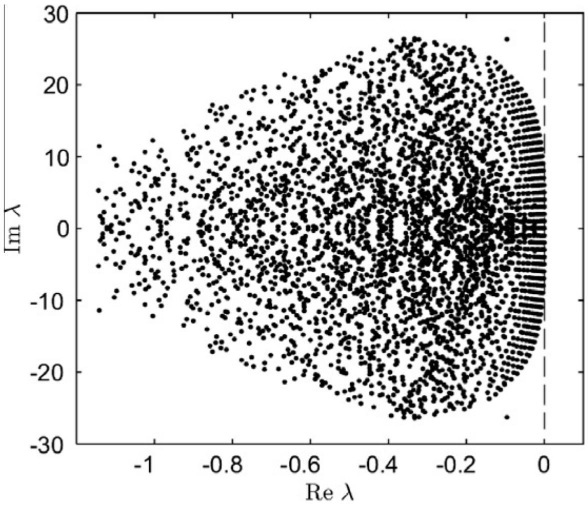


Fig. 5.2. Eigenvalues of the same convective operator as illustrated in Fig. 5.1(c), but after a RBF-FD hyperviscosity operator (of same stencil size) has also been applied at all node points. Note the very different scales along the two axes.

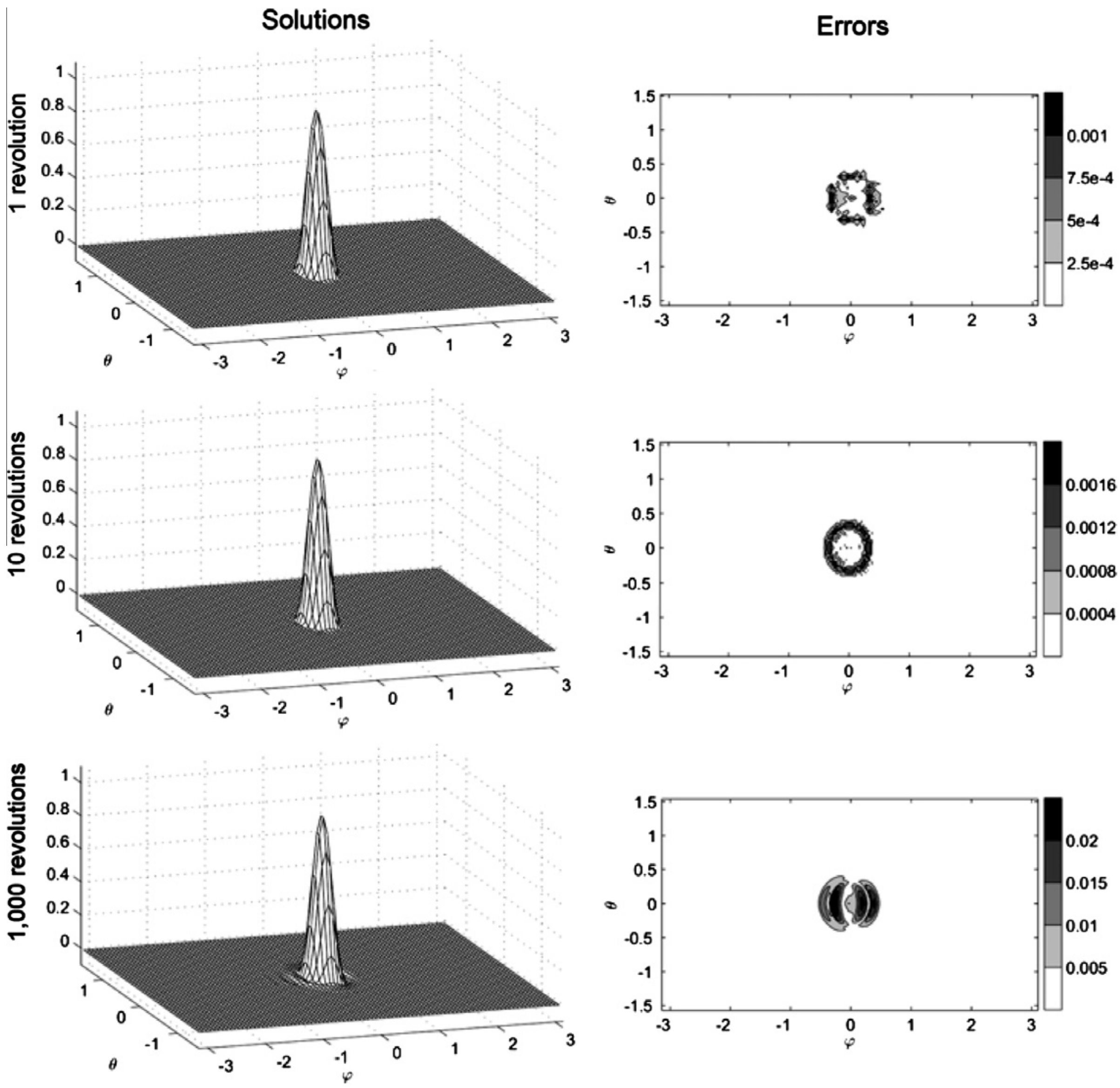


Fig. 5.3. The numerical solution and the magnitudes of the errors for the solid body rotation test case, using the filtered RBF-FD approach.

directly as Δ^8 rather than as $(\Delta^4)^2$). Even after 1000 revolutions, the cosine bell is intact and essentially without any dispersive wave train. The errors are still mainly confined to where the initial condition has a discontinuity in the second derivative.

Fig. 5.4 shows a log–log plot of the ℓ_2 error against N. When applying RBF-FD filter, the power of the Laplacian was here increased gradually with the stencil size, from Δ^2 for $n = 17$ to Δ^{10} for $n = 101$. In order to avoid ill-conditioning as N increases, ε was scaled inversely to the node spacing, keeping the condition number