Periodized Radial Basis Functions (RBFs) and RBF-Vortex Method for the Barotropic Vorticity Equation

by

Jianping Xiao

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Atmospheric and Space Sciences and Scientific Computing)
in The University of Michigan
2014

Doctoral Committee:

Professor John P. Boyd, Chair
Professor Robert Krasny
Associate Professor Christiane Jablonowski
Research Scientist Gabor Toth

For my parents

ACKNOWLEDGEMENTS

This thesis could not have been finished without the help from a number of persons to whom I owe a great debt of gratitude, and I would like to thank them for their valuable contributions.

I am especially grateful to my advisor Professor John P. Boyd, who supervised my study and research work during the past several years at the University of Michigan. Professor Boyd's wide knowledge in applied mathematics, atmospheric science, oceanic science and computational science, as well as logical way of thinking have been of great value to me. Having the opportunity to work with him has truly been an honor.

I would like to thank Prof. Robert Krasny, Prof. Christiane Jablonowski and Dr. Gabor Toth for serving on my thesis committee. A high quality thesis like this would not have been possible without their insightful suggestions and careful proofread. I would like to thank the AOSS staff and IT group for their support over the years of study and research in our department. Without their help, my work could not have progressed so smoothly.

Last but not least, I want to thank my family and friends both in the United States and China for their constant support during my years as a graduate student.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Periodized Radial Basis Functions and RBF-Vortex Method for the Barotropic Vorticity Equation

by

Jianping Xiao

Chair: John P. Boyd

Fluids spontaneously develop fronts, narrow spiral filaments and other features of rapid spatial variation which are very challenging for numerical methods. Like most competing numerical schemes, Radial Basis Function (RBF) methods are based on interpolation. It has been previously proved that the RBF approximation converges to the correct solution as the number of grid points increases. When the flow is varying rapidly, high accuracy requires a high density of interpolation points while smooth regions require a lower density of points. A method that can adaptively allocate more grid points to where the fronts develop and fewer grid points to where the flow is smooth is of great value in fluid simulation on the surface of a sphere. In this thesis, a method that combines the meshfree nature of RBF interpolation and the Lagrangian particle method is developed. On the one hand, the particles serving as fluid elements are advected by the velocity field such that rapidly varying regions are densely populated; on the other hand, the particles serving as RBF centers provide higher density of interpolation points and therefore give a better resolution of the regions.

CHAPTER I

Introduction

1.1 Radial Basis Functions (RBFs)

The approximation to a function using radial basis functions(RBFs), in any number of dimensions d, is

$$f(\vec{x}) \approx \sum_{j=1}^{N} a_j \phi(\parallel \vec{x} - \vec{c_j} \parallel),\ \vec{x} \in R^d \tag{1.1}$$

for some "kernel" $\phi(r)$, always a *univariate* function even when the dimension $d > 1$, and some set of N points $\vec{c_j}$, which are called the "centers". The symbol $||\ ||$ denotes the usual distance norm, $||\vec{x}|| = \sqrt{x_1^2 + x_2^2 + \ldots x_d^2}$. The coefficients a_j are always found by interpolation at a set of points $\vec{x_i}$, which are usually chosen to coincide with the centers. The coefficients are computed by solving the matrix problem

$$\vec{\vec{V}}\vec{a} = \vec{f} \tag{1.2}$$

where matrix $\vec{\vec{V}}$ is the interpolation matrix with entries $\vec{\vec{V}}_{i,j} = \phi(\parallel \vec{x_i} - \vec{x_j} \parallel)$; $\vec{f}$ is the vector whose entries are $f(\vec{x_i})$.

Interpolation using Radial Basis Functions (RBFs) first appears in the literature as a method for interpolating scattered topographic data (*Hardy*, 1971). Kansa

(*Kansa*, 1990a,b) first used RBFs to solve Partial Differential Equations (PDEs), where the partial derivative is approximated by taking the partial derivative of the basis functions. In recent two decades, Radial Basis Functions are growing in popularity for solving partial differential equations (*Barba and Rossi*, 2010; *Fedoseyev et al.*, 2002; *Moroney and Turner*, 2007; *Platte and Driscoll*, 2005; *Wong et al.*, 2002; *Flyer and Wright*, 2009) and have proved very useful in computer graphics and neural networks (*Fasshauer*, 2007; *Buhmann*, 2003; *Wendland*, 2005; *Iske*, 2004; *Buhmann*, 2000; *Schaback and Wendland*, 2006).

To solve a partial differential equation(PDE) problem, the procedure is similar to interpolation. Suppose the differential equation is

$$\mathcal{L}f(x) = g(x) \tag{1.3}$$

where $\mathcal{L}$ is a linear operator. The approximation is $f_N(x) = \sum_{i=1}^{N} a_i \phi(\| x - x_i \|)$. The discretization is

$$\vec{\vec{L}}\vec{a} = \vec{g} \tag{1.4}$$

Here, the elements of the discretization matrix $\vec{\vec{L}}$ are generated by applying the operator $\mathcal{L}$ on the basis functions,

$$\vec{\vec{L}} = \begin{bmatrix} \mathcal{L}\phi(\| x - x_1 \|)|_{x=x_1} & \dots & \mathcal{L}\phi(\| x - x_N \|)|_{x=x_1} \\ \vdots & \ddots & \vdots \\ \mathcal{L}\phi(\| x - x_1 \|)|_{x=x_N} & \dots & \mathcal{L}\phi(\| x - x_N \|)|_{x=x_N} \end{bmatrix}$$

and $\vec{g}$ is the vector whose elements are $g(x_i)$ at grid points x_i, with $i = 1, ..., N$. The interpolation coefficients are

$$\vec{a} = \vec{\vec{L}}^{-1}\vec{g} \tag{1.5}$$

1.1.1 Shape parameters, saturation error and other issues

Table 1.1 lists Five RBF species that are infinitely differentiable. All of the RBF kernels (Gaussian, Inverse Quadratic, Hyperbolic Secant, Multiquadric and Inverse Multiquadric) contain a parameter which controls the width of the basis function. This parameter, which is really the inverse width of the basis function, is commonly

Table 1.1: The five RBF kernels: Gaussian, Inverse Quadratic, Hyperbolic Secant, Multiquadric and Inverse Multiquadric

RBF names	RBF kernels
Gaussian	$\exp(-\epsilon^2 x^2)$
Inverse Quadratic	$1/(1+\epsilon^2 x^2)$
Hyperbolic Secant	$2/[\exp(\epsilon x)+\exp(-\epsilon x)]$
Multiquadric	$\sqrt{1+\epsilon^2 x^2}$
Inverse Multiquadric	$1/\sqrt{1+\epsilon^2 x^2}$

called the "shape" parameter. For theoretical purposes, it is convenient to work with an "absolute shape parameter" ϵ; for numerical purposes, it is more convenient to use a parameter α which is *relative* to the grid spacing, $h = 2\pi/N$, where

$$\epsilon \equiv \frac{\alpha}{h} = \frac{\alpha N}{2\pi} \tag{1.6}$$

Thus, the Gaussian kernel is

$$\phi(x) = \exp\left(-\epsilon^2 x^2\right) = \exp\left(-\frac{\alpha^2 N^2}{4\,\pi^2} x^2\right) \tag{1.7}$$

In applications, it is typical to choose the relative width parameter α for Gaussian RBFs to lie in the range 0.2 to 0.5, and similarly but with different ranges for the other RBF species discussed here. When $\alpha \to 0$, RBF interpolation error increases exponentially because of the ill-conditioning of the interpolation matrix. However, Fornberg proposed several strategies to do stable computations in this limit (*Driscoll and Fornberg*, 2002; *Fornberg and Wright*, 2004). When α is large, RBFs approximate

smooth functions as a sum of delta functions, which is obviously inaccurate. For moderately large α, the error will decrease exponentially with increasing number of grid points N but then plateau and decrease no further. This phenomenon is called "error saturation" as shown in Fig. 1.2. These perils of large and small α are avoided by choosing the relative shape parameter to be in the range 1/5 to 1/2, typically, as already indicated. In this moderate range, it has been proven that RBFs converge exponentially fast for approximating smooth functions. Fig. 1.1 shows the Gaussian kernel with three different shape parameters.

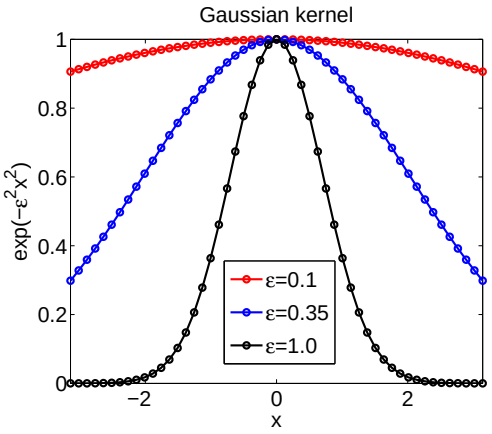

Figure 1.1: Gaussian kernel with different shape parameters. As the shape parameter ϵ becomes smaller, the Gaussian function becomes flatter. In the limit $\epsilon = 0$, the Gaussian function is a constant.

Although interpolation of smooth functions using RBFs are spectrally accurate, we have to solve a dense matrix problem, which is very expensive especially for large number of grid points. The RBF Finite Difference (RBF-FD) method proposed in (*Flyer et al.*, 2012; *Fornberg and Lehto*, 2011) reduces a dense interpolation matrix to a sparse matrix by computing the FD weights on the n nearest neighbors for all the grid points. The computational effort is reduced using RBF-FD method, but the accuracy of the method is not spectral. Another strategy to solve the linear dense matrix problem is to use iterative methods to find an approximate solution. However, in many cases, the dense matrices are very ill-conditioned indicating a

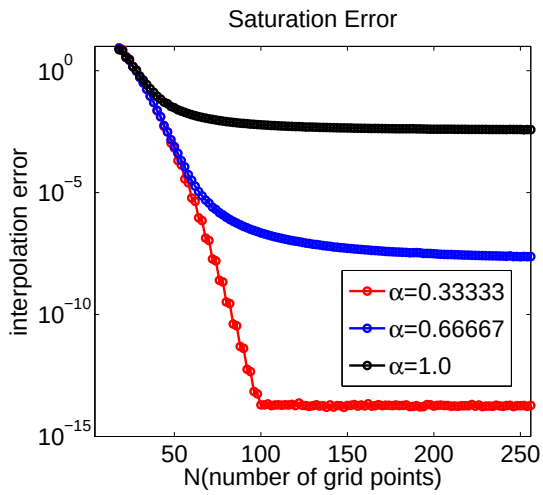

Figure 1.2: Saturation error of Gaussian RBF interpolation using different α. The interpolation error approaches a error plateau. In this graph, $f(x) = (\sin(3x)+5\cos(x))\exp(-5\cos(2x)+0.25\sin(x))$ is interpolated on $[-\pi, \pi]$ using Gaussian RBF.

slow convergence even when iterative methods are applied. This is because a larger $\lambda_{max}/\lambda_{min}$ usually has a slower convergence and a smaller $\lambda_{max}/\lambda_{min}$ indicates a faster convergence. Various preconditioners have been devised to assure fast RBF interpolations using iterative methods (*Torres and Barba*, 2009; *Brown et al.*, 2005; *Ling and Kansa*, 2004; *Baxter*, 2002; *Beatson et al.*, 1999; *Diago et al.*, 2004).

1.1.2 RBF methods in geoscience

Radial Basis Functions neural networks have been extensively used in weather predictions, climate and geophysical data analysis(*Hickernell and Hon*, 1998; *Santhanam and Subhajini*, 2011; *El-Shafie et al.*, 2012; *Binaghi et al.*, 2013; *Zhang et al.*, 2011; *Panakkat and Adeli*, 2007; *Shiguemori et al.*, 2008; *Paes et al.*, 2011). In the field of geophysical fluid dynamics, RBFs have been widely used, too. Recent development on geophysical flow simulation using RBF methods have been advanced by Fornberg and his colleagues (*Flyer and Wright*, 2009, 2007; *Fornberg and Piret*, 2008; *Wright et al.*, 2010; *Flyer and Lehto*, 2010; *Flyer et al.*, 2012). Specifically, RBFs methods are applied to solve Shallow Water Equations (SWE), transport equations on the

surface of the sphere as well as thermal convection equations in a 3-D spherical shell. The experiments are performed on a near-uniform grid either with or without local refinement. Compared with other methods (Spectral elements, Spherical harmonics) on the surface of the sphere, RBF methods use a much smaller number of nodes and longer time steps while achieve higher accuracy as discussed in (*Flyer and Fornberg*, 2011).

1.2 Vortex methods

The vorticity is defined as the curl of the velocity:

$$\vec{\omega} = \nabla \times \vec{u} \tag{1.8}$$

For incompressible flow, the velocity must satisfy the continuity equation:

$$\nabla \cdot \vec{u} = 0 \tag{1.9}$$

This condition can be satisfied by introducing a vector streamfunction $\vec{\psi}$ such that the velocity is defined as:

$$\vec{u} = \nabla \times \vec{\psi} \tag{1.10}$$

We can check $\nabla \cdot \vec{u} = \nabla \cdot (\nabla \times \vec{\psi}) = 0$. If the vector streamfunction is divergence-free, then

$$\vec{\omega} = \nabla \times \nabla \times \vec{\psi} = \nabla(\nabla \cdot \vec{\psi}) - \nabla^2 \vec{\psi} = -\nabla^2 \vec{\psi} \tag{1.11}$$

If the flow is on a surface only, then $\vec{\psi} = (0, 0, \psi)$ and $\vec{\omega} = (0, 0, \omega)$, we have

$$\nabla^2 \psi = -\omega \tag{1.12}$$

The conservation of vorticity following the flow can be written down as

$$\frac{d\omega}{dt} = \frac{\partial \omega}{\partial t} + \vec{u} \cdot \nabla \omega = 0 \tag{1.13}$$

which can be derived from the Eulerian equations as in (*Beale*, 1988). On the surface of a rotating sphere, the absolute vorticity consists of two parts: the relative vorticity ζ and the coriolis parameter $2\Omega \cos(\theta)$. Therefore, the conservation of absolute vorticity on the surface of a rotating sphere can be written as

$$\frac{d\left(\zeta + \cos(\theta)\right)}{dt} = 0 \tag{1.14}$$

where 2Ω is set to unity. Eqn. 1.14 is the Barotropic Vorticity Equation (BVE). For Eqn. 1.13 and Eqn. 1.14, a Lagrangian method is natural. The vortex method discretizes the vorticity field into Lagrangian elements that carry vorticity and move with the local fluid velocity.

The vortex method first appeared in the classic work by Rosenhead (*Rosenhead*, 1932), where he approximated the discontinuous velocity surface by a set of point vortices. The point vortex method approximates a continuous flow by a set of point vortices that travel with the flow. The circulation of a vortex is concentrated at a point. The velocity field at a given location is the superposition of the velocities induced by all other point vortices. Although there is an extensive literature on point vortex methods on the sphere (*Sakajo*, 2007a,b, 2004; *Pekarsky and Marsden*, 1998; *Newton and Sakajo*, 2008, 2006, 2007; *Newton and Ross*, 2006; *Klyatskin and Reznik*, 1989; *Kimura and Okamoto*, 1987; *Kidambi and Newton*, 1998b,a,c; *Jamaloodeen and Newton*, 2007; *Dibattista and Polvani*, 1998; *Bogomolov*, 1979, 1977), point vortices were abandoned for vortex blob methods except for computations on the surface of a sphere. On the one hand, this is because the streamfunctions of the point vortices have singularities at the locations of the point vortices. As vortices move close enough,

the velocities of the vortices become large and the simulations blow up in finite time. On the other hand, a sum of Dirac delta point vortices is a poor approximation to a smooth, continuous vorticity field. In geophysical fluid dynamics, point vortex models have not been taken seriously as a numerical method for simulating geophysical flows. However, due to the theoretical simplicities of point vortex methods, there are extensive publications on fluid dynamics on a rotating sphere using point vortex methods (*Reznik and Kizner*, 2007; *Reznik*, 1992, 1986; *Kuhlbrodt and Nevic*, 2000; *Hogg and Stommel*, 1985; *Gryanik*, 1988b,a, 1986; *McWilliams and Zabusky*, 1982).

In point vortex methods, the circulation concentrates on a point vortex. Unlike point vortex methods, the vortex blob methods (*Chorin and Bernard*, 1973) allow the vorticity to have a distribution that decays to zero, for example a Gaussian distribution. In the Gaussian vortex blob method, the fractional contribution from vortex i to location $\vec{x}$ is

$$\zeta_i(\vec{x}) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(\vec{x}-\vec{x}_i)^2}{2\sigma^2}\right) \tag{1.15}$$

The vorticity at location $\vec{x}$ is sum of the contribution from all the vortex blobs, that is,

$$\omega(\vec{x}) = \sum_{i=1}^{N} \Gamma_i \zeta_i(\vec{x}) \tag{1.16}$$

where Γ_i is the strength of the vortex.

The vortex blob methods were first introduced by *Chorin and Bernard* (1973). In the vortex blob methods, the continuous flow is represented as a sum of many vortex blobs. The convergence of simulations using vortex blob methods were provided in (*Beale and Majda*, 1982a; *Hald and Del Prete*, 1978; *Hald*, 1979, 1987). Beale & Majda studied and developed higher order kernels for vortex methods in two and three dimensions (*Beale and Majda*, 1982b, 1985). Numerical experiments of the 2-D Euler equations indicated that the higher order kernels yield a considerably more accurate

representation of the velocity field than the lower order kernels. Krasny analytically and numerically showed that the vortex blob methods converged as the inter-particle spacing converged to zero and as the smoothing parameter converged to zero (*Krasny*, 1986), for periodic vortex sheets roll-up simulations. General discussion of vortex methods can be found in (*Cottet and Koumoutsakos*, 2000; *Majda and Bertozzi*, 2002). Comprehensive review articles of flow simulations using vortex methods are found in (*Leonard*, 1980; *Koumoutsakos*, 2005).

Although the vortex blob methods provided convergent and accurate simulations for fluid flows, for long time simulations the accuracy is degraded by the vortex distortion problem. This is because the accuracy of the vortex blob method is very sensitive to the overlap ratio h/σ, where σ is the Gaussian blob width and h is the vortex distance (*Perlman*, 1985; *Barba et al.*, 2003). As the particles travel with the flow, due to strain, the vortex blobs will become sparsely populated in some areas while cluster together in other areas. If the ratio h/σ is too large, it means that the continuous flow field is not well covered and represented by the vortex blobs, and therefore simulation error arises. The cure to the vortex distortion problem is by vortex remeshing (*Barba et al.*, 2003, 2005), where after some time steps the particle circulations on a more regular grid are re-calculated. The prevailing remeshing scheme on a Cartesian coordinate consists in constructing a square lattice of new particles locations, and obtaining the circulation values from the old particles by interpolation using tensor-product kernels.

Adding viscosity to a vortex model is another difficulty in vortex methods. The first attempt to add viscous effects was by *Chorin* (1973). The diffusion of the vorticity was simulated via the Brownian motion of the vortex particles. This method is based on the statistical interpretation of the diffusion equation. However the random walk technique only applies to slightly viscous flow. The core spreading method simulates the diffusion effect by expanding the vortices. The idea behind this is that the

Gaussian function with growing core size is an exact solution to the diffusion equation. However, it was shown that the core spreading method does not converge to the Navier-Stokes equations (*Greengard*, 1985). A third damping method is the Particle Strength Exchange (PSE) method, which approximates the diffusion by smoothing the difference between the target particle and the surrounding particles using a Gaussian kernel (*Degond and Mas-Gallic*, 1989; *Koumoutsakos*, 1997; *Koumoutsakos and Leonard*, 1995; *Ploumhans and Winckelmans*, 2000). However, the PSE method relies heavily on the accuracy of the integration schemes, and therefore on the regularity of the particle locations.

The meshfree nature of RBF methods has been applied to do the vortex regridding/remeshing in fluid simulations using vortex methods (*Barba et al.*, 2003, 2005; *Barba and Rossi*, 2010; *Torres and Barba*, 2009). In these works, the Gaussian RBFs are used to find the circulations of the new particles via solving an interpolation problem given the vorticity of the old particles. Compared with other tensor-product interpolation kernels, the RBF interpolation does not introduce a sharp increase at the initial remeshing step, and keeps the simulation highly accurate for a long period of time. This is because vortex regridding/remeshing using tensor-product kernels brings back the idea of "mesh", i.e., a tensor-product grid, for a "meshless" method, i.e., the vortex methods. While the Radial Basis Functions are "meshless" kernels. They do not rely on the regularity of the grid, although best interpolations are achieved on a near-uniform grid. Accurate interpolations are usually possible using RBF kernels on an irregular grid.

CHAPTER II

Periodized Radial Basis Functions

2.1 RBF basics and imbricate series

2.1.1 Motivation

Previous studies on RBF methods focused on nonperiodic domains. Very little work has been done on RBF interpolation on a periodic domain and solving PDEs using Periodized Radial Basis Functions (PRBFs). The trigonometric basis has been the default choice on a periodic domain. However when applied to resolve very narrow sharp structures in a wide domain, much more grid points are needed on a uniform grid than on a nonuniform adaptive grid. Although a standard interpolation on a nonuniform grid using trigonometric basis is possible, the condition number of the interpolation matrix is very large, which results in a very inaccurate solution. In Periodic RBF interpolation, we can appropriately manipulate the free shape parameter ϵ to supress the ill-conditioning of the interpolation matrix. This makes accurate interpolation using PRBF on a highly nonuniform grid possible.

2.1.2 A review of previous work on periodic RBFs

We have found only three papers, all highly theoretical, which focus on periodic RBFs: the 1992 articles of Xu & Cheney (*Xu and Cheney*, 1992) and Light & Cheney

(*Light and Cheney*, 1992) and the 2006 article of Hubbert & Mueller (*Hubbert and Mueller*, 2006). RBFs are usually chosen to be Strictly Positive Definite (SPD). These authors establish conditions for a periodic RBF to be Strictly Positive Definite (SPD) or Almost Strictly Positive Definite (ASPD) and then collectively demonstrate the following for general periodic RBF:

1. A Strictly Positive Definite $\theta(x)$ ["kernel"] has a Fourier cosine series with *non-negative* coefficients.

2. An SPD basis function is symmetric, that is, $\theta(-x) = \theta(x)$.

3. The interpolation matrix for a periodic but otherwise arbitrary grid is *always invertible*, and the interpolation problem therefore always has a unique solution.

4. The interpolation matrix for a periodic uniform grid is circulant.

5. The circular matrix can be diagonalized by a similarity transformation using the Fourier transform matrix.

6. Series for the Fourier coefficients of RBF cardinal functions are found.

Using special formulas for circulant matrices, the eigenvalues of the interpolation matrix can be derived for a number of species of periodic RBFs. In particular, the condition number of the interpolation matrix for Periodic Inverse Quadratic (PIQ) functions is $p^{-N/2}$ where p is the positive constant such that the Fourier coefficients of the PIQ kernel $\theta^{IQ}(x;\epsilon)$ are $2p^n$. Both Periodic Gaussian (PGA) and the PIQ functions are Strictly Positive Definite. Hubbert and Mueller derive a closed form expression for the interpolation error in the least squares norm for the interpolation of $f(x) = \cos(x)$ by the PIQ basis. All these authors knew the closed form for PIQ, but none discussed the imbricate series for PIQ and PGA which we give below.

Platte and Driscoll (*Platte and Driscoll*, 2006), although focusing on a wider range of RBF issues, prove that periodic RBF differentiation matrices are stable provided

that the RBFs are positive definite. Abe and Iiguni (*Abe and Iiguni*, 2006a,b) have applied periodic Gaussian RBFs to neural networks for image processing and show that, for a uniform grid with N points, the coefficients can be computed in $O(N \log_2(N))$ operations. However, their data sets are not periodic and their accuracy is not spectral.

There are three reasons why the literature on periodic radial basis functions is sparse. First, periodic domains are rarer in applications than nonperiodic intervals. Second, Fourier pseudospectral methods are very fast due to the blazing speed of the Fast Fourier Transform for non–adaptive periodic problems. Third, when the RBFs are narrow compared to the periodicity interval, the RBFs closely resemble their infinite interval counterparts except near the edges of the interval $x \in [-\pi, \pi]$.

Nevertheless, we shall show that periodic RBFs are interesting objects with good value for applications.

2.1.3 Poisson Summation and imbricate series

Theorem II.1 (Imbricate Series/Poisson Summation)**.** *Any periodic function with period L has the two series representations*

$$f(x) = \chi \sum_{n=-\infty}^{\infty} g(\chi\, n)\ \exp(i\ 2\ \pi\ n\ x\ /L\ ) = \sum_{m=-\infty}^{\infty} G\left(\frac{2\ \pi}{L\ \chi}[x - m\ L\ ]\right) \qquad (2.1)$$

where the series on the left is the usual (complex) Fourier series, the series on the right is "imbricate" series, χ is an arbitrary positive constant, and where $g(k)$ and $G(x)$ are Fourier transforms of one another:

$$G(x) = \int_{-\infty}^{\infty} g(k)\ \exp(i\ k\ x)\ dk \qquad (2.2)$$

$$g(k) = \frac{1}{2\,\pi} \int_{-\infty}^{\infty} G(x)\ \exp(-\,i\,k\,x)\,dx \tag{2.3}$$

The theorem is valid if the following sufficient conditions are satisfied: (i) $g(k) \in \mathcal{L}^1(R)$*, that is, if the integral of the absolute value of* $g(k)$ *over the entire real axis is bounded and also (ii)* $G(x) = O\left((1+|x|)^{-\alpha}\right)$ *as* $|x| \to \infty$ *for some* $\alpha > 1$*.*

Proof. (Zayed, 1996, pg. 239). The theorem is a special case of the Poisson Summation Formula. □

Definition II.2 (Pattern Function)**.** The function $G(x)$, which is repeated with even spacing over all x to create a periodic function, is said to be the "pattern" function of the imbricate series.

Definition II.3 (Imbrication)**.** The periodic function $f(x)$ which is the sum of the imbricate series with pattern function $G(x)$ is said to be the "imbrication" of $G(x)$.

Fig. 2.1 is a schematic that illustrates the simple idea behind an imbricate series. Given an arbitrary function $G(x)$ which decays sufficiently fast as $|x| \to \infty$, duplicate an infinite number of copies and place one copy at $x = m\ L$ where m ranges over all the positive and negative integers and L is an arbitrary constant. The result will, by construction, be a periodic function. If $G(x)$ decays sufficiently fast — decay as $O(1/x^2)$ will suffice although this condition can be considerably relaxed — then the imbricate series can be easily proved to converge.

2.1.4 Erf-Unitary Function for periodic functions

Lemma II.4 (Erf-Unitary Function)**.** *The function*

$$\Omega(x) = \frac{1}{2}\{erf(\frac{\pi - x}{2S}) + erf(\frac{\pi + x}{2S})\} \tag{2.4}$$

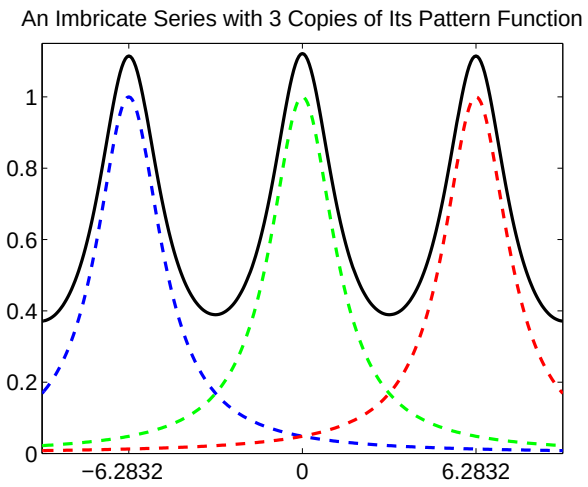

Figure 2.1: Schematic of an imbricate series. Thick, unmarked curve: the periodic function $f(x)$ which is the sum of the imbricate series. Dashed: Three of the infinite number of copies of the pattern function $G(x)$.

is a Periodic Partition-of-Unity, in the sense that

$$\sum_{m=-\infty}^{\infty} \Omega(x - 2\pi m; S) = 1, \forall x \tag{2.5}$$

for all values of $S > 0$*, a constant.*

Proof. Boyd (*Boyd*, 1997). □

Lemma II.5. *If* $f(x)$ *is a periodic function, then it may always be represented as the "imbricate" series*

$$f(x) = \sum_{m=-\infty}^{\infty} G(x - 2\pi m; S), \forall x \tag{2.6}$$

where the "pattern function" is

$$G(x) = f(x)\Omega(x; S) \tag{2.7}$$

with Ω *a Periodic Partition-of-Unity (Boyd, 1989).*

Proof.

$$\begin{aligned} G(x-2\pi m) &= f(x-2\pi m)\Omega(x-2\pi m;S) \\ &= f(x)\Omega(x-2\pi m) \end{aligned} \tag{2.8}$$

because $f(x)$ is periodic with periodic 2π. Since $f(x)$ is now a common factor for every term in the imbricate sum, it may be extracted from the sum:

$$\begin{aligned} f(x) &= \sum_{m=-\infty}^{\infty} G(x-2\pi m;S) \\ &= \sum_{m=-\infty}^{\infty} f(x-2\pi m)\Omega(x-2\pi m;S) \\ &= \sum_{m=-\infty}^{\infty} f(x)\Omega(x-2\pi m;S) \\ &= f(x)\sum_{m=-\infty}^{\infty} \Omega(x-2\pi m;S) \\ &= f(x) \end{aligned} \tag{2.9}$$

where we invoked the Periodic Partition-of-Unity property of Ω in the last line. □

The important thing here is, every periodic function, i.e., with period 2π, can be written as an imbricate series. In the next section, the periodized version of RBFs will be given using Poisson Summation theorem. Two forms of the Periodized RBFs will be given, i.e., imbricate series and Fourier series.

2.2 Periodizing RBFs

2.2.1 Construction of basis functions through the Poisson Summation Theorem

Any function $\phi(x)$ which decays as fast as $O(1/|x|^2)$ as $|x| \to \infty$ can be periodized by being made the "pattern function" of an "imbricate series" of the form

$$\theta(x) \equiv \sum_{m=-\infty}^{\infty} \phi(x - 2\pi m) \tag{2.10}$$

The function θ, which is the sum of an infinite number of copies of the pattern function, is by construction a periodic function of period 2π. The Poisson Summation Theorem shows that the Fourier coefficients of θ are the Fourier transform $\Phi(k)$ of the pattern function $\phi(x)$.

The three RBF species we shall discuss here are all good basis sets for approximating a function $G(x)$ on the real axis. One way of interpreting and justifying their periodic equivalents is to regard the RBF approximation (without periodization) as an approximation to the pattern function $G(x)$; applying the periodization process to both the pattern function and to the individual terms of the RBF approximation generates an approximation for the periodic function $f(x)$ as a series of the periodic RBFs. That is to say,

$$G(x) = \sum_{n=1}^{N} a_n \phi(x - x_n) \tag{2.11}$$

$$\Updownarrow$$

$$f(x) = \sum_{n=1}^{N} a_n \theta(x - x_n) \tag{2.12}$$

where $\theta(x) \equiv \sum_{m=-\infty}^{\infty} \phi(x - 2\pi m)$ is the periodic RBF and

$$f(x) \equiv \sum_{m=-\infty}^{\infty} G(x - 2\pi m) \tag{2.13}$$

In the following subsections, the periodized version of Radial Basis Functions will be generated.

2.2.2 Periodized Gaussians (PGA)

$$\phi(x) = \exp\left(-\epsilon^2 x^2\right) \tag{2.14}$$

where ϵ is the absolute inverse width or "shape parameter". The Periodic Gaussian (PGA) is

$$\begin{aligned} \theta^{GA}(x;\epsilon) &= \sum_{m=-\infty}^{\infty} \exp\left(-\epsilon^2[x - 2\pi m]^2\right) \\ &= \frac{1}{2\,\epsilon\,\sqrt{\pi}} \left\{1 + 2\sum_{n=1}^{\infty} \exp\left(-\frac{n^2}{4\epsilon^2}\right)\cos(nx)\right\} \\ &= \frac{1}{s\sqrt{\pi}}\,\theta_3(x/2;\epsilon) \end{aligned} \tag{2.15}$$

where $\theta_3(z;s)$ is the usual Jacobian Theta function defined below.

In practice, of course, the infinite imbricate series must be truncated. For the Periodic Gaussian basis, the "pattern function" decays so rapidly that usually just three terms suffice. Thus the PGA basis is quite efficient even though the Jacobian theta function cannot be evaluated in closed form. The Periodic Inverse Quadratic function [next subsection] has an imbricate series that converges very slowly; this basis is practical because a simple, explicit closed form for the basis-defining sum is

known.

2.2.3 Periodic Inverse Quadratic (PIQ)

$$\phi(x) = \frac{1}{1 + \epsilon^2 x^2} \tag{2.16}$$

imbricates to the PIQ kernel:

$$\begin{aligned} \theta^{IQ}(x; N, \epsilon) &\equiv \sum_{m=-\infty}^{\infty} \frac{1}{(1 \ + \ \epsilon^2(x \ - \ 2\pi \ m \ )^2)} \\ &= \frac{1}{2 \ \epsilon} \left\{ 1 \ + \ 2 \ \sum_{n=1}^{\infty} \ \exp(-n/\epsilon) \ \cos(n \ x) \right\} \\ &= \frac{1}{2 \ \epsilon} \frac{(1 - p^2)}{(1 + p^2) - 2 \ p \ \cos(x)} \end{aligned} \tag{2.17}$$

where

$$p = \exp(\ -1/\epsilon) \tag{2.18}$$

2.2.4 Periodized Sech (Psech)

$$\phi(x) = \mathrm{sech}(\epsilon x) \tag{2.19}$$

periodizes as the Periodic Hyperbolic Secant (Psech) kernel:

$$\begin{aligned} \theta^{sech}(x; N, \epsilon) &\equiv \sum_{m=-\infty}^{\infty} \mathrm{sech}[\epsilon(x - 2\pi m)] \\ &= \frac{1}{\epsilon} \left\{ (1/2) + 2 \sum_{n=1}^{\infty} [q^n/(1 + q^{2n})] \cos(nx) \right\} \\ &= \frac{2K(\sqrt{1 - k^2})}{\pi} \mathrm{dn} \left(\frac{2}{\pi} K(\sqrt{1 - k^2}) \epsilon x; k \right) \end{aligned} \tag{2.20}$$

where the required auxiliary relationships are

$$\epsilon \equiv \pi / \left[2 \log(1/q) \right] \Leftrightarrow \text{elliptic nome}\, q = \exp\left(-\frac{\pi}{2\,\epsilon} \right) \tag{2.21}$$

$$k \ \text{[elliptic modulus]} = EllipticModulus(q) \equiv \left(\frac{\theta_2(0;q)}{\theta_3(0;q)} \right)^2 \tag{2.22}$$

where dn is the usual Jacobian elliptic function, K is the complete elliptic integral and $\theta_2(x;q)$ and $\theta_3(x;q)$ are standard Jacobian theta functions. The necessary elliptic function and theta function can be found in (*Olver et al.*, 2010) and its online equivalent, the NIST Digital Library of Mathematical Functions (dllmf.nist.gov; see also functions.wolfram.com).

Since Sech functions are exponentially decaying functions, in numerical computations, we can truncate the summation series so that the computational effort is reduced significantly.

2.2.5 Periodic generalizations of the multiquadric (MQ) and inverse multiquadric (IMQ)

Imbricating the kernels

$$\phi = \sqrt{1 + \epsilon^2 r^2} \qquad \text{[Multiquadric (MQ)]} \tag{2.23}$$

and

$$\phi = \frac{1}{\sqrt{1 + \epsilon^2 r^2}} \qquad \text{[Inverse Multiquadric (IMQ)]} \tag{2.24}$$

fails because the resulting series are divergent due to the slow decay (or lack of decay) of the kernels. However, these kernels do have generalized Fourier Transforms:

$$\mathcal{FT}(\sqrt{1+\epsilon^2 x^2}) = -\frac{K_1(|K|/\epsilon)}{\pi|K|} \qquad \text{[MQ]} \tag{2.25}$$

$$\mathcal{FT}\left(\frac{1}{\sqrt{1+\epsilon^2 X^2}}\right) = -\frac{K_0(|K|/\epsilon)}{\pi\epsilon} \qquad \text{[IMQ]} \tag{2.26}$$

By following the same procedure as before, the periodization of MQ and IMQ would be the summation of the corresponding Fourier series over the integer wave numbers only, as long as the summation is convergent. However, the corresponding series have zeros mean. We experimented with adding the constant as a separate basis function, but a better option proved to be arbitrarily adding a constant to the Fourier series and thus defining

$$\theta^{MQ}(x;\epsilon) \equiv 1 + \sum_{n=1}^{\infty} K_1(n/\epsilon)/(\pi n)\ \cos(nx) \tag{2.27}$$

Similarly,

$$\theta^{IMQ}(x;\epsilon) \equiv 1 + \sum_{n=1}^{\infty} K_0(n/\epsilon)/(\pi\epsilon)\ \cos(nx) \tag{2.28}$$

Figs. 2.3 and 2.4 show that both proposed basis sets, despite the lack of a standard imbricate series, give spectral accuracy. Fig. 2.2 confirms the expected geometric rate of convergence, that is, the error is falling as $\exp(-qN)$ for some positive q, which appears as a straight line on this graph with log and linear scales.

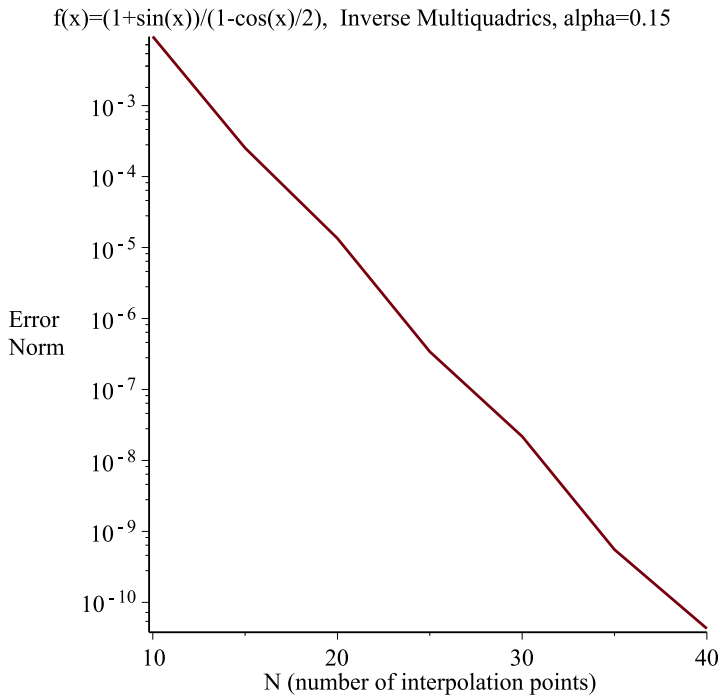

Figure 2.2: Periodic Inverse Multiquadric errors in the L_∞ norm for the approximation of $f(x) = (1+\sin(x))/(1-\cos(x)/2)$ with the RBF shape parameter $\alpha = 0.15$.

2.3 Cardinal functions (Lagrange basis) for periodic RBF

The cardinal functions $C_j(X)$ are linear combinations of the basis functions, which satisfy the "cardinal conditions"

$$C_j(x_k) = \begin{cases} 1, k \equiv j \\ 0, k \neq j \end{cases} \tag{2.29}$$

In the cardinal basis, the interpolant of a function is simply

$$f_N(\vec{x}) \equiv \sum_{j=1}^{N} f(\vec{x}_j) C_j(\vec{x}) \tag{2.30}$$

For any interpolating basis, the equivalent cardinal basis always exists so long as the interpolation (Vandermonde) matrix is invertible. Indeed, we can write without

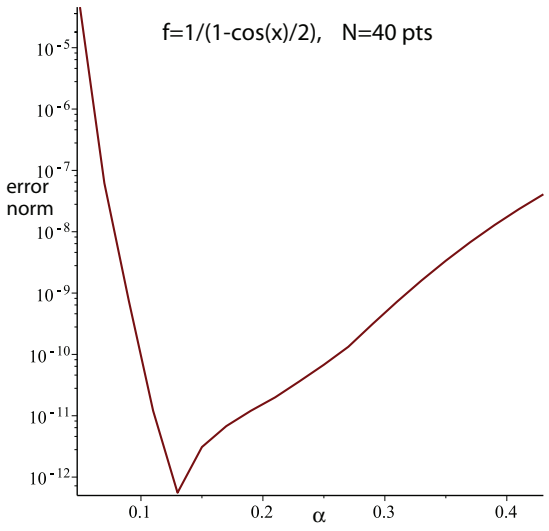

Figure 2.3: Maximum pointwise errors (errors in the L_∞ norm) for Periodized Multiquadrics (PMQ) for $f(x) = 1/(1-\cos(x)/2)$ with forty uniformly spaced interpolation points.

approximation, in any number of dimensions,

$$C_j(x) = \sum_{k=1}^{N} \overline{V}_{kj} \phi(\| \vec{x} - \vec{x}_k \|), \ \vec{x} \in R^d \tag{2.31}$$

where the $\overline{V}_{kj}$ are the elements of the *inverse* of the interpolation matrix whose elements are $V_{jk} = \phi(||\vec{x}_j - \vec{x}_k||)$.

To analyze cardinal functions for periodic RBFs, we need the following.

Definition II.6 (Periodic Grid)**.** Let the N interpolation points x_k lie on the interval $[-\pi, \pi)$. A periodic grid on the entire real axis is then defined as the union of these points plus all other points of the form

$$x_{k+mN} = x_k + 2\pi\, m, \qquad m = \text{any integer} \tag{2.32}$$

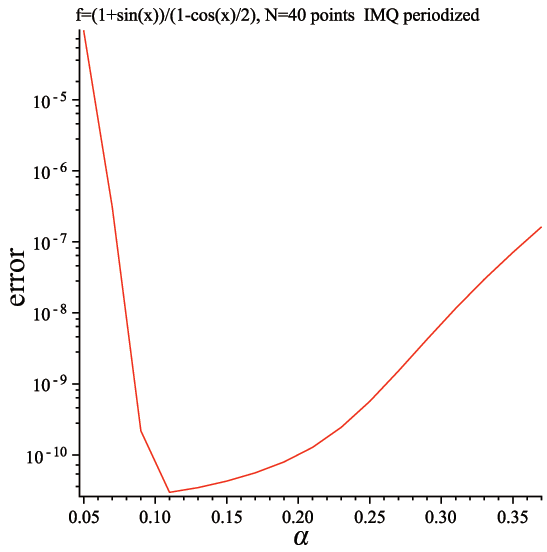

Figure 2.4: Maximum pointwise errors (errors in the L_∞ norm) for Periodized Inverse Multiquadrics (PIMQ) for $f(x) = (1 + \sin(x))/(1 - \cos(x)/2)$ with forty uniformly spaced interpolation points.