of the local RBF matrices roughly constant. Since the PDE's initial condition has a discontinuous second derivative, the highest achievable convergence rate is second order. The ℓ_2 error of $O(1/N)$ seen in Fig. 5.4 is consistent with second order convergence, since the node spacings in the MD sets are inversely proportional to $\sqrt{N}$. The proportionality constant for a given N is seen to improve significantly with increasing n (stencil size).

The filtered RBF-FD approach requires a strategy for choosing a suitable value for γ, i.e. the amount of hyperviscosity to apply. The suitable range is limited by two factors:

- γ too large
  - eigenvalues far out in the left half-plane, making the time step stability limited rather than accuracy limited,
  - damping also of lower modes, thereby degrading the accuracy.
- γ too small
  - time stepping instabilities due to eigenvalues in the right half-plane.

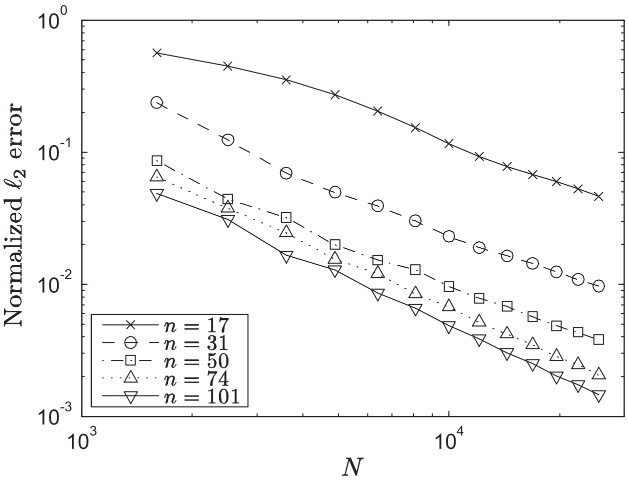


Fig. 5.4. The normalized ℓ_2 error after 10 revolutions, as a function of N, for the solid body rotation test case.