The infinite-interval cardinal functions can always be written as RBF series:

$$C_j^\infty(x) = \sum_{k=-\infty}^{\infty} W_{kj}\phi(x - x_k) \tag{2.33}$$

where for simplicity we retreat to one space dimension although the ideas below generalize rather easily to as many dimensions as we please. We impose the usual cardinal conditions over the entire infinite periodic grid:

$$C_j^\infty(x_k) = \begin{cases} 1, k = j \\ 0, k \neq j \end{cases} \qquad j, k = 0, \pm 1, \pm 2, \ldots \tag{2.34}$$

Theorem II.7 (Cardinal Function Imbrication)**.** *Define the N-point periodic RBF*

cardinal functions to be sums of the form

$$C_j(x) = \sum_{n=1}^{N} \overline{W}_{nj}\theta(x - x_n) \tag{2.35}$$

satisfying the cardinal conditions for the N points $x_k \in [-\pi, \pi)$*:*

$$C_j(x_k) = \begin{cases} 1, k = j \\ 0, k \neq j \end{cases} \qquad j, k = 1, 2, \ldots N \tag{2.36}$$

Then the periodic RBF cardinal functions are the imbrication of the infinite grid cardinal functions:

$$C_j(x) = \sum_{m=-\infty}^{\infty} C_j^{\infty}(x - 2\pi m) \tag{2.37}$$

Proof. Define $\tilde{C}_j$ by the first line in Eqn. 2.38. Substitute the series for the infinite grid cardinal function to obtain the second line. Interchanging the order of summation gives the third line. Next, the definition of the periodized basis function allows one to collapse each sum in m to a copy of the kernel θ. The definition of a periodic grid allows us to write each point in the form x_{n+mN} where n is restricted to the range $1, 2, \ldots N$, yielding the fifth line below. Because $x_n - x_{n+mN} = 2\pi m$, the periodicity $\theta(x)$ yields $\theta(x - x_{n+mN}) = \theta(x - x_n)$, generating the sixth line. The final row of the equation comes from the definition which is given immediately after the equations

themselves.

$$\begin{aligned}
\tilde{C}_j(x) &= \sum_{m=-\infty}^{\infty} C_j^{\infty}(x - 2\pi m) \\
&= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} W_{nj}\phi(x - x_n - 2\pi m) \\
&= \sum_{n=-\infty}^{\infty} W_{nj} \sum_{m=-\infty}^{\infty} \phi(x - x_n - 2\pi m) \\
&= \sum_{n=-\infty}^{\infty} W_{nj}\theta(x - x_n) \\
&= \sum_{n=1}^{N} \sum_{m=-\infty}^{\infty} W_{n+mN,j}\theta(x - x_{n+mN}) \\
&= \sum_{n=1}^{N} \sum_{m=-\infty}^{\infty} W_{n+mN,j}\theta(x - x_n) \\
&= \sum_{n=1}^{N} \overline{W}_{nj}\theta(x - x_n)
\end{aligned} \tag{2.38}$$

where we have defined $\overline{W}_{nj} \equiv \sum_{m=-\infty}^{\infty} W_{n+mN,j}$. Thus, $\tilde{C}_j(x)$ is of the correct series form to be a periodic RBF cardinal function; we now must verify that it satisfies the cardinal conditions, too. However, each infinite interval cardinal function is zero at all the points of the infinite periodic grid except for one. When we imbricate the infinite interval cardinal functions, using $h = 2\pi/N$

$$\begin{aligned}
\tilde{C}_j(x_k) &= \sum_{m=-\infty}^{\infty} C_j^{\infty}(x_k - 2\pi m) \\
&= \sum_{m=-\infty}^{\infty} C_j^{\infty}(\frac{2\pi}{N}k - 2\pi m) \\
&= \delta_{j,k} \qquad j = 1, 2, \ldots N \,\&\, k = 1, 2, \ldots N
\end{aligned} \tag{2.39}$$

where the infinite sum becomes redundant when j and k are each restricted to the range $[1, N]$. It follows that the sum defined as $\tilde{C}_j(x) = C_j(x)$. □

When the grid is uniform, all cardinal functions are translates of the cardinal

function peaked at the origin, which we shall denote by $C(x)$ without a subscript:

$$C_j(x) \equiv C(x - x_j) \tag{2.40}$$

We now recall the following theorem

Theorem II.8. *On a uniform, infinite grid with grid spacing h, the cardinal functions for Gaussian, Inverse Quadratic, Sech [Hyperbolic Secant], Multiquadric and Inverse Multiquadric RBFs are given by the common form*

$$C^\infty(x;\alpha,h) \sim \sin(\pi x/h)\,\frac{\aleph}{\sinh\left(\aleph\,\pi x/h\right)} \tag{2.41}$$

where $\aleph(\alpha)$ depends on the shape parameter α, a different function for each RBF species as catalogued in Table 2.1 and justified in (Boyd, 2011).

Proof. Trivial combination of theorems from *Boyd and Wang* (2009) and *Boyd* (2011).

□

Table 2.1: $\aleph$ in $C(X) \sim \aleph\,\sin(\pi X)/\sinh(\aleph\,\pi X)$

Type	ϕ [unit grid]	$\aleph$
Gaussian	$\exp(-\alpha^2X^2)$	α^2/π
sech	$\mathrm{sech}(\alpha X)$	α/π
IQ	$1/(1+\alpha^2X^2)$	$\alpha/2$
MQ	$\sqrt{1+\alpha^2X^2}$	$\{(2/\alpha)(K_0(\pi/\alpha)/K_1(\pi/\alpha)) + (4/\pi)\}^{-1} \approx \alpha/2 - \frac{3}{4\pi}\alpha^2 + \frac{15}{(16\pi^2)}\alpha^3$
IMQ	$1/\sqrt{1+\alpha^2X^2}$	$(\alpha\,K_0(\pi/\alpha)/(2\,K_1(\pi/\alpha)) \approx \alpha/2 - \frac{1}{4\pi}\alpha^2 + \frac{3}{16\pi^2)}\alpha^3$

Theorem II.9 (Approximate Periodized Cardinal Functions)**.**

$$\begin{aligned} C(x;\alpha,h) &= \sum_{m=-\infty}^{\infty} \frac{\aleph(\alpha)\,\sin(\pi[x-2\pi m]/h)}{\{\sinh\left(\aleph(\alpha)\,\pi[x-2\pi m]/h\right)\}} \\ &= \sin(\frac{\pi}{h}x)\aleph(\alpha)\frac{2K(\sqrt{1-k^2})}{\pi}\,cs\left(\frac{2}{\pi}K(\sqrt{1-k^2})\frac{\pi\aleph(\alpha)}{h}x;k\right) \end{aligned} \tag{2.42}$$

for the PGA, PIQ and Periodic Hyperbolic Secant RBFs with the appropriate choice of $\aleph(\alpha)$ *where*

$$\begin{aligned} q &= \exp\left(-\frac{h}{2\,\aleph(\alpha)}\right) = \exp\left(-\frac{\pi}{N\,\aleph(\alpha)}\right) \\ k &= EllipticModulus(q) \equiv \left(\frac{\theta_2(0;q)}{\theta_3(0;q)}\right)^2 \end{aligned} \tag{2.43}$$

where cs is the usual Jacobian elliptic function and $\theta_2(x;q)$ *and* $\theta_3(x;q)$ *are standard Jacobian theta functions.*

All cardinal functions C_j *are translates of* C*:*

$$C_j(x;\alpha,h) \equiv C(x - x_j;\alpha,h) \tag{2.44}$$

Proof. Combination of the two preceding theorems. □

Note that for all spectrally accurate RBF species, $\alpha \to 0$ is equivalent to $\aleph \to 0$. (This is what Fornberg calls the "flat limit" of very wide RBFs.) The pair of hyperbolic tangent functions become a "top hat" function, the cardinal function Fourier series becomes a trigonometric polynomial, and the cardinal function becomes the Dirichlet kernel of Fourier series theory, which is also the usual trigonometric interpolation cardinal function.

2.4 Convergence rates

We shall assume that there are N grid points on the periodicity interval, $x \in [-\pi, \pi]$. We shall not require that the points be uniformly distributed. We apply a periodic grid as defined in the preceding section: $x_{j+kN} \equiv x_j + 2\pi k$ for all integers k.

It will be convenient to use a cardinal function representation. Let $C_n^\infty(x)$ denote the non–periodized infinite interval cardinal functions constructed from the RBF kernel $\phi(x)$. The periodic cardinal functions $C_j(x)$ are the imbrication of $C_n^\infty(x)$ as in

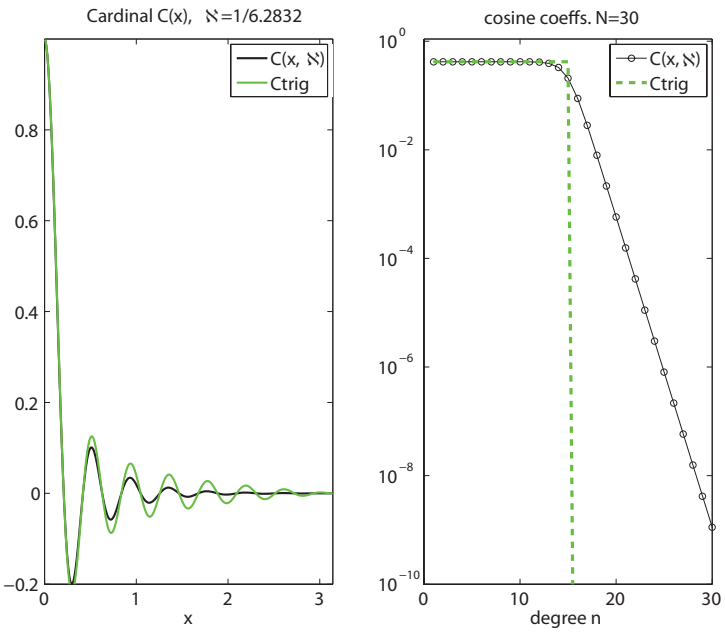

Figure 2.5: Comparison of the RBF and trigonometric cardinal functions. For the Gaussian RBF, $\alpha = 1/\sqrt{2}$ and $\aleph = 1/(2\pi)$ — and 30 interpolation points. Left: $C(x)$ itself; the curve of larger amplitude is the trigonometric cardinal function for the same number of points. Right: the Fourier coefficients of the RBF cardinal function (solid) and the trigonometric cardinal function (dashed).

Theorem 2. Since $C_n^\infty(x_m) = \delta_{nm}$, it follows that

$$C_j(x_{m+kN}) = \begin{cases} 1, j = m, & k = 0, \pm 1, \pm 2, \ldots \\ 0, j \neq m, & k = 0, \pm 1, \pm 2, \ldots \end{cases} \tag{2.45}$$

$$C_{j+kN}(x) = C_j(x) \tag{2.46}$$

The periodic RBF interpolant is

$$\begin{aligned} f(x) \approx f_N(x) &\equiv \sum_{j=1}^{N} f(x_j)\, C_j(x) \\ &= \sum_{j=1}^{N} f(x_j) \left\{ \sum_{m=-\infty}^{\infty} C_j^\infty(x - 2\pi m) \right\} \end{aligned} \tag{2.47}$$

Using the Partition-of-Unity, every periodic function can be rewritten in an "imbricate" form, and we can write without approximation

$$f(x) \equiv \sum_{m=-\infty}^{\infty} g(x - 2\pi m) \tag{2.48}$$

Let us now approximate this parent function on the unbounded interval by the infinite cardinal function series

$$g(x) \approx g^{\infty}(x) \equiv \sum_{n=-\infty}^{\infty} g(x_n) C_n^{\infty}(x) \tag{2.49}$$

Note that this is an infinite series even though we use only N grid points in calculating the periodic RBF interpolant.

The imbrication of g^{∞} is

$$\begin{aligned}
\mathrm{Imb}(g^{\infty}) &= \sum_{m=-\infty}^{\infty} g^{\infty}(x - 2\pi m) \\
&= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} g(x_n) C_n^{\infty}(x - 2\pi m) \\
&= \sum_{n=-\infty}^{\infty} g(x_n) \left\{ \sum_{m=-\infty}^{\infty} C_n^{\infty}(x - 2\pi m) \right\} \\
&= \sum_{n=-\infty}^{\infty} g(x_n) C_n(x) \\
&= \sum_{j=1}^{N} \sum_{k=-\infty}^{\infty} g(x_{j+kN}) C_{j+kN}(x) \\
&= \sum_{j=1}^{N} \sum_{k=-\infty}^{\infty} g(x_{j+kN}) C_j(x) \\
&= \sum_{j=1}^{N} C_j(x) \left\{ \sum_{k=-\infty}^{\infty} g(x_{j+kN}) \right\}
\end{aligned} \tag{2.50}$$

However, the fact that g is the pattern function for f implies that

$$\begin{aligned} f(x_j) &= \sum_{m=-\infty}^{\infty} g(x_j - 2\pi m) \\ &= \sum_{k=-\infty}^{\infty} g(x_{j+kN}) \end{aligned} \tag{2.51}$$

Therefore

$$\begin{aligned} \mathrm{Imb}(g^\infty) &= \sum_{j=1}^{N} C_j(x) \left\{ \sum_{k=-\infty}^{\infty} g(x_{j+kN}) \right\} \\ &= \sum_{j=1}^{N} C_j(x) f(x_j) \\ &= f_N(x) \end{aligned} \tag{2.52}$$

It follows that if g^∞ is a good approximation to $g(x)$,

$$g^\infty \;\; = \;\; g + E^\infty \tag{2.53}$$

where $|E^\infty| << 1$, then imbrication gives

$$\begin{aligned} f_N(x) &= \mathrm{Imb}(g^\infty) \\ &= \mathrm{Imb}(g) + \mathrm{Imb}(E^\infty) \\ &= f(x) + \mathrm{Imb}(E^\infty) \end{aligned} \tag{2.54}$$

Thus, the accuracy of RBF approximation an infinite interval implies the periodic interpolant must be equally accurate. The formal statement is the following.

Theorem II.10 (Equiconvergence Theorem)**.** *Suppose that $f(x)$ is periodic with period 2π and is analytic in a strip of width w about the real axis, $|\Im(x)| \leq w$, $\Re(x) =$ anything. Suppose we can construct a pattern function $g(x)$ which is analytic in the same strip and decays exponentially fast within the strip as $|\Re(x)| \to \infty$, such*

as $g(x) \equiv \mathfrak{P}(x; L)\, f(x)$.

Then if such a $g(x)$ *can be approximated by RBFs with the kernel* $\phi(x)$ *so as to achieve a given rate of convergence, then the approximation of* $f(x)$ *on the periodicity interval by interpolation using a basis* $\theta(x - x_j)$ *which is the periodization of* ϕ *will converge at the same rate. More precisely, denoting the RBF approximations by* g^∞ *and* $f_N(x)$

$$g^\infty \;=\; g + E^\infty \tag{2.55}$$

then

$$f_N(x) \;=\; f(x) + Imb(E^\infty) \tag{2.56}$$

In particular, if the convergence rate on the infinite interval is geometric, the periodic RBF interpolant will also converge at a geometric rate.

We now need to prove the following:

Theorem II.11 (Geometric Convergence on the Infinite Interval)**.** *Let* N *denote the number of points on* $x \in [-\infty, \infty]$ *and generate an infinite grid by periodically replicating this* N*-point grid over the entire real axis. Suppose* $g(x)$ *is analytic in a strip of width* w *about the real axis,* $|\Im(x)| \leq w$, $\Re(x) =$ *anything and decays exponentially fast within the strip as* $|\Re(x)| \to \infty$, *such as* $g(x) \equiv \mathfrak{P}(x; L)\, f(x)$. *Then the error in*

$$g(x) \approx g^\infty(x) \equiv \sum_{n=-\infty}^{\infty} g(x_n) C_n^\infty(x) \tag{2.57}$$

is bounded by $\exp(-N\mu)$ *for some positive constant* μ.

Proof. Platte and Driscoll (*Platte and Driscoll*, 2005) and Platte (*Platte*, 2011) have established a geometric rate of convergence for RBF interpolation on a *finite* interval

for any of the five RBF species considered here. Let us suppose that we approximate $g(x)$ on the truncated, finite domain $x \in [-M\pi, M\pi]$ using $P = MN$ interpolation points. For fixed M, the Platte-Driscoll theory proves that the RBF approximation error decreases geometrically fast. Because by assumption $g(x)$ is decaying exponentially fast with $|x|$, the error in truncating the domain decreases exponentially fast with M. For fixed M and increasing N, the approximation error in the finite interval will eventually decrease below the domain truncation error, $E_D \sim O(|g(M\pi)|)$; the overall maximum pointwise error can decrease further only if the size of the finite domain is increased. If $g(x)$ is decreasing like a Gaussian on $x \in [-\infty, \infty]$, then minimizing the sum of the truncation error and of the approximation error on the finite, truncated interval requires $M = \sqrt{N}$ and thus $P = N^{3/2}$. The error decreases as $O(\exp(-\mu' P^{2/3}))$ for some constant μ'. This is only a *subgeometric* rate of convergence as measured by the usual computational standard of the total number of interpolation points P.

However, our computational problem involves only the N points on the periodicity interval; no computer will ever be tasked with evaluating the infinite sum that defines g^∞. Since $P = N^{3/2}$, $O(\exp(-\mu' P^{2/3})) \sim O(\exp(-\mu' N))$. We have thus demonstrated that the rate of convergence is geometric in N.

We have sloughed over one issue, which is that the asymptotic rate of convergence μ' depends upon the distance d to the expansion interval of the singularity of $f(x)$ which is closest to the expansion interval. When we make a change of coordinate, $y = x/(M\pi)$, to transform a very large approximation interval in x to the canonical interval for RBF approximation on a finite domain, $y \in [-1, 1]$, we find that in the coordinate y, the distance from this singularity to the interval is $d' = d/M$. For a fixed number of interpolation points P in y, this would seem to imply slower and slower rate of convergence as M increases. However, the logarithm of the error is also

proportional to P, and P increases with increasing M. The error bound is

$$\exp(-Pcd') = \exp(-MNc(d/M)) = \exp(-cdN) \tag{2.58}$$

where $c > 0$ is a constant independent of M, N and d whose numerical value is irrelevant here, though given by the Platte-Driscoll theory. The error does indeed fall geometrically with N. □

The combination of the two previous theorems then trivially proves the following.

Theorem II.12 (Geometric Rate-of-Convergence Theorem)**.** *Suppose that $f(x)$ is periodic with period 2π and is analytic in a strip of width w about the real axis, $|\Im(x)| \leq w$, $\Re(x) =$ anything. Let N denote the number of points on $x \in [-\pi, \pi]$. Then the periodic RBF approximation error is geometrically converging, that is,*

$$E(x) \equiv |f(x) - f_N(x)| \leq \exp(-N\mu) \tag{2.59}$$

Figure 2.6 shows a typical instance of interpolation using Periodic Gaussian RBFs which provides experimental confirmation of our analysis.

2.5 Symmetry-respecting RBFs

Many problems have parity symmetries; the eigenfunctions of the Mathieu equation, discussed below, have definite parity with respect to both $x = 0$ and $x = \pi/2$ and thus divide into four classes. Since it is usually much cheaper to solve several small problems than one big problem, it is highly desirable to build basis functions that explicitly incorporate such symmetries as explained in *Boyd* (1999).

For parity about the origin, it is convenient to choose the grid points to lie on $x \in (0, \pi]$. Then

$$\theta^S(x) \equiv \theta(x - x_j) + \theta(x + x_j) \tag{2.60}$$

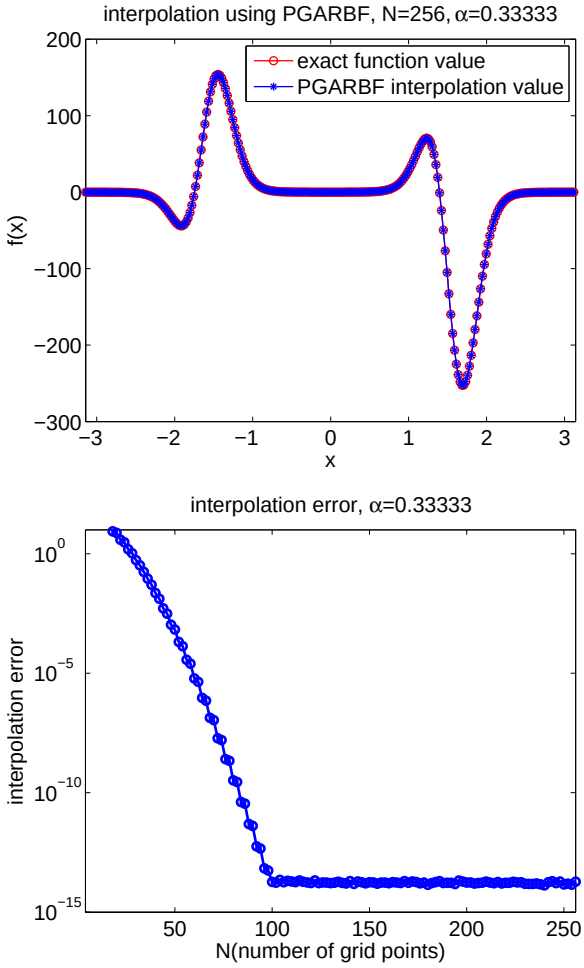

Figure 2.6: Up: Exact values and values interpolated using PGA of function $f(x) = (\sin(3x) + 5\cos(x))\exp(-5\cos(2x) + 0.25\sin(x))$, over the domain $x \in [-\pi, \pi]$. Down: Maximum pointwise errors (L_∞ error norms) for the interpolation of $f(x) = (\sin(3x) + 5\cos(x))\exp(-5\cos(2x) + 0.25\sin(x))$ using Periodic Gaussian RBFs as a function of the number of grid points N.

satisfies the property of parity with respect to the origin,

$$\theta^S(x) = \theta^S(-x) \qquad \forall\, x \tag{2.61}$$

because of the property of Strictly Positive Definite periodic RBF, proved by earlier workers, that $\theta(-x) = \theta(x)$. Similarly,

$$\begin{aligned} \theta^A(x) &\equiv \theta(x - x_j) - \theta(x + x_j) \\ &= \theta(x - x_j) - \theta(-x - x_j) \end{aligned} \tag{2.62}$$

is antisymmetric, that is,

$$\theta^A(x) = -\theta^A(-x) \qquad \forall\, x \tag{2.63}$$

Analogously, basis functions which have definite parity with respect to *both* $x = 0$ and $x = \pi/2$,

$$\theta^{SS}(x; x_j) = \theta(x - x_j) + \theta(-x - x_j) + \theta(\pi - x - x_j) + \theta(\pi + x - x_j) \tag{2.64}$$

$$\theta^{SA}(x; x_j) = \theta(x - x_j) + \theta(-x - x_j) - \theta(\pi - x - x_j) - \theta(\pi + x - x_j) \tag{2.65}$$

$$\theta^{AS}(x; x_j) = \theta(x - x_j) - \theta(-x - x_j) + \theta(\pi - x - x_j) - \theta(\pi + x - x_j) \tag{2.66}$$

$$\theta^{AA}(x; x_j) = \theta(x - x_j) - \theta(-x - x_j) - \theta(\pi - x - x_j) + \theta(\pi + x - x_j) \tag{2.67}$$

These four species are schematically illustrated in Fig. 2.7.

Some care is necessary in choosing good points and basis functions to match. For example, let us suppose that N is odd. The grid spacing is space $h = 2\pi/N$. The origin will not be a grid point for any odd N. We therefore choose the grid

$$x_j = -\pi + jh, j = 1, 2, \ldots N \tag{2.68}$$

so that $x = \pi$ is always a grid point. Since an odd number cannot be divided in two, when we split the grid points and basis functions according to parity, we must employ $(N+1)/2$ basis functions symmetric about the origin but one fewer basis function which is antisymmetric with respect to $x = 0$. The grid used in computations for symmetric functions is the set of all positive grid points including $x = \pi$ while antisymmetric calculations include all positive points with the exception of $x = \pi$.

When the functions also have parity with respect to $x = \pi/2$, the four-fold splitting described above can be implemented. However, when $(N-1)/2$ is *odd*, the functions which are antisymmetric with respect to the origin cannot be split into two and matched with the grid points on the interval $x \in [\pi/2, \pi)$, excluding π where all functions antisymmetric about the origin are zero. We therefore must restrict N for four-fold symmetry to *odd* integers such that $(N-1)/2$ is *even*; the allowed values are therefore restricted to $N = \{5, 9, 13, 17, \ldots\}$.

2.6 Numerical Example: Mathieu functions

In this section, we will solve an eigenvalue problem using periodic Gaussian and Inverse Quadratic RBF basis functions. However, the procedure can be generalized to almost any ODE or PDE with periodic coefficients on a periodic domain.

2.6.1 Mathieu equation: Symmetry, asymptotics and resolution

The Mathieu functions have been intensively investigated since their introduction in 1868. (*Olver et al.*, 2010; *McLachlan*, 1947; *Morse and Feshbach*, 1953). Mathieu functions are the eigenfunctions in the angular elliptical coordinate when the Helmholtz equation is solved in an elliptical domain. As such, they arise in many applications as catalogued in McLachlan's book (*McLachlan*, 1947) and in Chapter 28 of the NIST Digital Library (*Olver et al.*, 2010). Although they arise through the method of separation of variables for one particular partial differential equation, they

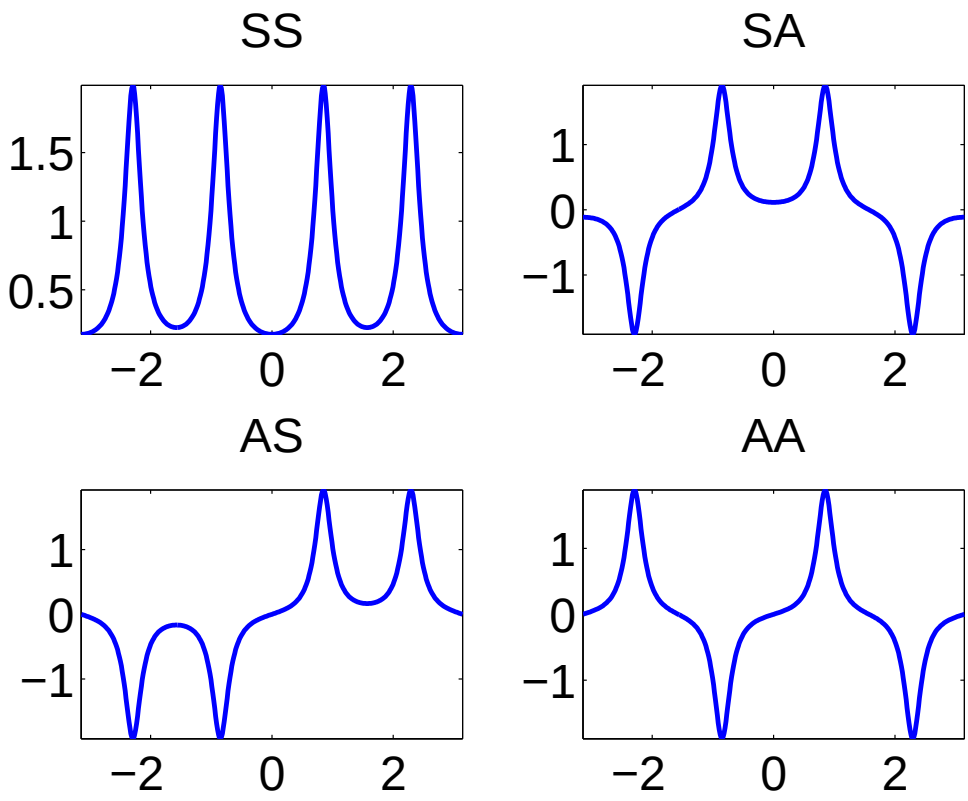

Figure 2.7: Symmetrized RBF basis functions. The first letter of each two-symbol code is "S" if the basis function is symmetric with respect to reflection about $x = 0$ and "A" if the functions changes sign upon reflection; the second letter has the same meaning except for reflection about $x = \pi/2$, which is marked by the vertical dashed line in each plot.

are a good spectral basis for a wide range of PDEs in elliptical coordinates (*Shen and Wang*, 2009; *Wang and Zhang*, 2011).

The Mathieu eigenfunction solves

$$u_{xx} - 2q\cos(2x)u(x) = -\lambda u(x) \tag{2.69}$$

where q is a real parameter on the range $q \in [0, \infty]$ and λ is the eigenvalue. For all q, the eigenfunctions fall into four distinct symmetry classes which are catalogued in Table 2.2.

Table 2.2: Symmetry Classes of Mathieu Eigenfunctions

Symbol	Parity: $x = 0$	Parity: $x = \pi/2$	Period	Fourier basis
$\mathrm{ce}_{2n}(x;q), n = 0, 1, \ldots$	Even	Even	π	$\cos(2mx), m = 0, 1, 2$
$\mathrm{ce}_{2n+1}(x;q), n = 0, 1, \ldots$	Even	Odd	2π	$\cos([2m-1]x), m = 0, 1, 2$
$\mathrm{se}_{2n}(x;q), n = 1, \ldots$	Odd	Odd	π	$\sin(2mx), m = 1, 2$
$\mathrm{se}_{2n+1}(x;q), n = 0, 1, \ldots$	Odd	Even	2π	$\sin([2m-1]x), m = 1, 2$

In the limit $q = 0$,

$$\mathrm{ce}_{2n}(x;0) = \cos(2nx), \qquad \lambda = -4n^2, \qquad n = 0, 1, \ldots \tag{2.70}$$

$$\mathrm{ce}_{2n+1}(x;0) = \cos([2n+1]x), \qquad \lambda = -(2n+1)^2, \qquad n = 0, 1, \ldots \tag{2.71}$$

$$\mathrm{se}_{2n}(x;0) = \sin(2nx), \qquad \lambda = -4n^2, \qquad n = 1, \ldots \tag{2.72}$$

$$\mathrm{se}_{2n+1}(x;0) = \sin([2n+1]x), \qquad \lambda = -(2n+1)^2, \qquad n = 1, \ldots \tag{2.73}$$

For small q, the eigenfunctions are well approximated by trigonometric functions (*Olver et al.*, 2010). In the large q limit, the eigenfunctions are approximated by asymptotic expansions in which each order is a finite sum of Hermite functions. The leading order for the doubly-symmetric modes is

$$\mathrm{ce}_m(x;q) \sim \text{const } \exp(-\sqrt{q}\sin^2(x+\pi/2))H_m(\sqrt{2}q^{1/4}\sin(x+\pi/2)), q \gg 1 \tag{2.74}$$

The Hermite form is most easily derived from an equivalent form of the Mathieu equation. Define

$$x = y + \pi/2, \qquad \nu \equiv \lambda + 2q \tag{2.75}$$

then the Mathieu equation is

$$u_{yy} - 4q\sin^2(y)u(y) = -\nu u(y) \tag{2.76}$$

where we have used the identities $\cos(2y+\pi) = -\cos(2y)$ and $1-\cos(2y) = 2\sin^2(y)$. For small y, power series expansion of the sine function gives the parabolic cylinder

equation, $u_{yy} - 4qy^2u(y) = -\nu u(y)$,whose eigenfunctions on an unbounded domain are the Hermite functions. Of course, the domain is periodic rather than infinite, but because the Hermite functions decay exponentially as $|y|$ increases, the Hermite approximation is not bad when the mode number is small and the parameter q is large.

The Hermite asymptotics imply that the peaks of the eigenfunctions become narrower and narrower as q increases. Define $N_\delta(\delta)$ to be such that all Fourier coefficients $a_n \leq \delta$ for $n \geq N_\delta(\delta)$ when the eigenfunctions are normalized so that the largest Fourier coefficient is one. If we arbitrarily pick $\delta = 2.2 \times 10^{-16}$, then the Hermite asymptotic approximation suggests

$$N_\delta(\delta) \sim const.\sqrt[4]{q}, \qquad q >> 1 \tag{2.77}$$

Numerical experimentation has shown that the constant is about 6 for the ground state and also that the asymptotic approximation is superb for large q. We must expect a similar scaling for other numerical methods *without adaptation*.

Table 2.3: Fourier resolution required to achieve machine precision for the lowest eigenmode for different Mathieu parameters q; N_δ is the number of basis functions of the same double parity class as the eigenmode

q	N_δ	$6\sqrt[4]{q}$	relative error [when $N = N_\delta$]
10	14	10.7	.238
100	21	19.0	0.096
1000	36	33.7	0.063
10,000	61	60.	0.016
100,000	108	106.70	0.012
1,000,000	191	189.74	0.0066
10,000,000	338	337.40	0.0018
100,000,000	601	600	0.0017
1×10^9	1068	1067	0.00097

2.6.2 RBF discretization of Mathieu equation

The discretization of the second order derivative can be obtained by differentiating the corresponding periodic RBF basis functions $\theta_j(x)$ twice, with respect to x. For periodic Gaussian RBFs,

$$\begin{aligned}\theta_{x,j}(x) = \frac{d\theta_j(x)}{dx} &= \frac{d}{dx}\sum_{m=-N}^{N}\exp(-\epsilon^2(x-x_j-2m\pi)^2) \\ &= \sum_{m=-N}^{N}\{-2\epsilon^2(x-x_j-2m\pi)\exp(-\epsilon^2(x-x_j-2m\pi)^2)\}\end{aligned} \tag{2.78}$$

$$\begin{aligned}\theta_{xx,j}(x) = \frac{d^2\theta_j(x)}{dx^2} &= \frac{d^2}{dx^2}\sum_{m=-N}^{N}\exp(-\epsilon^2(x-x_j-2m\pi)^2) \\ &= \sum_{m=-N}^{N}\{[-2\epsilon^2+4\epsilon^4(x-x_j-2m\pi)^2]\exp(-\epsilon^2(x-x_j-2m\pi)^2)\}\end{aligned} \tag{2.79}$$

For periodic Inverse Quadratic RBFs,

$$\theta_x(x) \equiv \frac{d\theta_j(x)}{dx} = -\frac{e^{-1/\epsilon}(1-e^{-2/\epsilon})}{\epsilon}\frac{\sin(x)}{[1+e^{-2/\epsilon}-2e^{-1/\epsilon}\cos(x)]^2} \tag{2.80}$$

$$\theta_{xx}(x) \equiv \frac{d^2\theta_j(x)}{dx^2} = \frac{e^{-1/\epsilon}(1-e^{-2/\epsilon})}{\epsilon[1+e^{-2/\epsilon}-2e^{-1/\epsilon}\cos(x)]^2}\left[\frac{4e^{-1/\epsilon}\sin(x)^2}{1+e^{-2/\epsilon}-2e^{-1/\epsilon}\cos(x)}-\cos(x)\right] \tag{2.81}$$

A function $f(x)$ with period 2π can be expanded as,

$$f(x) = \sum_{j=1}^{N} a_j\theta(x-x_j) \tag{2.82}$$

which is substituted into the Mathieu equation yielding

$$\sum_{j=1}^{N}[\theta_{xx}(x-x_j)-2q\cos(2x)\theta(x)]a_j = \lambda\sum_{j=1}^{N}\theta(x-x_j)a_j \tag{2.83}$$

We discretize the domain $[-\pi, \pi]$ into N intervals where N is also the number of grid points. (Note that since a periodic interval is topologically a circle and therefore $x = \pi$ and $x = -\pi$ are *identical*, one of the two points must be omitted to avoid redundancy and a singular interpolation matrix; our convention is to delete $x = \pi$.) The grid is

$$x_j = -\pi + j\frac{2\pi}{N}, j = 0, ..., N-1 \tag{2.84}$$

(To exploit symmetry, we then take a subset of this by keeping only those $x_j \in [0, \pi/2]$.) Then, the above equations can be rewritten as the generalized matrix eigenvalue problem

$$\vec{\vec{A}}\vec{a} = \lambda\vec{\vec{B}}\vec{a} \tag{2.85}$$

where $\vec{\vec{A}}$ is the discretization matrix on the left hand side(LHS) of Mathieu equation with the elements,

$$\vec{\vec{A}}_{i,j} = \theta_{xx}(x_i - x_j) - 2q\,\cos(2x_i)\theta(x_i - x_j)$$

and $\vec{\vec{B}}$ is the discretization matrix on the right hand side (RHS) of the Mathieu equation whose matrix elements are

$$\vec{\vec{B}}_{i,j} = \theta_j(x_i) \tag{2.86}$$

2.6.3 Spurious eigenvalues from periodized RBFs

Table 2.4 shows the eigenvalues of the Mathieu equation with $q = 0$ as computed using thirty periodized Gaussian RBFs. As explained in Chapter 7 of (*Boyd*, 1999), many of the eigenvalues of the matrix eigenvalue problem which falls out of the discretization are poor approximations to those of the differential equation. For smooth differential equations, usually only the *largest* eigenvalues are inaccurate. The table shows that here, alas, spurious eigenvalues are interspersed with accurate eigenvalues

even for small mode number. In his book (*Boyd*, 1999), Boyd advocates checking eigenvalue calculations by running each case twice with a different number of degrees of freedom N, and accepting only those eigenvalues which differ little from one resolution N to another. The table clearly demonstrates the necessity of this caution.

When the Maple parameter "Digits" was increased to 32, imposing a floating point precision of 32 decimal digits, the spurious eigenvalue disappeared and all the matrix eigenvalues were very close approximations to nonpositive integers.

When Digits is reduced to 10, but α increased to 0.5, the incorrect eigenvalues also disappear. (Note that so large an α causes the error to saturate at relatively large values, and is not recommended).

The cause of the spurious small eigenvalues is thus almost certainly the notorious ill-conditioning of RBFs when the shape parameter α is too small relative to the precision of floating point arithmetic.

2.6.4 Adaptive grids

The Hermite function asymptotic approximation to the Mathieu functions shows that the spatial scale of the oscillations becomes smaller and smaller as the Mathieu parameter $q \to \infty$, proportional to $q^{1/4}$. For $q = 10^8$, the Fourier Galerkin method needs about 600 basis functions (even cosines) to resolve the ground state (ce_0) eigenvalue to sixteen decimal places. But the ground state and the other eigenmodes have utterly negligible amplitude outside narrow zones at $x = \pm\pi/2$ as illustrated in Figs. 2.8 and 2.9. The uniform Fourier grid is extremely wasteful. For large q, an adaptive grid is much more efficient.

We shall illustrate adaptive grids using the shifted coordinate, $y = x - \pi/2$, the modified eigenvalue $\nu \equiv \lambda + 2q$ and the transformed equation $u_{yy} - 4q\sin^2(y)u(y) + \nu u(y) = 0$.

Table 2.4: Eigenvalues of symmetric modes of Mathieu equation with $N = 30$

Mode	exact λ	Digits=10/α=0.25	Digits=32/α=0.25	Digits=10/α=0.5
Spurious	-	4.05958759734745822	.273800268075611e-25	.206934972636144e-9
Spurious	-	-3.11473959534470124	-	-
Spurious	-	-36.2597865482689770	-	-
Spurious	-	-52.554190 + i 14.8624	-	-
Spurious	-	-52.554190 - i 14.8624	-	-
Spurious	-	-220.042978041256106	-	-
Spurious	-	404.027044291163179	-	-
1	0	6.422561716e-10	0	2.290259500e-9
2	1	1.000000000185287	1.000000000000000	0.999999999764141
3	1	0.999999999838865	1.000000000000000	1.000000000185500
4	4	3.999999999151707	4.000000000000000	3.999999999523566
5	4	3.999999998710813	4.000000000000000	3.999999999391799
6	9	8.999999999052327	9.000000000000000	8.999999995948817
7	9	8.999999993342662	9.000000000000000	8.999999999320595
8	16	15.999999979102717	16.000000000000000	16.000000001352959
9	16	15.999999982029369	16.000000000000000	15.999999998751971
10	25	25.000000180935661	25.000000000000000	25.000000003577390
11	25	25.000000065647399	25.000000000000000	25.000000005130897
12	36	36.000001917099461	36.000000000000000	36.000000032679722
13	36	36.000000235394317	36.000000000000000	36.000000033153320
14	49	49.000001901481056	49.000000000000000	49.000000346489074
15	49	48.999995707317360	49.000000000000000	49.000000348987243
16	64	63.999809832303974	64.000000000000000	64.000004158663217
17	64	63.999251017453886	64.000000000000000	64.000004181804272
18	81	80.993470252996772	81.000000000000000	81.000049834199885
19	81	80.965140591885785	81.000000000000000	81.000049791460626
20	100	99.962221607817071	100.000000000000000	100.00057785704419
21	100	100.096910918191327	100.000000000000000	100.00057781743992
22	121	122.499446837142017	121.000000000000000	121.00642793070180
23	121	129.247850830095358	121.000000000000000	121.00642823915618
24	144	-	144.000000000003461	144.06699569113613
25	144	-	144.066996808346971	144.00000000000345
26	169	-	169.000000086107382	169.61788113787765
27	169	-	169.000000086107954	169.61787820829832
28	196	-	196.001607202398301	200.02683005486756
29	196	-	196.001607202329676	200.02682533724413
30	225	-	224.999999997070203	225.00001175290069

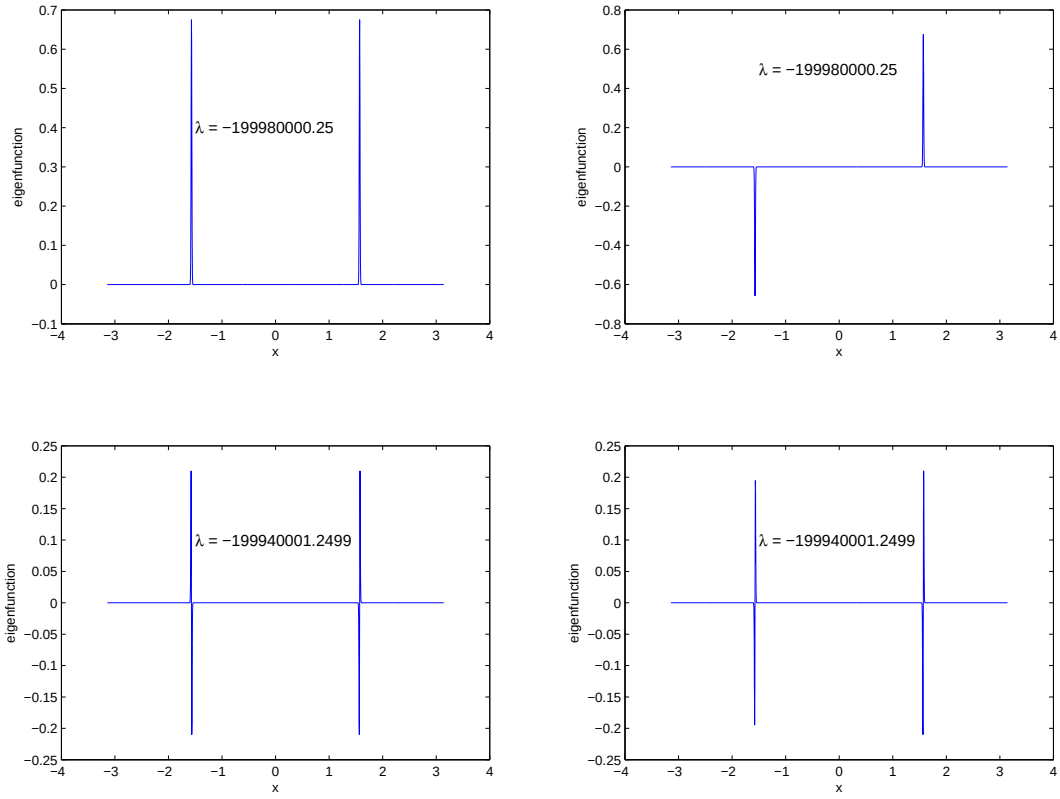

Figure 2.8: The lowest four eigenmodes of the Mathieu equation for $q = 10^8$. The functions are computed using 2000 PIQ basis functions on a uniform grid with $\alpha = 0.20$.