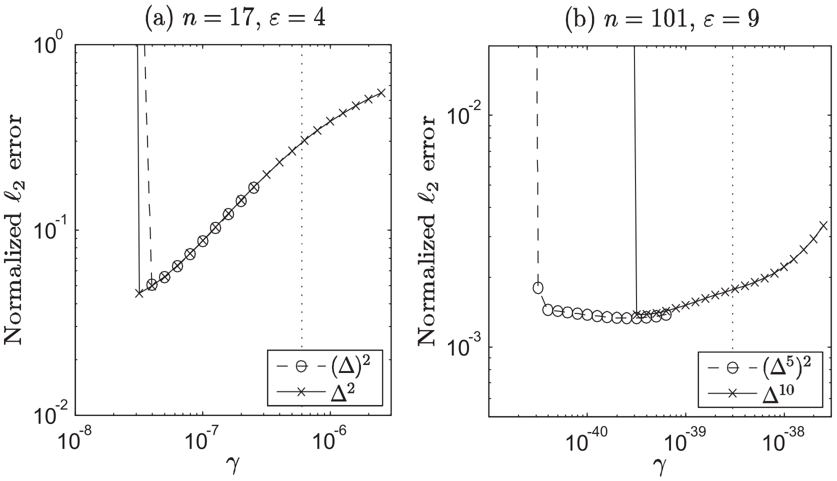


Fig. 5.5. Influence of the choice of γ on the final accuracy after 10 revolutions when using $N = 25600$ nodes on the sphere. The vertical dotted lines mark where γ has become so large that the time step in the RK4 scheme becomes limited by stability rather than by accuracy when using Δ^2 and Δ^{10} resp. When instead using $(\Delta)^2$ and $(\Delta^5)^2$, RK4 is accuracy limited in the range shown by the circular markers. (a) Small stencil, $n = 17$, $\varepsilon = 4$. (b) Large stencil, $n = 101$, $\varepsilon = 9$.

We illustrate these effects by displaying how the resulting accuracy varies with γ. Considering for $N = 25600$ the $n = 17$ and $n = 101$ cases shown at the right edge (top and bottom curves) of Fig. 5.4, we obtain the results in Fig. 5.5. As this shows, the good range for γ is quite wide. For larger stencil sizes, the range becomes wider still if Δ^{2p} is split into $(\Delta^p)^2$. There are several options for identifying suitable γ-values:

- Trial and error in very short test runs. The sharp lower cutoff becomes clear already after a very brief time integration.
- Trial and error for the same values for ε and n, but with much lower values of N. Numerical experiments suggest that a good strategy is to scale γ as $\gamma \sim N^{-2p}$ when using Δ^{2p} or $(\Delta^p)^2$-type hyperviscosity.
- Calculate an approximation for the eigenvalue with the largest real part using an iterative eigenvalue routine for sparse matrices. This approach can be very fast, but some such routines (e.g. Matlab's 'eigs') occasionally fail to converge.

6. Second test case: vortex rollup

The stationary vortex roll-up test case is an example of a deformational flow that models idealized cyclogenesis. It was introduced in [20], and has since been implemented with several methods, including global RBFs [5,6]. The governing PDE is again given by (5.1), but with the wind field (5.2) replaced by

$$\begin{cases} u &= \omega(\theta)\cos\theta, \\ v &= 0. \end{cases}$$

Here, $\omega(\theta)$ is the angular velocity, given by

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

where $\rho(\theta) = \rho_0 \cos\theta$ and ρ_0 is a parameter controlling the radial extent of the vortex. This is again a test problem with a known analytic solution:

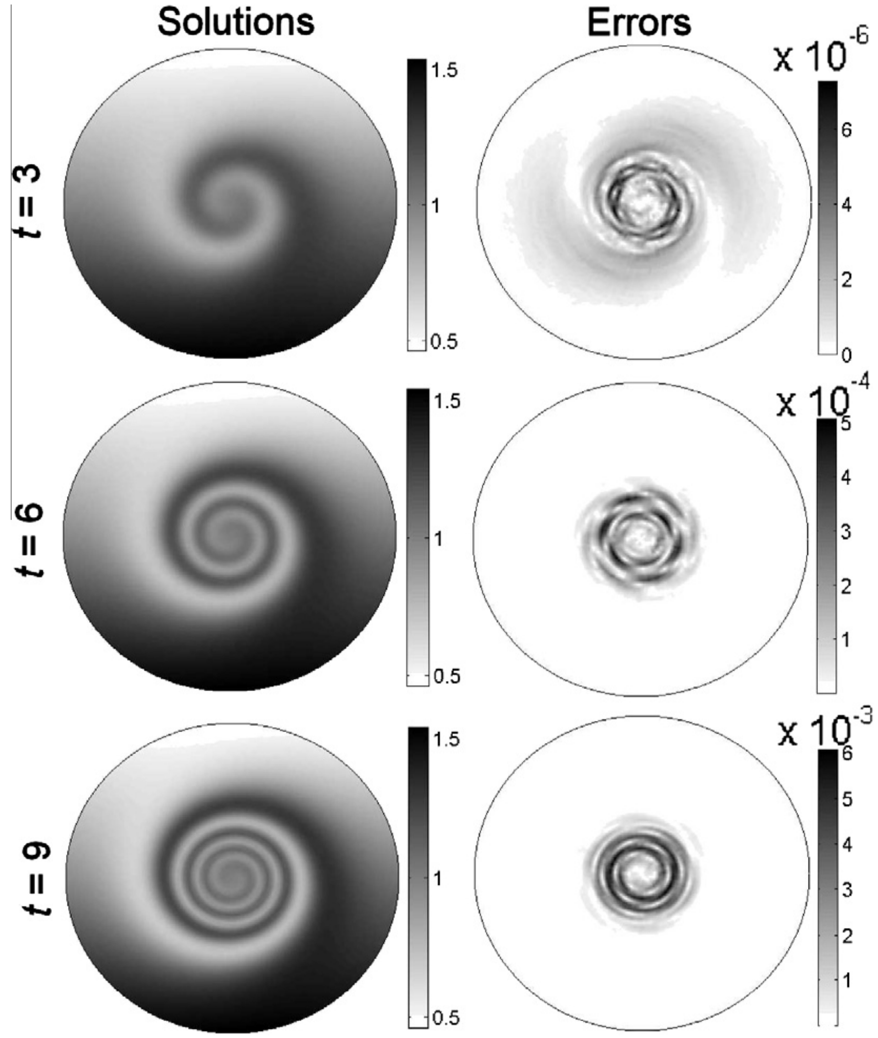


Fig. 6.1. The numerical solution and the magnitude of the errors for the vortex roll-up test case at various times. Here, $N = 25600$ and $n = 50$.

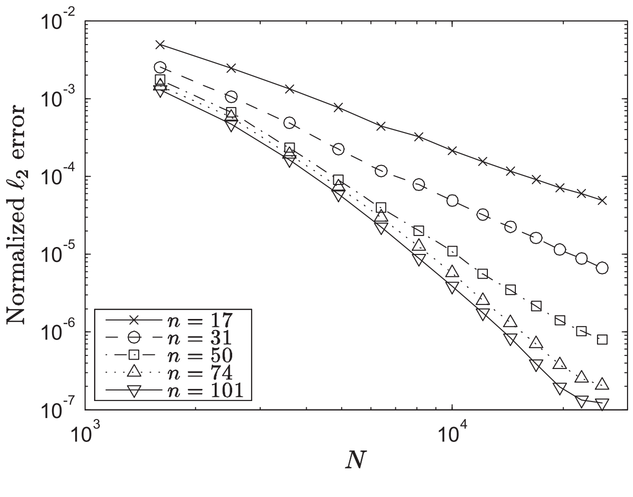


Fig. 6.2. The normalized ℓ_2 error at $t = 3$ as a function of N for the vortex roll-up test case.

$$h(\varphi,\theta,t) = 1 - \tanh\left(\frac{\rho(\theta)}{\zeta}\sin(\varphi - \omega(\theta)t)\right),$$

where ζ is the characteristic width of the frontal zone. To simplify comparisons with other results in the literature, we choose $\rho_0 = 3$ and $\zeta = 5$. The solution to this test problem develops increasingly fine structures as time progresses.