2.6.5 Truncated uniform grid

RBFs are much more tolerant of irregular grids than are other spectral basis sets. The simplest nonstandard grid is a uniform grid that is *truncated* so that the grid points fill the interval $y \in [0, L]$ instead of $y \in [0, \pi/2]$(*Huang et al.*, 2014). To choose L, recall that the Hermite functions $\psi_n(y)$, which approximate the Mathieu functions for large q as noted earlier, change from oscillation for small y to exponential decay at large $|y|$ at the"turning point" [a zero of its second derivative] at (*Boyd*, 1999)

$$y_t = S(q)\sqrt{2n_H + 1} \tag{2.87}$$

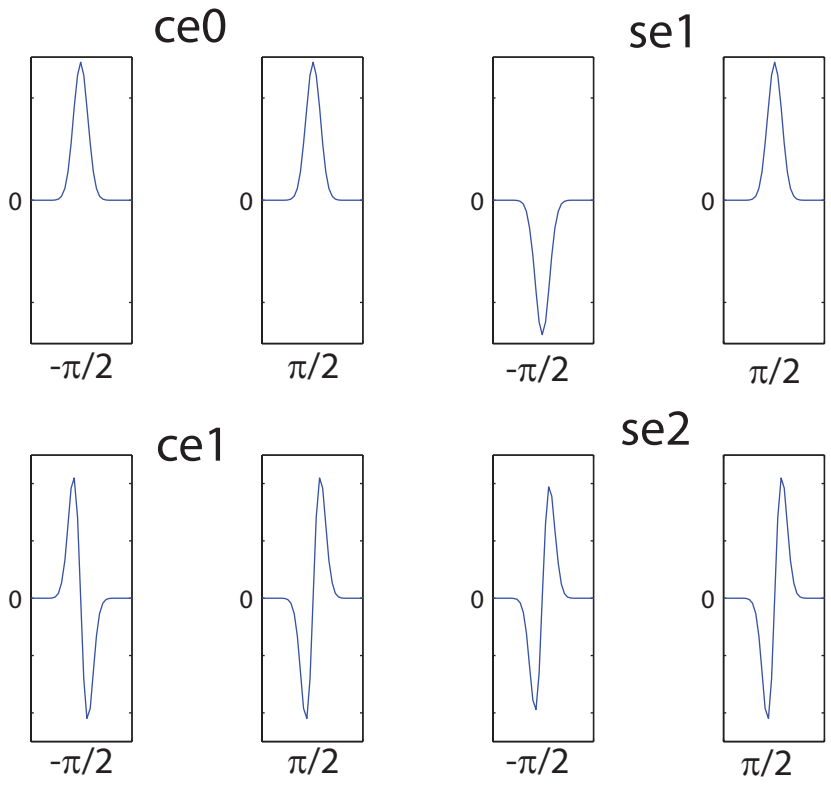

Figure 2.9: Same as previous graph, but zoomed in around $x = \pm\pi/2$. The lowest four eigenmodes of the Mathieu equation for $q = 10^8$. The functions are computed using 2000 PIQ basis functions on a uniform grid with $\alpha = 0.20$.

where n_H is the degree of the Hermite function and

$$S(q) = \frac{1}{\sqrt{2}\, q^{1/4}} \tag{2.88}$$

It follows that we should choose L, the size of our truncated grid, to be a multiple of $S(q)$.

Fig. 2.10 shows a typical experiment. The leftmost points are for $L = S(q)$, which is clearly too small. For the lowest four modes, roughly $L = 6S(q)$ is optimum.

In higher precision arithmetic, Periodic Gaussian RBFs achieve relative errors smaller than 10^{-16} with $L = 6S(q)$, $\alpha = 1/5$ and just twenty interpolation points — a vast improvement over the 600 interpolation points required for the Fourier basis!

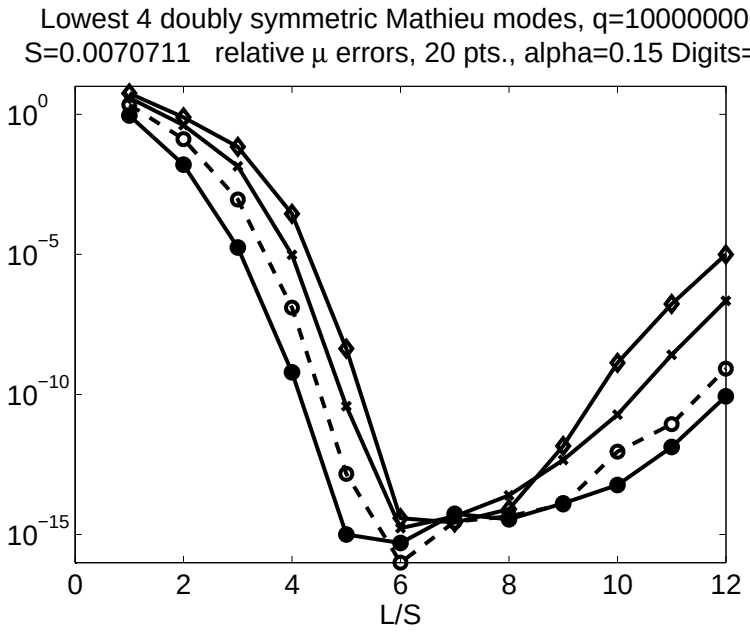

Figure 2.10: Truncated Uniform Grid: Relative errors, $|\nu_{RBF} - \nu|/\nu$, for the lowest four doubly symmetric Mathieu modes, ce_0 [bottom curve], ce_2, ce_4 and ce_6 [top curve], plotted versus the grid parameter L. All computations were done in 16 decimal digit precision using Gaussian RBFs with $\alpha = 0.15$ and 20 interpolation points on the interval $y \in [0, L]$. For $q = 10^8$, the scale factor in the Hermite asymptotic approximations is $S(10^8) = 0.0071$, so $L = 0.042$, the minima of the curves, is about $6S(q)$.

The turning points, $y = \pm S(q)\sqrt{2n_H + 1}$, move further and further from the origin as the ode number n_H increases. The eigenmode oscillates over a wider and wider interval (bounded by the turning points) with increasing n_H. It follows that one needs a *wider* and *wider* truncated grid as the *mode number* increases. Fig. 2.11 confirms this expectation: The optimum grid width is $L = 11S(q)$, nearly double what is optimum for the lowest mode.

The practical advice is to optimize L for the highest mode which is of interest. Although $L = 11S(q)$ is not the best for the ground state (lowest mode), the need to use a relatively large number of interpolation points to resolve the higher mode means that for $\tilde{N} = 50$ points, for example, the eigenvalue for $\mathrm{ce}_0(x)$ is resolved with the relative error of less than 10^{-30} for $\alpha = 0.16$, far better than the relative error in ν_{38}, 1.0×10^{-15} in 16-digit arithmetic.

The curves show that the error becomes smaller and smaller as α decreases, but it is well-known that RBFs are exponentially ill-conditioned as $\alpha \to 0$ unless one makes a very complicated and expensive change of basis (*Fornberg et al.*, 2011). The smallest α that is accurate in sixteen-decimal-digit-precision is $\alpha = 0.15$, but this and 50 points suffice to yield an error ce_{38} of about 10^{-12}.

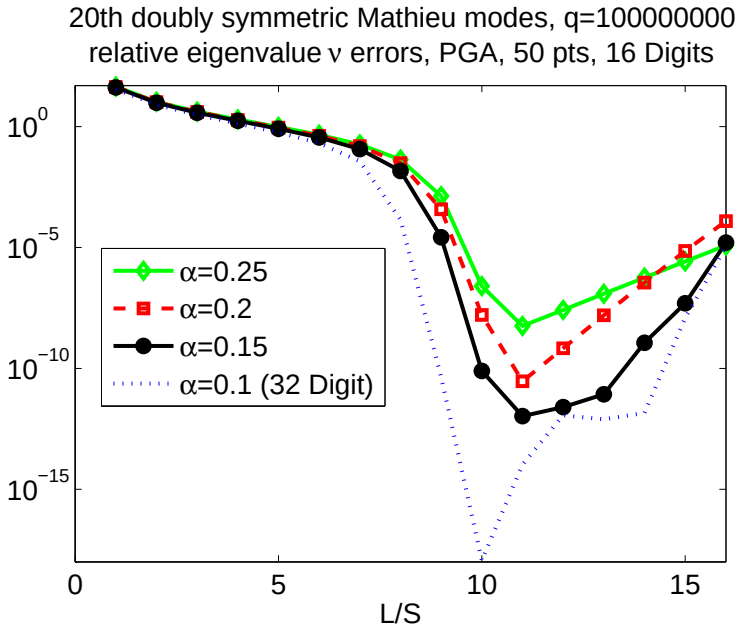

Figure 2.11: Truncated Uniform Grid: Relative errors, $|\nu_{RBF} - \nu|/\nu$, for the twentieth (lowest) doubly symmetric Mathieu modes, $ce_{38}(y, q = 10^8)$ plotted versus the grid parameter L for four different values of the RBF inverse width [shape] parameter α. Periodic, doubly-symmetrized Gaussian RBFs with 50 interpolation points on the interval $y \in [0, L]$. Computations were done in 16 decimal digit precision. This precision failed miserably for $\alpha = 1/10$, so the curve is dotted and without markers to indicate that these small errors are attainable only with higher precision arithmetic. The minimum error is at $L = 11S(q)$.

2.6.6 Arctan/Tan periodic mapping

The truncated uniform grid is simple, but has no grid points at all over most of the interval. This is likely to be a drawback for moderate q.

Boyd introduced (*Boyd*, 1992) a periodic change of coordinate such that a uniform

grid in the computational coordinate ξ is mapped into a very non-uniform grid in the physical coordinate y:

$$y = \arctan(L\tan(\xi)); \tag{2.89}$$

where L is a user-choosable constant, the "map parameter". For $L < 1$, the high resolution regions are centered around $y = 0, \pm\pi, \pm 2\pi, \ldots$. The density of the grid in y is proportional to the first derivative of the mapping function,

$$\frac{dy}{d\xi} = \frac{2L}{1 + L^2 + (1 - L^2)\cos(2\xi)} \tag{2.90}$$

One option is to apply a Fourier spectral method on the uniform grid in ξ. Unfortunately, the transformed differential equation is much more complicated than the original differential equation in y because of the "metric factors" generated by the transformation. In one dimension, this is not a great burden, but a change of coordinates is messy in higher dimensions.

It is far simpler to solve the problem in the original coordinate y, and employ the mapping only to generate the grid. Fourier methods fail on highly nonuniform grids. In contrast, radial basis functions are very tolerant of nonuniform grids and are triumphant here.

There is one technical complication. Spectrally accurate RBFs contain a "shape parameter" ϵ which specifies the inverse width of the basis function. On a uniform grid, it is convenient to specify the absolute shape parameter as $\epsilon = \alpha/h$ where α is the relative shape parameter and h is the grid spacing. When $\alpha \gg 1/4$, RBFs are very inaccurate because each basis function is narrow compared to the interval between the grid points and the RBFs have little overlap. A smooth function is therefore approximated by sum of delta function-like spikes. When $\alpha \ll 1/4$, the interpolation matrix is extremely ill conditioned. On a uniform grid, standard practice is to choose α to be of moderate value in the range $\alpha \approx 1/10 - 1/2$.

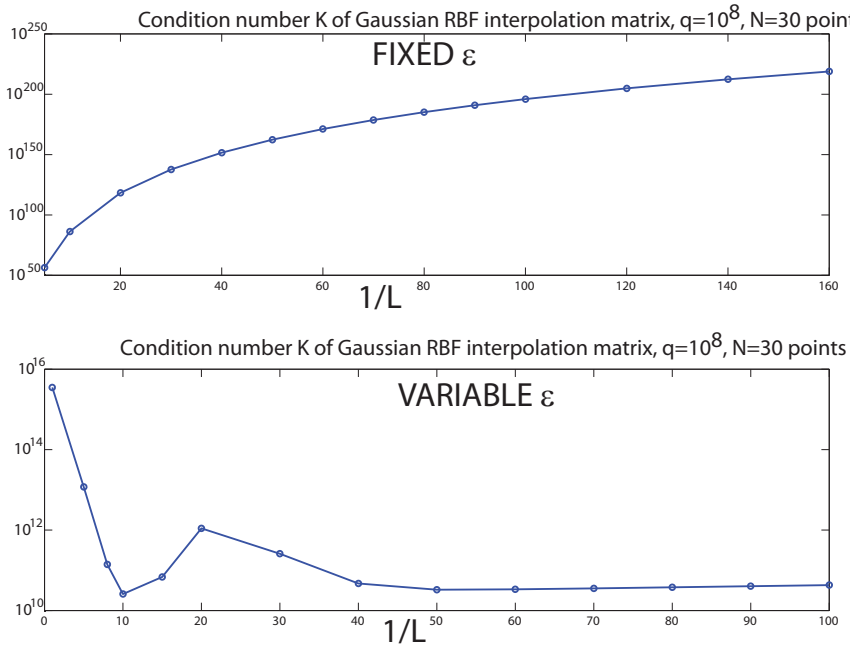

Figure 2.12: Arctan/Tan Mapping: Condition number of the Gaussian RBF interpolation matrix. The upper panel shows the condition number (on a logarithmic scale) when the shape parameter is the same for all basis functions $\epsilon = \alpha/h_{av}$ where $h_{av} = \pi/(2N), \alpha = 1/3$ is the average grid spacing. The lower panel shows the condition number for variable ϵ where the formula is given in the text.

Figure 2.12 shows that when we apply this prescription by replacing h by its average value, the condition number of the interpolation matrix grows exponentially as the map parameter L is decreased. A far better procedure, already widely used, is to choose the absolute width parameter ϵ to be *different* for each basis function where here we chose

$$\epsilon_j = \frac{\alpha}{h_{av}} / \frac{dy}{d\xi}(\xi_j) \tag{2.91}$$

where the average grid spacing is

$$h_{av} \equiv \frac{\pi}{2N} \tag{2.92}$$

Thus, basis functions centered at points in the high resolution region are narrow while RBFs centered where the grid points are far apart are very wide. The lower panel of the figure shows that with this variable-width basis, the condition number stabilizes at a value which is considerably smaller than the reciprocal of machine precision in Matlab (or IEEE double precision), allowing accurate calculations without the use of multiple precision arithmetic. (An $O(1)$ condition number is never possible with a standard RBF basis.)

Figs. 2.13 and 2.14 show that the arctan/tan mapping is much more effective than a uniform grid. More than 600 points are needed to barely resolve $\mathrm{ce}_0(x)$ for $q = 10^8$, but with the mapping, thirty points yield near machine precision accuracy. For higher modes (Fig 2.14), a few more points are needed, but the mapping is still vastly superior to a uniform grid.

Many other adaptive strategies are possible. Our point is not an exhaustive comparison, but merely a demonstration that periodic RBFs are very adaptation-tolerant.

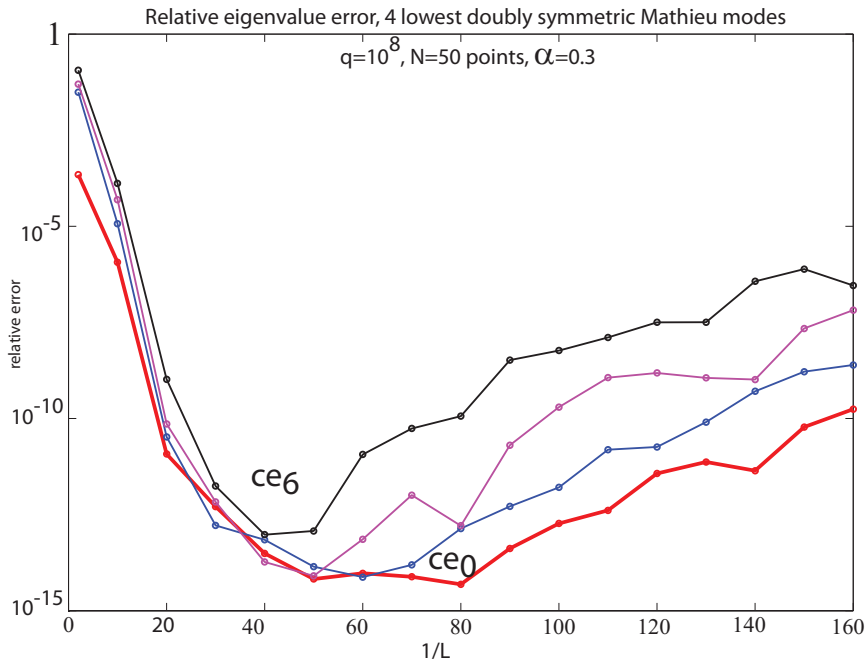

Figure 2.13: Arctan/Tan Mapping: Relative errors, $|\nu_{RBF} - \nu|/\nu$, for the lowest four doubly symmetric Mathieu modes, ce_0, ce_2, ce_4 and ce_6, plotted against the reciprocal of the mapping parameter L . The computations were done using 16-digit floating point precision; the RBF relative width parameter is α=0.3.

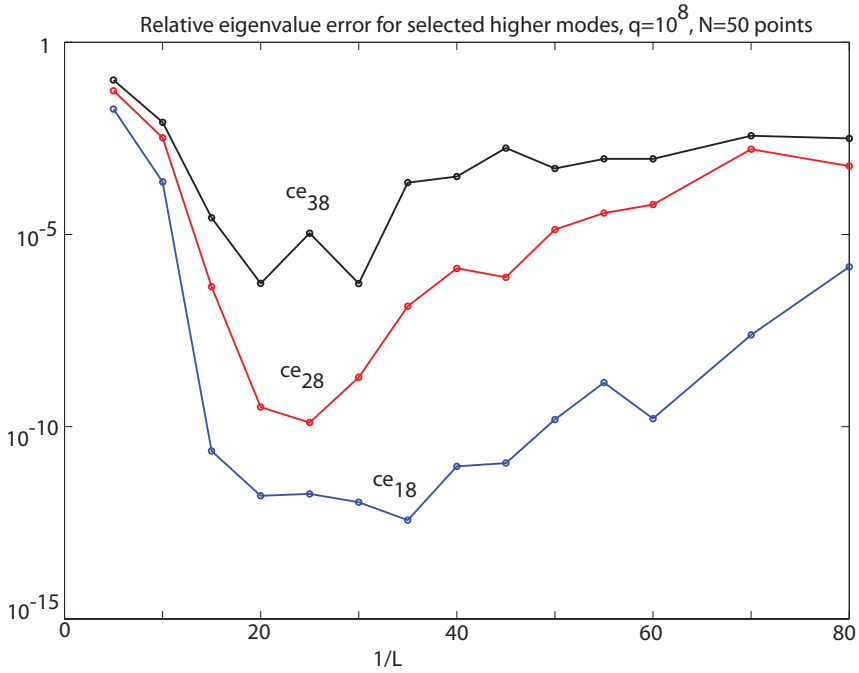

Figure 2.14: Arctan/Tan Mapping: Relative errors, $|\nu_{RBF} - \nu|/\nu$, for three higher doubly symmetric Mathieu modes, ce_{18}, ce_{28} and ce_{38}, plotted against the reciprocal of the mapping parameter L. The computations were done using 16-digit arithmetic with α=0.3.

CHAPTER III

Periodized RBF interpolations with Tikhonov regularization

3.1 Introduction

3.1.1 Motivation

Nonlinear interactions are common in fluid simulations. The nonlinear interactions produce smaller and smaller scales. On a finite grid, the continuous production of small scales will eventually blow up the simulation. This is because as the length scale gets smaller than twice of the grid spacing, the modes cannot be resolved but get aliased to larger wave lengths. This kind of aliasing is not predicted by the dynamic equations. So we have to eliminate the non-physical energy cascade using certain filter techniques. Tikhonov regularization has been a very powerful method in filtering noises. In this chapter, the Tikhonov regularization is studied and applied as a fast implementation of hyperviscosity in fluid simulations using RBF methods.

3.1.2 Tikhonov Regularization

In this section, we will review the principle of Tikhonov regularization as a filter. Further discussion can be seen in (*Hansen*, 1992; *Hansen and O'Leary*, 1993). The Tikhonov solution x_{tk2} solves the problem

$$\min_x(||\vec{\vec{P}}\vec{x} - \vec{b}||_2^2 + \lambda||\vec{x}||_2^2) \tag{3.1}$$

and the solution is formally given by

$$\vec{x}_{tk2} = (\vec{\vec{P}}^T\vec{\vec{P}} + \lambda\vec{\vec{I}})^{-1}\vec{\vec{P}}^T\vec{b} = \vec{\vec{P}}_{tik}^{-1}\vec{b} \tag{3.2}$$

To understand the minimization problem, we need to look at the following overdetermined linear system

$$\begin{bmatrix} \vec{\vec{P}} \\ \sqrt{\lambda}\vec{\vec{I}} \end{bmatrix} \vec{x} = \begin{bmatrix} \vec{b} \\ \vec{0} \end{bmatrix} \tag{3.3}$$

Usually this system has no exact solutions unless vector $\vec{b}$ lies in the range of matrix $\begin{bmatrix} \vec{\vec{P}} \\ \sqrt{\lambda}\vec{\vec{I}} \end{bmatrix}$. Based on *Theorem 11.1* in Trefethen's Book (*Trefethen and Bau*, 1997), the projection of the residual $\begin{bmatrix} \vec{\vec{P}}\vec{x} - \vec{b} \\ \sqrt{\lambda}\vec{\vec{I}}\vec{x} \end{bmatrix}$ onto the column space of matrix $\begin{bmatrix} \vec{\vec{P}} \\ \sqrt{\lambda}\vec{\vec{I}} \end{bmatrix}$ must be zero, that is

$$\begin{bmatrix} \vec{\vec{P}} \\ \sqrt{\lambda}\vec{\vec{I}} \end{bmatrix}^T \begin{bmatrix} \vec{\vec{P}}\vec{x} - \vec{b} \\ \sqrt{\lambda}\vec{\vec{I}}\vec{x} \end{bmatrix} = 0 \tag{3.4}$$

Rearrange Eqn. 3.4, we have

$$[\vec{\vec{P}}^T\vec{\vec{P}} + \lambda\vec{\vec{I}}]\vec{x} = \vec{\vec{P}}^T\vec{b} \tag{3.5}$$

the solution of which is Eqn. 3.2.

For matrix P, the SVD is

$$\vec{\vec{P}} = \vec{\vec{U}}\vec{\vec{S}}\vec{\vec{V}}^T \tag{3.6}$$

where, $\vec{\vec{U}} = [u_1, ..., u_n]$, $\vec{\vec{S}} = diag(s_1, ..., s_n)$, and $\vec{\vec{V}} = [v_1, ..., v_n]$, with $s_1 \geq s_2 \geq s_3 ... \geq s_n \geq 0$. The matrices $\vec{\vec{U}}$ and $\vec{\vec{V}}$ are both unitary, i.e., $\vec{\vec{U}}^{-1} = \vec{\vec{U}}^T$ and $\vec{\vec{V}}^{-1} = \vec{\vec{V}}^T$.

$$\begin{aligned}
\vec{\vec{P}}_{tk2}^{-1} &= (\vec{\vec{P}}^T\vec{\vec{P}} + \lambda\vec{\vec{I}})^{-1}\vec{\vec{P}}^T = (\vec{\vec{V}}\vec{\vec{S}}^T\vec{\vec{U}}^T\vec{\vec{U}}\vec{\vec{S}}\vec{\vec{V}}^T + \lambda\vec{\vec{V}}\vec{\vec{I}}\vec{\vec{V}}^T)^{-1}\vec{\vec{V}}\vec{\vec{S}}^T\vec{\vec{U}}^T \\
&= \vec{\vec{V}}(\vec{\vec{S}}^T\vec{\vec{S}} + \lambda\vec{\vec{I}})^{-1}\vec{\vec{S}}^T\vec{\vec{U}}^T = \vec{\vec{V}}diag(\frac{s_i^2}{s_i^2+\lambda}\frac{1}{s_i})\vec{\vec{U}}^T = \sum_i(\frac{s_i^2}{s_i^2+\lambda}\frac{1}{s_i})\vec{v}_i\vec{u}_i^T \\
&= \sum_i w_i\frac{1}{s_i}\vec{v}_i\vec{u}_i^T
\end{aligned} \tag{3.7}$$

and,

$$\vec{x}_{tk2} = \sum_i w_i\frac{(\vec{u}_i^T\vec{b})}{s_i}\vec{v}_i \tag{3.8}$$

with the weighting function $w_i = {s_i}^2/({s_i}^2 + \lambda)$.

We had seen from the weighting function, as $s_i^2 \to 0$, $w_i \to 0$; while $s_i^2 \gg \lambda$, $w_i \to 1$. This shows that small SVD components have smaller effects on $\vec{x}_{tk2}$. In the limit $\lambda \to 0$, Eqn. (3.8) recovers the non-regularized case with $w_i = 1$.

$$\vec{x} = \sum_i \frac{(\vec{u}_i^T\vec{b})}{s_i}\vec{v}_i \tag{3.9}$$

The Tikhonov regularization has been a very powerful method in filtering out the influence of noise. The choice of the Tikhonov parameter λ is based on the L-curve method. This method was used by Lawson and Hanson (*Lawson and Hanson*, 1974) and further studied by Hansen (*Hansen*, 1992; *Hansen and O'Leary*, 1993). The basic idea is the regularized solution $||\vec{x}_\lambda||_2^2$ is a monotonic decreasing function of λ and the residual $||\vec{\vec{P}}\vec{x} - \vec{b}||_2^2$ is a monotonic increasing function. To understand this argument, we will need to take a look at the regularized solution Eqn. 3.8. The 2-norm of the solution is

$$||\vec{x}||_2^2 = \left[\sum_i w_i\frac{(\vec{u}_i^T\vec{b})}{s_i}\vec{v}_i\right]^T\left[\sum_j w_j\frac{(\vec{u}_j^T\vec{b})}{s_j}\vec{v}_j\right] \tag{3.10}$$

Since $\vec{\vec{V}}$ is unitary and $\vec{v}_i\vec{v}_j = \delta_{i,j}$, therefore

$$||\vec{x}||_2^2 = \sum_i w_i^2 \frac{(\vec{u}_i^T\vec{b})^2}{s_i^2} = \sum_i \left[\frac{{s_i}^2}{{s_i}^2+\lambda}\right]^2 \frac{(\vec{u}_i^T\vec{b})^2}{s_i^2} \tag{3.11}$$

Since $w_i^2 = [\frac{{s_i}^2}{{s_i}^2+\lambda}]^2$ are monotonically decreasing functions of λ and $\frac{(\vec{u}_i^T\vec{b})^2}{s_i^2}$ are all positive, therefore, $||\vec{x}||_2^2$ is a monotonically decreasing function of λ. Next, we have a look at the residual $||\vec{\vec{P}}\vec{x} - \vec{b}||_2^2$, where we will expand $\vec{b} = \sum_i (\vec{u}_i^T\vec{b})\vec{u}_i$ and

$$\vec{\vec{P}}\vec{x} = \sum_i s_i\vec{u}_i\vec{v}_i^T \sum_j w_j \frac{(\vec{u}_j^T\vec{b})}{s_j}\vec{v}_j \tag{3.12}$$

Again, due to the unitarity of matrices $\vec{\vec{U}}$ and $\vec{\vec{V}}$, the residual equation becomes

$$\begin{aligned}
&||\vec{\vec{P}}\vec{x} - \vec{b}||_2^2 \\
&= \left[\sum_i\sum_j s_i w_j \frac{(\vec{u}_j^T\vec{b})}{s_j}\vec{u}_i\delta_{i,j} - \sum_i (\vec{u}_i^T\vec{b})\vec{u}_i\right]^T \left[\sum_k\sum_l s_k w_l \frac{(\vec{u}_l^T\vec{b})}{s_k}\vec{u}_k\delta_{k,l} - \sum_j (\vec{u}_j^T\vec{b})\vec{u}_j\right] \\
&= \left[\sum_i s_i w_i \frac{(\vec{u}_i^T\vec{b})}{s_i}\vec{u}_i - \sum_i (\vec{u}_i^T\vec{b})\vec{u}_i\right]^T \left[\sum_j s_j w_j \frac{(\vec{u}_j^T\vec{b})}{s_j}\vec{u}_j - \sum_j (\vec{u}_j^T\vec{b})\vec{u}_j\right] \\
&= \left[\sum_i (1-w_i)(\vec{u}_i^T\vec{b})\vec{u}_i\right]^T \left[\sum_j (1-w_j)(\vec{u}_j^T\vec{b})\vec{u}_j\right] \\
&= \sum_i \left[(1-w_i)(\vec{u}_i^T\vec{b})\right]^2
\end{aligned} \tag{3.13}$$

Since $0 < w_i < 1$ are monotonic decreasing functions of λ, therefore $0 < 1-w_i < 1$ are monotonic increasing functions of λ. Therefore, residual $||\vec{\vec{P}}\vec{x} - \vec{b}||_2^2$ is a monotonic increasing function of λ. The optimum Tikhonov parameter for the minimization problem Eqn. 3.1 is a λ that balances the two competing terms: the increasing $||\vec{\vec{P}}\vec{x} - \vec{b}||_2^2$ and the decreasing $||\vec{x}||_2^2$. A plot of $||\vec{x}||_2^2$ against $||\vec{\vec{P}}\vec{x} - \vec{b}||_2^2$ is a L-shape curve, which is called *L-curve*(*Calvetti et al.*, 2000). The *L-curve* has been a guide to select

the optimum Tikhonov parameter for the minimization problem 3.1.

3.1.3 Tikhonov regularization for global RBF interpolation

Hickernell and Hon (1999) interpreted Radial Basis Function interpolation with Tikhonov regularization as roughness-minimizing splines. For global RBF interpolation, the interpolation matrix is symmetric and positive definite. The interpolation matrix can then be written as

$$\vec{\vec{P}} = \vec{\vec{U}}\vec{\vec{S}}\vec{\vec{U}}^T \tag{3.14}$$

The Tikhonov regularized solution becomes

$$\vec{x}_{tk2} = \sum_i w_i \frac{(\vec{u}_i^T \vec{b})}{s_i} \vec{u}_i \tag{3.15}$$

where, as shown before, $w_i = \frac{s_i^2}{s_i^2+\lambda}$, $\lambda > 0$. Fig. 3.1 shows the weighting function of Tikhonov regularization with three different parameters.

3.2 Properties of the 1D Periodic Gaussian RBF(PGARBF) interpolation

Suppose we have a set of grid points $x_0 = -\pi, x_1 = -\pi + h, ..., x_{N-1} = \pi - h$; to satisfy periodic condition, we let $x_{i+mN} = x_i$. To interpolate a function using PGARBF, we will have to solve the linear system,

$$\vec{\vec{A}}\vec{a} = \vec{f}; \tag{3.16}$$

where, matrix $\vec{\vec{A}}$ is a symmetric matrix with elements $\vec{\vec{A}}_{i,j} = \sum_{m=-\infty}^{+\infty} \phi(x_i - x_j + 2m\pi)$. $\phi(x)$ is the Gaussian radial basis function. However, the following properties can similarly be applied to other PRBFs.

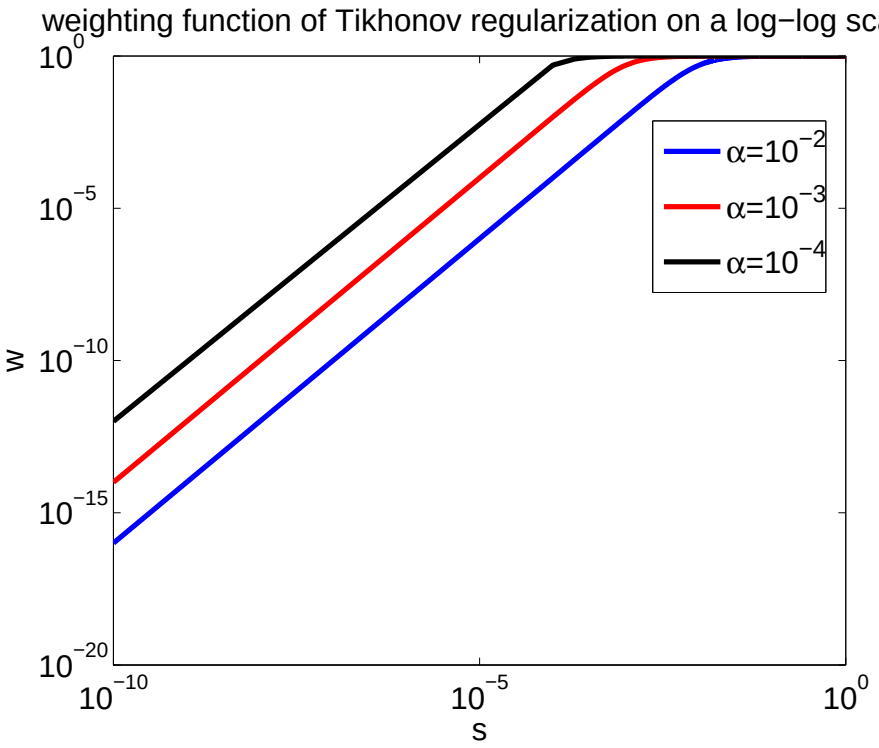

Figure 3.1: Weighting function $\frac{s_i^2}{s_i^2+\lambda}$ of Tikhonov regularization using different $\lambda = \alpha^2$. The figure is plotted on a log-log scale. The curves start to decay at $s_i \approx \alpha$. When s_i is greater than α, the weighting function is close to unity. As s_i approaches zero, weighting function decays rapidly, and effectively filters out components of small singular values.

3.2.1 The 1D PGARBF interpolation matrix: a circulant matrix

Definition III.1. Let $\vec{\vec{A}}$ be an $N \times N$ matrix. $\vec{\vec{A}}$ is a circulant matrix if and only if

$$\vec{\vec{A}}_{i,j} = \vec{\vec{A}}_{k,l}, \text{ whenever } j - i \equiv l - k (\text{mod N})$$

Theorem III.2. *The 1D Periodic Gaussian RBF interpolation matrices are symmetric and circulant.*

Proof. Circulant:

On a uniform grid, $x_i = ih$, $2\pi = Nh$,

$$\vec{\vec{A}}_{i,j} = \sum_{m=-\infty}^{+\infty} \phi(|x_i - x_j + 2m\pi|) = \sum_{m=-\infty}^{+\infty} \phi(|(j - i + mN)h|) \tag{3.17}$$

Eqn. (3.17) shows $\vec{\vec{A}}$ is just a function of $j-i$. All elements $\vec{\vec{A}}_{k,l}$ with $j-i \equiv k-l \pm mN$ are the same.

Symmetric:

$$\vec{\vec{A}}_{i,j} = \sum_{m=-\infty}^{+\infty} \phi(|(i-j+mN)h|) = \sum_{m=-\infty}^{+\infty} \phi(|(j-i-mN)h|) = \vec{\vec{A}}_{j,i} \tag{3.18}$$

□

3.2.2 Positive definiteness of the PGARBF interpolation matrix

Definition III.3. A real symmetric matrix $\vec{\vec{A}}$ is called positive definite if its associated quadratic form is non-negative, i.e.,

$$\sum_{j=1}^{N}\sum_{k=1}^{N} c_j c_k \vec{\vec{A}}_{j,k} \geq 0 \tag{3.19}$$

for a non-zero $\vec{c} = [c_1, ..., c_N]^T \in \mathbb{R}^N$, and it is zero only for $\vec{c} \equiv 0$.

Theorem III.4. *The 1D Periodic Gaussian RBF interpolation matrix is positive definite.*

Proof. The Fourier series of a Periodized Gaussian RBF is the Jacobian Theta function*Boyd and Xiao* (2013),

$$\begin{aligned}
\theta^{GA}(x;\epsilon) &= \sum_{m=-\infty}^{\infty} \exp(-\epsilon^2(x-2m\pi)^2) \\
&= \frac{1}{2\epsilon\sqrt{\pi}}\{1 + 2\sum_{n=1}^{\infty} \exp(-\frac{n^2}{4\epsilon^2})\cos(nx)\} \\
&= \frac{1}{2\epsilon\sqrt{\pi}}\{\sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2})\exp(inx)\}
\end{aligned} \tag{3.20}$$

The relation between Jacobian theta fuction and PGARBF can be derived as follows: the periodicity of the Gaussian imbricated series restricts the wave numbers to be

integers only; and then we do a regular Fourier transform of a single Gaussian centered at $x_i = 0$ to obtain the Fourier coefficients, and then an inverse Fourier transform is applied.

$$\begin{aligned}
\sum_{i=1}^{N}\sum_{j=1}^{N} c_i c_j \vec{\vec{A}}_{i,j} &= \sum_{i=1}^{N}\sum_{j=1}^{N} c_i c_j \theta^{GA}(x_i - x_j, \epsilon) \\
&= \frac{1}{2\epsilon\sqrt{\pi}} \sum_{i=1}^{N}\sum_{j=1}^{N} c_i c_j \{ \sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \exp(in(x_i - x_j)\} \\
&= \frac{1}{2\epsilon\sqrt{\pi}} \sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \{ \sum_{i=1}^{N}\sum_{j=1}^{N} c_i c_j \exp(inx_i) \exp(-inx_j)\} \quad (3.21) \\
&= \frac{1}{2\epsilon\sqrt{\pi}} \sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \{ \sum_{i=1}^{N} c_i \exp(inx_i) \sum_{j=1}^{N} c_j \exp(-inx_j)\} \\
&= \frac{1}{2\epsilon\sqrt{\pi}} \sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \mid \sum_{i=1}^{N} c_i \exp(inx_i) \mid^2 \geq 0
\end{aligned}$$

So the PGARBF interpolation matrix is positive definite. All the eigenvalues of $\vec{\vec{A}}$ are positive. □

3.2.3 Eigen-analysis of the PGARBF interpolation matrix

The PGARBF interpolation matrices are positive definite, symmetric and circulant.

Theorem III.5. *The eigenvalues of a circulant matrix is given by Hubbert and Mueller (2006)*

$$\lambda_j = \sum_{l=0}^{N} c_l e^{-i\frac{2\pi}{N}jl} = \sum_{l=0}^{N} c_l e^{-ihjl}, j = 0, 1, ..., N-1. \quad (3.22)$$

and, the corresponding eigenvectors are

$$\vec{u}_j = (1, e^{-ihj}, e^{-i2hj}, ..., e^{-i(N-1)hj})^T \quad (3.23)$$

where, $h = \frac{2\pi}{N}$*,* c_l *are the elements of first row of matrix* $\vec{\vec{A}}$*, i.e.,* $\vec{\vec{A}}_{1,l}$*.*

Corollary III.6. *The eigenvalues of the PGARBF interpolation matrix are*

$$\lambda_j = \frac{\sqrt{\pi}}{\alpha} \exp(-\frac{j^2\pi^2}{N^2\alpha^2}) \tag{3.24}$$

and, the corresponding eigenvectors are

$$u_j(\vec{x}) = \exp(-ij\vec{x}) \tag{3.25}$$

where, $u_j(\vec{x}) = [\exp(-ijx_0), \exp(-ijx_1), ..., \exp(-ijx_{N-1})]$*.*

Proof. The first row of the PGARBF interpolation matrix $\vec{\vec{A}}_{1,l} = c_l$. On a uniform grid, $x_l = lh, h = \frac{2\pi}{N}$,

$$\begin{aligned} c_l &= \frac{1}{2\epsilon\sqrt{\pi}}\{\sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \exp(in(x_0 - x_l))\} \\ &= \frac{1}{2\epsilon\sqrt{\pi}}\{\sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \exp(-inlh)\} \end{aligned} \tag{3.26}$$

From Theorem(III.5), the eigenvalues of matrix $\vec{\vec{A}}$ are

$$\begin{aligned} \lambda_j &= \sum_{l=0}^{N-1} c_l \exp(-ijlh) \\ &= \sum_{l=0}^{N-1} \frac{1}{2\epsilon\sqrt{\pi}}\{\sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \exp(-inlh) \exp(-ijlh)\} \\ &= \frac{1}{2\epsilon\sqrt{\pi}} \sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2})\{\sum_{l=0}^{N-1} \exp(-i(n+j)lh)\} \\ &= \frac{1}{2\epsilon\sqrt{\pi}} \sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2})\{N\delta(n+j)\} \end{aligned} \tag{3.27}$$

Since, if $n + j \neq 0$,

$$\begin{aligned}\sum_{l=0}^{N-1} \exp(-i(n+j)lh) &= \frac{1-\exp(-i(n+j)h)^N}{1-\exp(-i(n+j)h)} = \frac{1-\exp(-i(n+j)Nh)}{1-\exp(-i(n+j)h)} \\ &= \frac{1-\exp(-i(n+j)N\frac{2\pi}{N})}{1-\exp(-i(n+j)h)} = \frac{1-\exp(-i(n+j)2\pi)}{1-\exp(-i(n+j)h)} \\ &= 0\end{aligned} \tag{3.28}$$

if $n + j = 0$, then,

$$\sum_{l=0}^{N-1} \exp(-i(n+j)lh) = \sum_{l=0}^{N-1} \exp(-i0) = N \tag{3.29}$$

Therefore,

$$\lambda_j = \frac{N}{2\epsilon\sqrt{\pi}} \exp(-\frac{j^2}{4\epsilon^2}) = \frac{\sqrt{\pi}}{\alpha} \exp(-\frac{j^2\pi^2}{N^2\alpha^2}) \tag{3.30}$$

Here, we have used $\epsilon = \frac{\alpha}{h} = \frac{N\alpha}{2\pi}$.