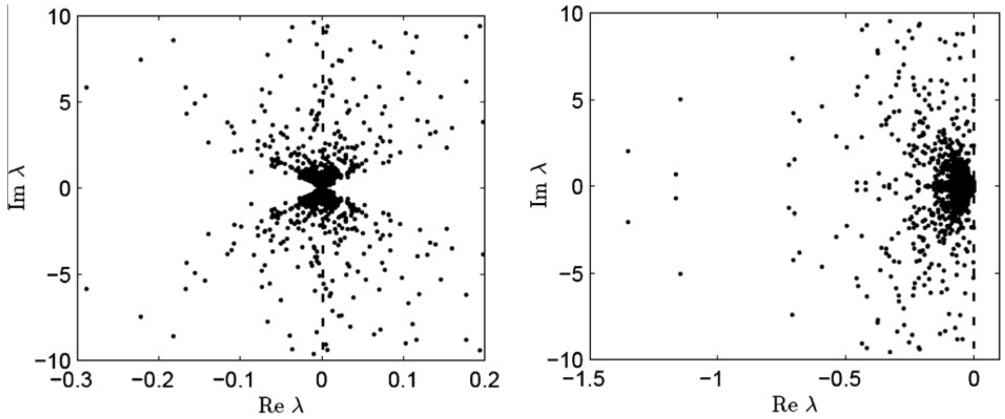


Fig. 6.3. Eigenvalues of (a) the global DM and (b) the filtered DM for the vortex roll-up test case. The imaginary axis is marked by a dashed line.

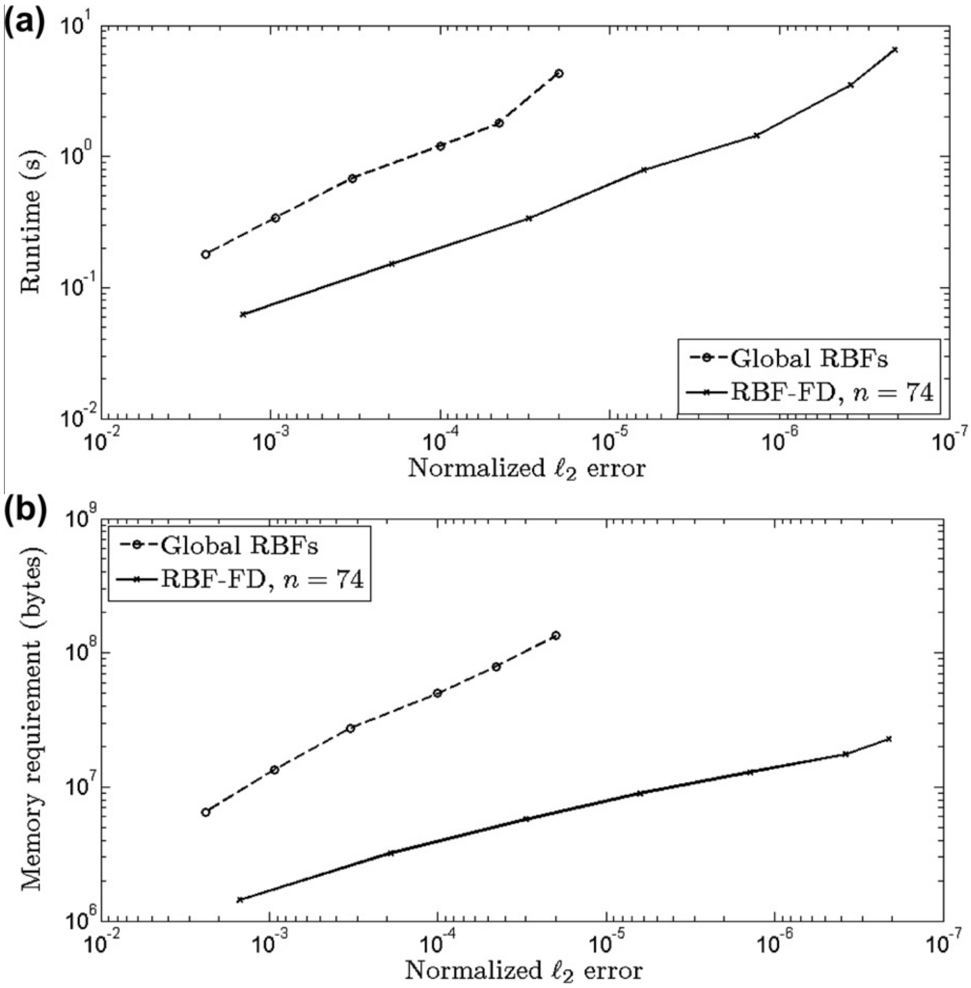


Fig. 6.4. The computational costs of the filtered RBF-FD method and the global RBF method for the vortex rollup test case. (a) Run time to reach $t = 3$ when using a single processor 1.83 GHz PC. (b) Memory requirement.