Since $x_l = lh$, the eigenvector is simply $u_j(\vec{x}) = \exp(-ij\vec{x})$. The notation $\exp(-ij\vec{x})$ means $[\exp(-ijx_0), \exp(-ijx_1), ..., \exp(-ijx_{N-1})]$. □

3.2.4 Properties of 1D harmonic operator discretized using PGARBF

In this section, we study the properties of 1D harmonics operator, defined as $-\frac{d^2}{dx^2}$, discretized using PGARBF.

$$\begin{aligned}\theta_{xx}^{GA}(x;\epsilon) &= -\frac{d^2}{dx^2}\{\sum_{m=-\infty}^{\infty} \exp(-\epsilon^2(x-2m\pi)^2)\} \\ &= -\frac{d^2}{dx^2}\{\frac{1}{2\epsilon\sqrt{\pi}} \sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2}) \exp(inx)\} \\ &= \frac{1}{2\epsilon\sqrt{\pi}}\{\sum_{n=-\infty}^{\infty} n^2 \exp(-\frac{n^2}{4\epsilon^2}) \exp(inx)\}\end{aligned} \tag{3.31}$$

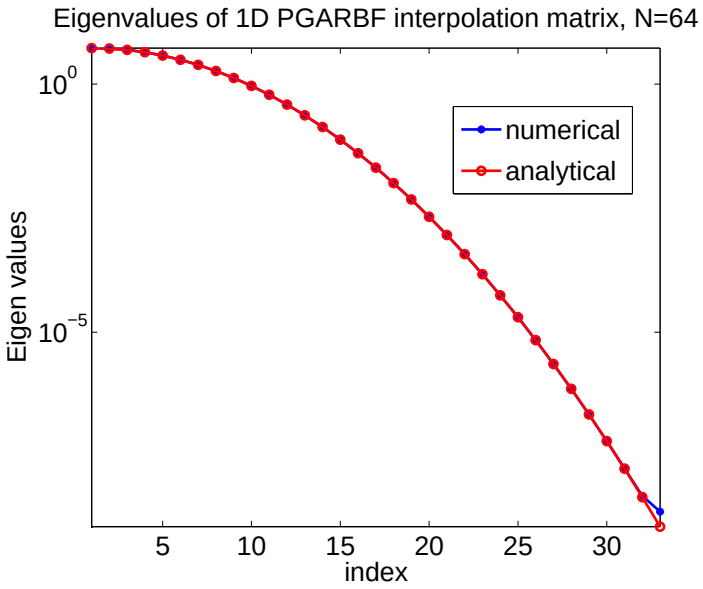

Figure 3.2: Eigen spectrum of 1D PGARBF interpolation matrix. Numerical values and analytical values fall on the same curve.

The analysis of 1D harmonic operator is quite similar to that of Section 3.2. The discretized form of the 1D harmonic operator is $\vec{\vec{P}}$, whose element $\vec{\vec{P}}_{j,k}$ is given by

$$\vec{\vec{P}}_{j,k} = \theta_{xx}^{GA}(x_j - x_k; \epsilon) = \frac{1}{2\epsilon\sqrt{\pi}}\{\sum_{n=-\infty}^{\infty} n^2 \exp(-\frac{n^2}{4\epsilon^2}) \exp(in(x_j - x_k))\} \tag{3.32}$$

Then, the properties of matrix $\vec{\vec{P}}$ immediately follows from Section 3.2 that

1. Matrix $\vec{\vec{P}}$ is symmetric, circulant and positive definite.

2. The eigenvalues of $\vec{\vec{P}}$ are given $\mu_j = \frac{\sqrt{\pi}}{\alpha} j^2 \exp(-\frac{j^2\pi^2}{N^2\alpha^2})$, and the eigenvectors are trigonometric functions.

3.2.5 Condition number of matrices $\vec{\vec{A}}$ and $\vec{\vec{P}}$

The spectrum of matrices $\vec{\vec{A}}$ and $\vec{\vec{P}}$ are $\lambda_1, \lambda_2, ..., \lambda_M$ and $\mu_1, \mu_2, ..., \mu_M$, respectively, with $M = N/2$. This comes from the fact that, numerically, each eigenvalue (with eigenvector $\exp(-in\vec{x})$) actually corresponds to two real eigenvectors, i.e., $\sin(nx), \cos(nx)$. And therefore, on a N points grid, we can only resolve $M = N/2$

eigenvalues. The condition number of matrices, $\vec{\vec{A}}$ and $\vec{\vec{P}}$, are $C(\vec{\vec{A}}) = \frac{\lambda_{max}}{\lambda_{min}}$ and $C(\vec{\vec{P}}) = \frac{\mu_{max}}{\mu_{min}}$ accordingly. Based on the eigenvalues given in the previous sections, we have

$$C(\vec{\vec{A}}) = \frac{\sqrt{\pi}/\alpha}{(\sqrt{\pi}/\alpha)\exp(-\frac{(N/2)^2\pi^2}{N^2\alpha^2})} = \exp(\frac{\pi^2}{4\alpha^2}) \tag{3.33}$$

To find the maximum of the spectrum of matrix $\vec{\vec{P}}$, we have,

$$\frac{d\mu(j)}{dj} = 2\frac{\sqrt{\pi}}{\alpha} j \exp(-\frac{j^2\pi^2}{N^2\alpha^2}) - (\frac{\sqrt{\pi}}{\alpha} j^2)\frac{2j\pi^2}{N^2\alpha^2}\exp(-\frac{j^2\pi^2}{N^2\alpha^2}) = 0 \tag{3.34}$$

Solve Eqn. (3.34), we have

$$j_{max} = \frac{N\alpha}{\pi} \tag{3.35}$$

So,

$$\mu_{max} = \frac{\sqrt{\pi}}{\alpha}\frac{N^2\alpha^2}{\pi^2}\exp(-1) \tag{3.36}$$

The smallest eigenvalue μ_{min} is

$$\mu_{min} = \frac{\sqrt{\pi}}{\alpha}(\frac{N}{2})^2\exp(-\frac{\pi^2}{4\alpha^2}) \tag{3.37}$$

So the condition number of matrix $\vec{\vec{P}}$ is

$$C(\vec{\vec{P}}) = \frac{\mu_{max}}{\mu_{min}} = \frac{4\alpha^2}{\pi^2}\exp(\frac{\pi^2}{4\alpha^2} - 1) \tag{3.38}$$

Fig. (3.4) shows the condition number of matrix $\vec{\vec{A}}$. One merit has to be noted here is, the condition number of matrix $\vec{\vec{A}}$ and $\vec{\vec{P}}$ are independent of the number of grid points N. Only the shape parameter α enters the expression of condition number. Eqn. 3.33 agrees with the condition number of Boyd's numerical result (*Boyd and Gildersleeve*, 2011), where the analytical expression is given by fitting

the numerical condition number on a non-periodic domain. However, for a Gaussian RBF, the difference between periodic and non-periodic condition numbers is quite small. Since the GARBF decays very rapidly, and only at the boundary, can we see the small difference of the interpolation matrix, and therefore it is understandable that the two results are the same. For a moderate shape parameter, i.e., $\alpha \approx 0.4$, PGARBF interpolation is accurate for arbitrary number of grid points. While, as we decrease α, the condition number increases exponentially, hence, numerically stable PGARBF interpolation via direct matrix inversion is impossible. Fornberg (*Driscoll and Fornberg*, 2002; *Fornberg and Wright*, 2004) proposed several schemes to do a stable computation in the $\alpha \to 0$ limit on a non-periodic domain.

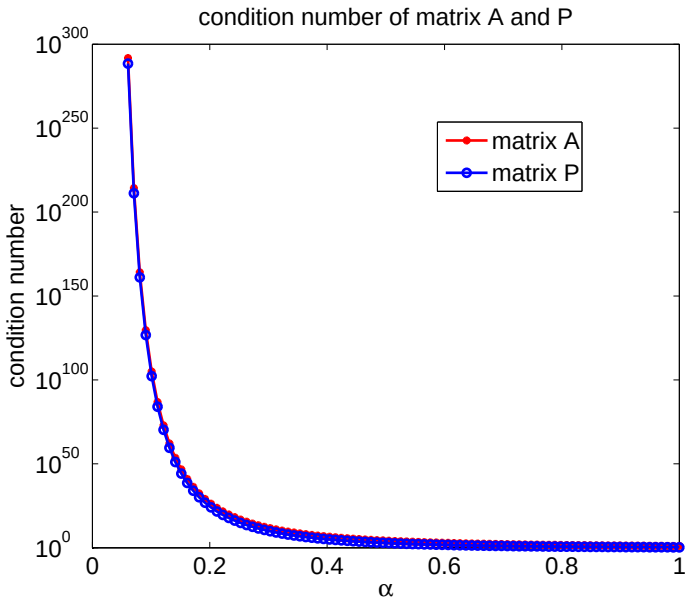

Figure 3.3: The condition number of 1D PGARBF interpolation matrix and 1D harmonic operator discretized using PGARBF. The shape parameter used in this computation is $\alpha = 1/3$.

3.3 Eigenvalues of other PRBF interpolation matrices

To compute the Eigenvalues of other PRBF interpolation matrices, we need to prove a very general theorem.

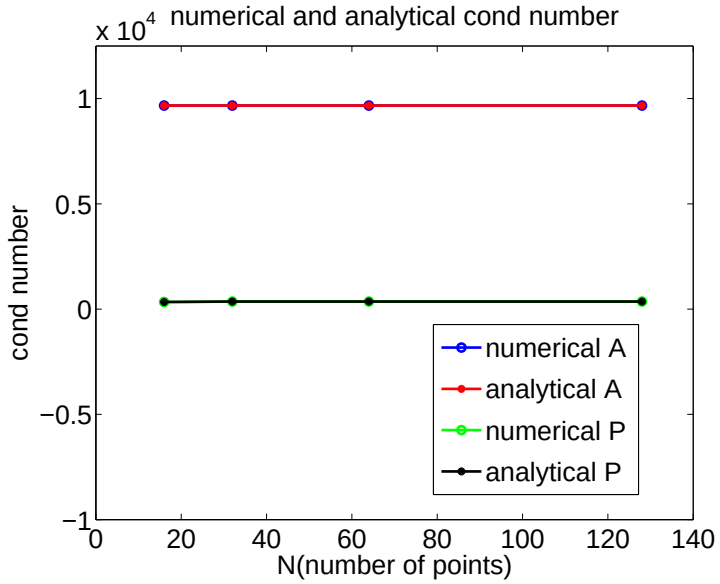

Figure 3.4: The numerical and analytical condition number of 1D PGARBF interpolation matrix and 1D harmonic operator discretized using PGARBF. The shape parameter used in this computation is $\alpha = 1/3$.

Theorem III.7. *A periodic RBF basis function has the form* $\phi(x) = \sum_{n=-\infty}^{\infty} \phi(n) \exp(-inx)$ *on a periodic domain* $[0, 2\pi]$*, then, the eigenvalue of the interpolation matrix* $\vec{\vec{A}}$ *is given by* $N\phi(j)$*, where* $\vec{\vec{A}}_{ij} = \phi(x_i - x_j)$*,* $x_i = ih, i = 0, ..., N-1$ *with* $h = \frac{2\pi}{N}$.

Proof. It is straightforward to see the interpolation matrix $\vec{\vec{A}}$ is circulant. It follows from Theorem. III.5 that the eigenfunction of this function is $\vec{u}_j = (1, e^{-ihj}, ..., e^{-i(N-1)hj})^T$. The eigenvalue of matrix $\vec{\vec{A}}$ is given by

$$\begin{aligned}
\lambda_j &= \sum_{l=0}^{N} c_l e^{-i\frac{2\pi}{N}jl} = \sum_{l=0}^{N} c_l e^{-ijlh} = \sum_{l=0}^{N} \sum_{n=-\infty}^{\infty} \phi(n) e^{-in(x_0 - x_l)} e^{-ijlh} \\
&= \sum_{n=-\infty}^{\infty} \phi(n) \{ \sum_{l=0}^{N} e^{-ilh(n-j)} \} = \sum_{n=-\infty}^{\infty} \phi(n)(N\delta_{n,j}) \\
&= N\phi(j)
\end{aligned} \tag{3.39}$$

□

The second line we have switched the two summations and replaced $x_j = jh$. In 1D space, for other types of PRBFs, i.e., Periodic Inverse Quadratic, Periodized

Sech, Periodic Multiquadric and Periodized Inverse Multiquadric, we have given the Fourier series representations in the paper (*Boyd and Xiao*, 2013). From there, we can compute the eigenvalues of the interpolation matrices of these PRBFs by Theorem III.7. Table 3.1 summarizes the eigenvalues of the interpolation matrices of the five species of periodized RBFs. In Table 3.1, we had used the fact $\epsilon = \frac{\alpha}{h} = \frac{N\alpha}{2\pi}$.

Table 3.1: The Fourier transform of the five Periodized RBF species: Periodic Gaussian, Periodic Inverse Quadratic, Periodic Hyperbolic Secant, Periodic Multiquadric and Periodic Inverse Multiquadric and the corresponding eigenvalues of the interpolation matrices.

PRBFs	Fourier Form	Eigenvalues
PGA	$\frac{1}{2\epsilon\sqrt{\pi}}\{\sum_{n=-\infty}^{\infty}\exp(-\frac{n^2}{4\epsilon^2})\exp(inx)\}$	$\frac{\sqrt{\pi}}{\alpha}\exp(-\frac{j^2\pi^2}{N^2\alpha^2})$
PIQ	$\frac{1}{2\epsilon}\{\sum_{n=-\infty}^{\infty}\exp(-\frac{n}{\epsilon})\exp(inx)\}$	$\frac{\pi}{\alpha}\exp(-\frac{2\pi j}{N\alpha})$
Psech	$\frac{1}{\epsilon}\{\sum_{n=-\infty}^{\infty}\frac{q^n}{1+q^{2n}}\exp(inx)\}$	$\frac{2\pi}{\alpha}\frac{q^j}{1+q^{2j}}$
PMQ	$\sum_{n=-\infty}^{\infty}\frac{K_1(n/\epsilon)}{n\pi}\exp(inx)$	$\frac{NK_1(2\pi j/(N\alpha))}{j\pi}$
PIMQ	$\sum_{n=-\infty}^{\infty}\frac{K_0(n/\epsilon)}{\epsilon\pi}\exp(inx)$	$\frac{2K_0(2\pi j/(N\alpha))}{\alpha}$

3.4 Properties of 2D PGARBF interpolation

To extend the argument of 1D PGARBF to a 2D problem, we will be working on a 2D tensor product grid, on a $[0, 2\pi]\times[0, 2\pi]$ domain. The grid is defined as (x_j, y_k) with $x_j = jh, y_k = kh$, where $h = \frac{2\pi}{N}$, and N is the grid size in one direction. A 2D PGARBF basis is defined as,

$$\begin{aligned}
\theta^{GA,2D}(x,y;\epsilon) &= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty}\exp(-\epsilon^2((x+2n\pi)^2+(y+2m\pi)^2)) \\
&= \sum_{n=-\infty}^{\infty}\exp(-\epsilon^2(x+2n\pi)^2)\sum_{m=-\infty}^{\infty}\exp(-\epsilon^2(y+2m\pi)^2) \\
&= \frac{1}{4\epsilon^2\pi}\{\sum_{n=-\infty}^{\infty}\exp(-\frac{n^2}{4\epsilon^2})\exp(inx)\}\{\sum_{m=-\infty}^{\infty}\exp(-\frac{m^2}{4\epsilon^2})\exp(imy)\} \\
&= \theta^{GA}(x;\epsilon)\theta^{GA}(y;\epsilon)
\end{aligned} \tag{3.40}$$

3.4.1 Interpolation using 2D PGARBF

Suppose, $\vec{\vec{A}}^x$ is the PGARBF interpolation matrix in x direction, with elements $\vec{\vec{A}}^x_{j,k} = \theta^{GA}(x_j - x_k; \epsilon)$; and $\vec{\vec{A}}^y$ is the PGARBF interpolation matrix in y direction, with elements $\vec{\vec{A}}^y_{j,k} = \theta^{GA}(y_j - y_k; \epsilon)$, then the 2D PGARBF interpolation matrix on this 2D grid is simply the tensor product of the two interpolation matrices, that is,

$$\vec{\vec{T}} = \vec{\vec{A}}^x \otimes \vec{\vec{A}}^y \tag{3.41}$$

Theorem III.8. *Suppose, $\vec{\vec{A}}$ has eigenvalues λ_j and eigenvectors $\vec{U}_j$, $\vec{\vec{B}}$ has eigenvalues μ_k and eigenvectors $\vec{V}_k$, then, $\vec{\vec{A}} \otimes \vec{\vec{B}}$ has eigenvalues $\lambda_j \mu_k$, with the corresponding eigenvectors $\vec{U}_j \otimes \vec{V}_k$.*

Corollary III.9. *The eigenvalues of a 2D PGARBF interpolation matrix $\vec{\vec{T}}$ are $\gamma = \frac{\pi}{\alpha^2} \exp(-\frac{(j^2+k^2)\pi^2}{N^2\alpha^2})$, with eigenvectors $\exp(-ij\vec{x}) \otimes \exp(-ik\vec{y})$.*

Proof. The 2D PGARBF interpolation matrix $\vec{\vec{T}} = \vec{\vec{A}}^x \otimes \vec{\vec{T}}^y$. $\vec{\vec{A}}^x$ has eigenvalues $\frac{\sqrt{\pi}}{\alpha} \exp(-\frac{j^2\pi^2}{N^2\alpha^2})$ and the corresponding eigenvector $\exp(-ij\vec{x})$. $\vec{\vec{A}}^y$ has eigenvalues $\frac{\sqrt{\pi}}{\alpha} \exp(-\frac{k^2\pi^2}{N^2\alpha^2})$ and the corresponding eigenvector $\exp(-ij\vec{y})$. From Theorem III.8, we have, the eigenvalues of $\vec{\vec{T}}$ are

$$\begin{aligned} \gamma &= \frac{\sqrt{\pi}}{\alpha} \exp(-\frac{j^2\pi^2}{N^2\alpha^2}) \frac{\sqrt{\pi}}{\alpha} \exp(-\frac{k^2\pi^2}{N^2\alpha^2}) \\ &= \frac{\pi}{\alpha^2} \exp(-\frac{(j^2+k^2)\pi^2}{N^2\alpha^2}) \end{aligned} \tag{3.42}$$

The corresponding eigenvectors of $\vec{\vec{T}}$ are $\exp(-ij\vec{x}) \otimes \exp(-ik\vec{y})$. □

The following are the spectrum of the 2D PGARBF interpolation. Since the spectrum are highly degenerated, that is, usually there are several eigenvectors corresponding to a single eigenvalue, so we just counted this single eigenvalue once, and compared with the analytical value. The Fig. 3.4.1 verified Corollary III.9. Fig. 3.5 showed 2 eigenmodes of the 2D PGARBF interpolation matrix.

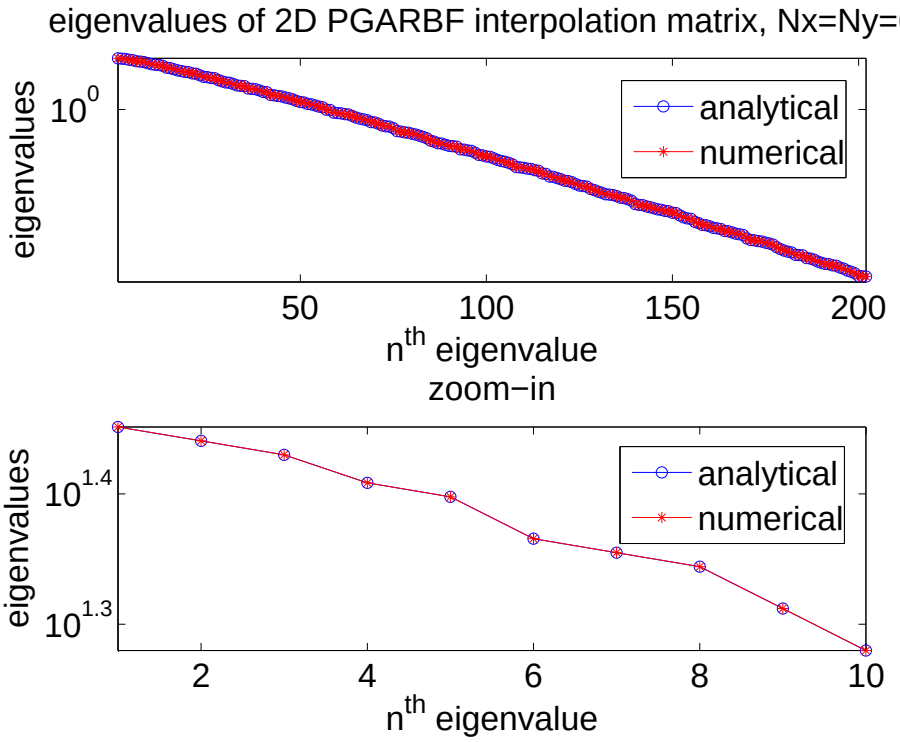

Figure 3.5: The spectrum on a 2D periodic domain using PGARBF. The relative shape parameter $\alpha = 1/3$. In this figure, we counted each distinct eigenvalue only once. Numerical and Analytical values are the same.