6.1. Convergence and time stability of RBF-FD implementation

Fig. 6.1 shows the numerical solution and the magnitude of the error at $t = 3$, 6 and 9. Since we use a relatively uniform MD node set (rather than refine locally near the vortex center), the errors naturally become the largest where the gradients are the largest. However, even at $t = 9$, the errors are of the order 10^{-3}, and the numerical solution is visually indistinguishable from the exact one. Like for the first test problem, RBF-FD without filtering would not work well because of a large number of rapidly growing spurious modes.

Fig. 6.2 shows the normalized ℓ_2 error as a function of N. Since the solution in this test remains infinitely differentiable for all times, the RBF-FD convergence rate increases as n gets larger (as seen through the increasing slopes of the curves in Fig. 6.2). It reaches about 8th order for $n = 101$.

6.2. Comparison to global RBFs

The global RBF differentiation matrix for this test case has some eigenvalues with positive real part. Fig. 6.3 illustrates this, and shows also how the global A^{-1}-type filter moves all eigenvalues to the imaginary axis or into the left half plane. Even though the original DM has unstable eigenmodes, the smooth initial condition contains only very small amounts of them, and they do not grow fast enough to become a serious issue until after our end time of $t = 9$. Although required in the RBF-FD case, filtering is unnecessary in this test problem when using global RBFs and only integrating over relatively short times.

As we have noted several times already, our interest in RBF-FD methods is motivated by their lower computer time and memory requirements. Fig. 6.4 displays these quantities as functions of the error. While the global RBF methods are more accurate for a given N, they are significantly more costly than RBF-FD methods for given error levels, with increasing advantages for the (filtered) RBF-FD approach when the accuracy requirements are increased.

As shown in [5], locally refined node sets greatly improve the effectiveness of global RBF methods, and will quite certainly do the same also for RBF-FD methods. This has however not been tested here.

7. Conclusions

RBF-FD operators are fast to generate and very fast to apply. They can offer high accuracy and great geometric flexibility. Thanks to their local nature, they are likely to prove very well suited for implementations on multiprocessor computer systems. In the context of solving PDEs by the MOL approach, the primary uncertainty has in the past concerned the numerical stability of the approach, especially for convective type PDEs that do not offer any natural dissipation that is helpful against temporal growth of spurious modes. We have here developed a filter approach that is related to traditional hyperviscosity and which can be applied quickly in any number of dimensions. It has here been tested on two standard convective PDEs test problems for flows on the surface of a sphere. In both cases, the stabilized RBF-FD approach proved to be highly competitive against the global RBF method, which in turn (for the same test problems) was extensively compared against a number of other methods in [5,6]. We have also devised a simple filter approach for global RBF discretizations.

Acknowledgements

The work of Bengt Fornberg was supported by the NSF Grants DMS-0611681, DMS-0914647 and ATM-0620068. Erik Lehto was supported by the NSF grant ATM-0620100. Discussions with Natasha Flyer (NCAR) and with Grady Wright (Boise State University) are gratefully acknowledged.

References

[1] V. Borue, S.A. Orszag, Numerical study of three-dimensional Kolmogorov flow at high Reynolds numbers, J. Fluid Mech. 306 (1996) 293–323.
[2] G. Chandhini, Y.V.S.S. Sanyasiraju, Local RBF-FD solutions for steady convection–diffusion problems, Int. J. Numer. Meth. Eng. 72 (2007) 357–378.
[3] T.A. Driscoll, B. Fornberg, Interpolation in the limit of increasingly flat radial basis functions, Comput. Math. Appl. 43 (2002) 413–422.
[4] N. Flyer, B. Fornberg, Radial basis functions: developments and applications to planetary scale flows, Comput. Fluids, in press, doi:10.1016/j.compfluid.2010.08.005.
[5] N. Flyer, E. Lehto, Rotational transport on a sphere: local node refinement with radial basis functions, J. Comput. Phys. 229 (2010) 1954–1969.
[6] N. Flyer, G.B. Wright, Transport schemes on a sphere using radial basis functions, J. Comput. Phys. 226 (2007) 1059–1084.
[7] N. Flyer, G.B. Wright, A radial basis function method for the shallow water equations on a sphere, Proc. R. Soc. Lond. Ser. A 465 (2009) 1949–1976.
[8] B. Fornberg, A numerical study of 2D turbulence, J. Comput. Phys. 25 (1977) 1–31.
[9] B. Fornberg, A Practical Guide to Pseudospectral Methods, Cambridge University Press, Cambridge, 1996.
[10] B. Fornberg, Calculation of weights in finite difference formulas, SIAM Rev. 40 (3) (1998) 685–691.
[11] B. Fornberg, N. Flyer, J.M. Russell, Comparisons between pseudospectral and radial basis function derivative approximations, IMA J. Numer. Anal. 30 (2010) 149–172.
[12] B. Fornberg, E. Larsson, N. Flyer, Stable computations with Gaussian radial basis functions, SIAM J. Sci. Comput., submitted for publication.
[13] B. Fornberg, D. Merrill, Comparison of finite difference- and pseudospectral methods for convective flow over a sphere, Geophys. Res. Lett. 24 (1997) 3245–3248.
[14] B. Fornberg, C. Piret, A stable algorithm for radial basis functions on a sphere, SIAM J. Sci. Comput. 30 (2007) 60–80.
[15] B. Fornberg, C. Piret, On choosing a radial basis function and a shape parameter when solving a convective PDE on a sphere, J. Comput. Phys. 227 (2008) 2758–2780.

[16] B. Fornberg, G. Wright, Stable computation of multiquadric interpolants for all values of the shape parameter, Comput. Math. Appl. 48 (2004) 853–867.
[17] 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.
[18] B. Fornberg, J. Zuev, The Runge phenomenon and spatially variable shape parameters in RBF interpolation, Comput. Math. Appl. 54 (2007) 379–398.
[19] H.O. Kreiss, J. Oliger, Comparison of accurate methods for the integration of hyperbolic equations, Tellus 24 (1972) 199–215.
[20] R.D. Nair, J. Côté, A. Staniforth, Cascade interpolation for semi-Lagrangian advection over the sphere, Quart. J. Meteor. Soc. 125 (1999) 1445–1468.
[21] R.B. Platte, T.A. Driscoll, Eigenvalue stability of radial basis function discretizations for time-dependent problems, Comput. Math. Appl. 51 (2006) 1251–1268.
[22] Y.V.S.S. Sanyasiraju, G. Chandhini, Local radial basis function based grid free scheme for unsteady incompressible viscous flows, J. Comput. Phys. 227 (2008) 99–113.
[23] R. Schaback, Multivariate interpolation by polynomials and radial basis functions, Constr. Approx. 21 (2005) 293–317.
[24] Y.Y. Shan, C. Shu, Z.L. Lu, Application of local MD-DQ method to solve 3D incompressible viscous flows with curved boundary, Comp. Model. Eng. Sci. 25 (2008) 99–113.
[25] C. Shu, H. Ding, K.S. Yeo, Local radial basis function-based differential quadrature method and its application to solve two dimensional incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 192 (2003) 941–954.
[26] D. Stevens, H. Power, M. Lees, H. Morvan, The use of PDE centers in the local RBF Hermitean method for 3D convective-diffusion problems, J. Comput. Phys. 228 (2009) 4606–4624.
[27] H. Wendland, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, Adv. Comput. Math. 4 (1995) 389–396.
[28] 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.
[29] R.S. Womersley, I. Sloan, How good can polynomial interpolation on the sphere be?, Adv Comp. Math. 14 (2001) 195–226.
[30] G.B. Wright, N. Flyer, D.A. Yuen, A hybrid radial basis function-pseudospectral method for thermal convection in a 3D spherical shell, Geochem. Geophy. Geosyst. 11 (2010) (Article number Q07003).
[31] G.B. Wright, B. Fornberg, Scattered node compact finite difference-type formulas generated from radial basis functions, J. Comput. Phys. 212 (2006) 99–123.
[32] G.B. Wright, B. Fornberg, An algorithm for stable computations with flat radial basis functions, submitted for publication.
[33] Z. Wu, Compactly supported positive definite radial functions, Adv. Comput. Math. 4 (1995) 283–292.