3.4.2 2D harmonic operator discretized using PGARBF

The 2D harmonic operator is defined as, $\triangle = -(\frac{d^2}{dx^2} + \frac{d^2}{dy^2})$. Apply this operator to a 2D PGARBF basis, we have

$$\begin{aligned}
\triangle\theta^{GA,2D}(x,y;\epsilon) &= \triangle\{\theta^{GA}(x;\epsilon)\theta^{GA}(y;\epsilon)\} \\
&= \frac{1}{4\epsilon^2\pi}\{\sum_{n=-\infty}^{\infty} n^2 \exp(-\frac{n^2}{4\epsilon^2})\exp(inx)\}\{\sum_{m=-\infty}^{\infty}\exp(-\frac{m^2}{4\epsilon^2})\exp(imy)\} \\
&+ \frac{1}{4\epsilon^2\pi}\{\sum_{n=-\infty}^{\infty} \exp(-\frac{n^2}{4\epsilon^2})\exp(inx)\}\{\sum_{m=-\infty}^{\infty} m^2\exp(-\frac{m^2}{4\epsilon^2})\exp(imy)\} \\
&= \theta^{GA}_{xx}(x;\epsilon)\theta^{GA}(y;\epsilon) + \theta^{GA}(x;\epsilon)\theta^{GA}_{yy}(y;\epsilon)
\end{aligned} \tag{3.43}$$

Again, the discretization of the 2D harmonic operator using PGARBF is $\vec{\vec{P}}^{2D}$, which is given by

$$\vec{\vec{P}}^{2D} = \vec{\vec{P}}^{x} \otimes \vec{\vec{A}}^{y} + \vec{\vec{A}}^{x} \otimes \vec{\vec{P}}^{y} \tag{3.44}$$

where, $\vec{\vec{P}}^{x}$ and $\vec{\vec{P}}^{y}$ are the discretization of 1D harmonic operator in x and y direction respectively; Similarly, $\vec{\vec{A}}^{x}$ and $\vec{\vec{A}}^{y}$ are the PGARBF interpolation matrices in x and y direction respectively.

Theorem III.10. *The eigenvalues of a 2D PGARBF interpolation matrix are given by*

$$\nu = \frac{\pi}{\alpha^2}(j^2 + k^2)\exp(-\frac{(j^2 + k^2)\pi^2}{N^2\alpha^2}) \tag{3.45}$$

Proof. The eigenvalue of $\vec{\vec{P}}^{x}$ is $\frac{\sqrt{\pi}}{\alpha} j_1^2 \exp(-\frac{j_1^2\pi^2}{N^2\alpha^2})$, and the corresponding eigenvector is $\exp(-ij_1\vec{x})$. The eigenvalue of $\vec{\vec{A}}^{x}$ is $\frac{\sqrt{\pi}}{\alpha}\exp(-\frac{k_1^2\pi^2}{N^2\alpha^2})$, with the corresponding eigenvector $\exp(-ik_1\vec{x})$. The same arguments applies to $\vec{\vec{P}}^{y}$ and $\vec{\vec{A}}^{y}$.
Suppose the eigenvalue of operator $\vec{\vec{P}}^{2D}$ is ν, and the corresponding eigenvector is $\exp(-im\vec{x}) \otimes \exp(-in\vec{y})$. Then,

$$\vec{\vec{P}}^{2D}\{\exp(-im\vec{x}) \otimes \exp(-in\vec{y})\} = \nu\{\exp(-im\vec{x}) \otimes \exp(-in\vec{y})\} \tag{3.46}$$

Since,

$$\begin{aligned} &\vec{\vec{P}}^{x} \otimes \vec{\vec{A}}^{y}\{\exp(-ij_1\vec{x}) \otimes \exp(-ik_1\vec{y})\} \\ &= (\frac{\pi}{\alpha^2} j_1^2 \exp(-\frac{(j_1^2 + k_1^2)\pi^2}{N^2\alpha^2}))\{\exp(-ij_1\vec{x}) \otimes \exp(-ik_1\vec{y})\} \end{aligned} \tag{3.47}$$

and,

$$\begin{aligned} &\vec{\vec{A}}^{x} \otimes \vec{\vec{P}}^{y}\{\exp(-ij_2\vec{x}) \otimes \exp(-ik_2\vec{y})\} \\ &= (\frac{\pi}{\alpha^2} k_2^2 \exp(-\frac{(j_2^2 + k_2^2)\pi^2}{N^2\alpha^2}))\{\exp(-ij_2\vec{x}) \otimes \exp(-ik_2\vec{y})\} \end{aligned} \tag{3.48}$$

Compare the above three eigen equations, we have $m = j_1 = j_2$; and $n = k_1 = k_2$. We let $m = j$ and $n = k$, then,

$$\begin{aligned}\nu &= \frac{\pi}{\alpha^2} j^2 \exp(-\frac{(j^2+k^2)\pi^2}{N^2\alpha^2}) + \frac{\pi}{\alpha^2} k^2 \exp(-\frac{(j^2+k^2)\pi^2}{N^2\alpha^2}) \\ &= \frac{\pi}{\alpha^2}(j^2+k^2)\exp(-\frac{(j^2+k^2)\pi^2}{N^2\alpha^2})\end{aligned} \tag{3.49}$$

□

3.5 Filter high frequency waves using Tikhonov regularization

In this section, we will prove how the Tikhonov regularization damps high frequency components in an interpolation process which is followed by two numerical experiments using Tikhonov regularization. As we have seen from the eigen analysis of 1D and 2D PGARBF interpolation matrices, the eigen modes are simply Fourier series. The eigenvalues of the PGARBF interpolation matrices decay exponentially.

Theorem III.11. *Applying Tikhonov regularization with regularization parameter λ to an interpolation problem is effectively applying a filter to the problem. In the space of eigenvectors, an eigenvector with eigenvalue λ_j is damped by a factor of w_j, where w_j are the weighting function of Tikhonov regularization.*

Proof. Suppose, we have a grid $\vec{x}_1, \vec{x}_2, ..., \vec{x}_N$. The interpolation of a function $\psi(\vec{x})$ can be written as $\psi_N(\vec{x}) = \sum_{j=1}^{N} a_j \phi_j(\vec{x})$, where in our case, $\phi_j(\vec{x})$ is the PRBF basis corresponding to center $\vec{x}_j$, i.e., $\phi_j(\vec{x}) = \theta^{GA}(\vec{x} - \vec{x}_j; \epsilon)$ for Periodic Gaussian RBF. This interpolation equation is forced to satisfy the interpolation condition on all the grid points, that is,

$$\psi_N(\vec{x}_j) = \psi(\vec{x}_j), j = 1, ..., N; \tag{3.50}$$

The above interpolation is written in a matrix-vector product form

$$\vec{\vec{A}}\vec{a} = \vec{\psi} \tag{3.51}$$

where, matrix $\vec{\vec{A}}$ is the PRBF interpolation matrix with elements $\vec{\vec{A}}_{i,j} = \phi_j(\vec{x}_i)$, and $\vec{a}$ is a vector containing the expansion coefficients. Suppose, matrix $\vec{\vec{A}}$ has N eigenvectors $\vec{u}_j$ forming a set of complete basis in an N dimensional space. We can rewrite $\vec{\vec{A}}$ as,

$$\vec{\vec{A}} = \sum_{j}^{N} \lambda_j \vec{u}_j \vec{u}_j^T \tag{3.52}$$

In a Tikhonov regularization interpolation,

$$\vec{a}_{new} = \sum_{j=1}^{N} w_j \frac{\vec{u}_j^T \vec{\psi}_{old}}{\lambda_j} \vec{u}_j \tag{3.53}$$

$$\begin{aligned} \vec{\psi}_{new} &= \sum_{i}^{N} \lambda_i \vec{u}_i \vec{u}_i^T \vec{a}_{new} = (\sum_{i}^{N} \lambda_j \vec{u}_i \vec{u}_i^T)(\sum_{j=1}^{N} w_j \frac{\vec{u}_j^T \vec{\psi}_{old}}{\lambda_j} \vec{u}_j) \\ &= \sum_{i}^{N} \lambda_i \vec{u}_i w_i \frac{\vec{u}_i^T \vec{\psi}_{old}}{\lambda_i} = \sum_{i}^{N} w_i (\vec{u}_i^T \vec{\psi}_{old}) \vec{u}_i \end{aligned} \tag{3.54}$$

So the original i^{th} component of initial function $\vec{\psi}$ has been damped by a factor of w_i, with small eigenvalues components damped more quickly (The weighting function is plotted in Fig. 3.1). Therefore, Tikhonov regularization can be used as a filter to high frequency modes. The damping factor for each Fourier mode is plotted in Fig. 3.6. □

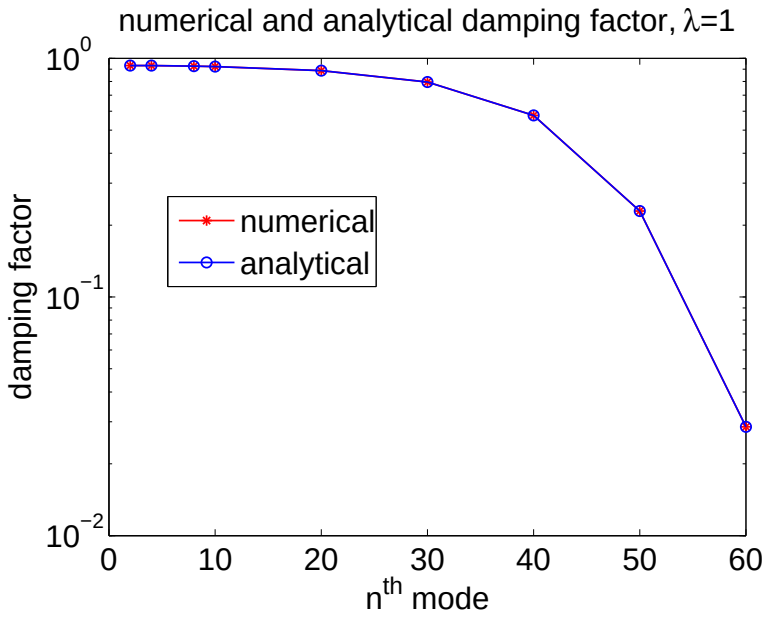

Figure 3.6: Damping factor for different trigonometric modes using standard Tikhonov regularization. Numerical and analytical damping factor are the same.

3.6 Tikhonov regularization for RBFs interpolation on a sphere

In this section, we superpose two modes of Legendre polynomial, i.e., $P_1(\mu) + P_{10}(\mu)$, which gives

$$P_{sum}(\mu) = P_1(\mu) + P_{10}(\mu) \tag{3.55}$$

Here, $\mu = \cos(\theta)\sin(\theta_p) + \sin(\theta)\sin(\theta_p)\cos(\lambda - \lambda_p)$, and (θ_p, λ_p) is the rotated angle. In this part, we let $\theta_p = \pi/2, \lambda_p = 0$. To see the filtering effect of Tikhonov regularization in RBF interpolation, we interpolated function $P_{sum}(\mu)$ using Gaussian Radial Basis Function(GRBF) on a sphere, and then applied the Tikhonov regularization. The following patterns, i.e., Fig. 3.7 and Fig. 3.8 show that, as Tikhonov regularization parameter λ increases, the high mode, i.e., $P_{10}(\mu)$ becomes diminished, and the resulting mode becomes simply the lower mode $P_1(\mu)$. Numerically, we have successfully filtered out higher mode, $P_{10}(\mu)$.

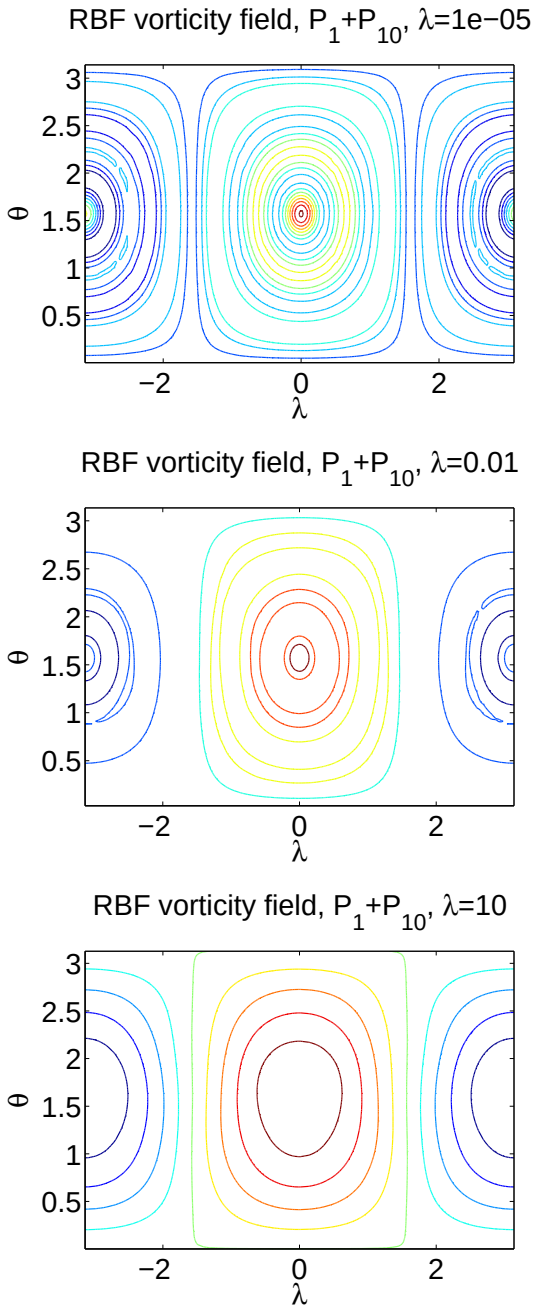

Figure 3.7: Interpolate $P_1(\mu)+P_{10}(\mu)$ using Gaussian RBF on a sphere with $\lambda = 10^{-5}$, 10^{-2} and 10. The number of icosahedral grid points used are $N = 1442$, and Gaussian RBF shape parameter is $\alpha = 1/3$.

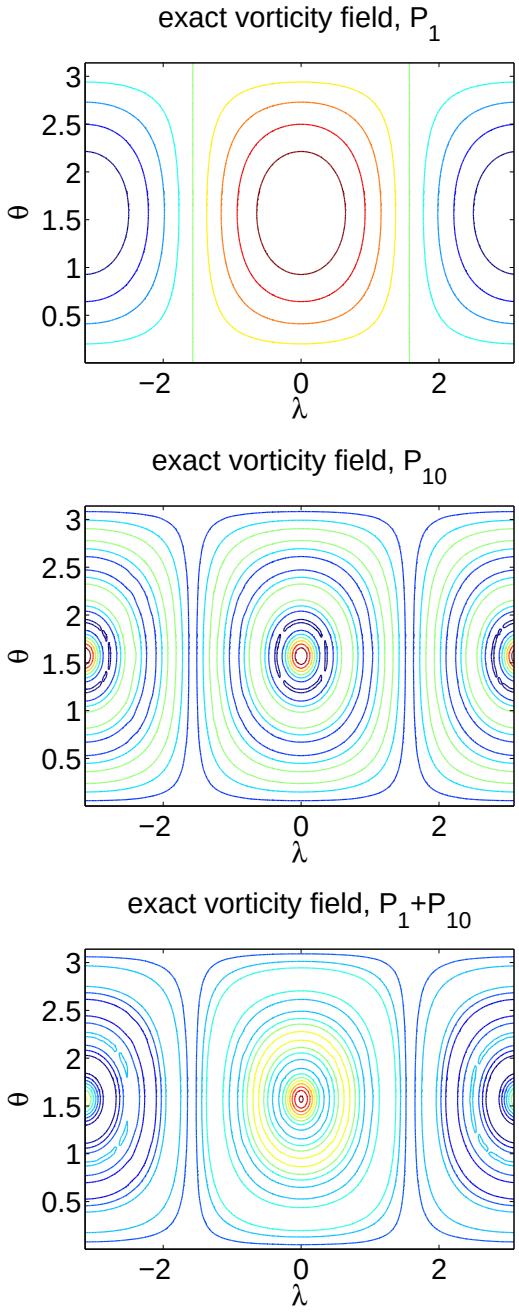

Figure 3.8: Exact $P_1(\mu)$, $P_{10}(\mu)$ and $P_1(\mu) + P_{10}(\mu)$ for comparison. The figures from RBF interpolations with difference Tikhonov regularization parameters are plotted int Fig. 3.7

CHAPTER IV

RBF-Vortex Method for the Barotropic Vorticity Equation

4.1 Introduction

4.1.1 Barotropic Vorticity Equation (BVE)

The Barotropic Vorticity Equation (BVE) on the sphere is the conservation of the absolute vorticity following the motion,

$$\frac{D\zeta_{abs}}{Dt} = 0 \tag{4.1}$$

where the absolute vorticity is

$$\zeta_{abs} = \zeta + 2\Omega\cos(\theta) \tag{4.2}$$

and the total derivative (particle following derivative) is

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \frac{u}{a\sin(\theta)}\frac{\partial}{\partial\lambda} + \frac{v}{a}\frac{\partial}{\partial\theta} \tag{4.3}$$

where, a is the radius of the earth and $\Omega = 2\pi/86400s$ is the angular frequency of the earth's rotation. The velocities are given in terms of the streamfunction ψ as

$$v = a\frac{D\theta}{Dt} = -\frac{1}{\sin(\theta)}\frac{\partial\psi}{\partial\lambda} \tag{4.4}$$

$$u = a\sin(\theta)\frac{D\lambda}{Dt} = \frac{\partial\psi}{\partial\theta} \tag{4.5}$$

Note that v is the southward velocity, which is the opposite of the usual meteorological convention. The streamfunction ψ is determined from the vorticity by solving the Poisson equation.

$$\Delta\psi = \zeta, \qquad \Delta\psi = \frac{1}{a^2}\frac{1}{\sin(\theta)^2}\frac{\partial^2\psi}{\partial\lambda^2} + \frac{1}{a^2}\frac{\partial^2\psi}{\partial\theta^2} + \cot(\theta)\frac{1}{a^2}\frac{\partial\psi}{\partial\theta} \tag{4.6}$$

The vorticity equation can be rewritten as

$$\frac{\partial\zeta}{\partial t} = -\frac{\partial\psi}{\partial\theta}\frac{1}{a\sin(\theta)}\frac{\partial\zeta}{\partial\lambda} + \frac{1}{a\sin(\theta)}\frac{\partial\psi}{\partial\lambda}\frac{\partial\zeta}{\partial\theta} - \frac{2\Omega}{a}\frac{\partial\psi}{\partial\lambda} \tag{4.7}$$

To faciliate the computation, we will nondimensionalize so that the length scale is a, the earth's radius, so that the nondimensional distances are $m/6.36E6$. Similarly, we nondimensionalize time so that the unit vorticity is 2Ω so that one day is 12.56 nondimensional time units and a unit velocity is equivalent to a speed of $924m/s$.

The BVE is a simple mathematical model for the description of planetary scale horizontal motions of the atmosphere. The first numerical weather prediction is performed on BVE using finite difference method (*Charney et al.*, 1950). There is very little work on solving BVE using vortex methods. Recent developments are in (*Wang*, 2010; *Bosler*, 2013; *Bosler et al.*, 2014). In this chapter, an adaptive RBF method is devised to simulate geophysical flows using BVE.

4.1.2 Distance on the sphere

There are two popular ways to measure distance on the sphere. The first is the *Euclidean* distance between two points in ordinary three-dimensional space; the Euclidean distance is the length of a line segment connecting the two points which passes through the interior of the sphere. The second distance metric is the *geodesic* distance, which is the length of the shortest arc along the surface of the sphere which connects the two points.

The geodesic distance is defined for two three-dimensional vectors $\vec{x}$ and $\vec{x}_j$ to be

$$r_G = \arccos(\vec{x} \cdot \vec{x}_j) \tag{4.8}$$

When one of the vectors $\vec{x}$ or $\vec{x}_j$ is the north pole, the geodesic distance is the colatitude θ of the other point.

However, in the RBF literature, the Euclidean distance is more common and is the definition employed here:

$$\begin{aligned} r_E =& \parallel \vec{x} - \vec{x}_j \parallel \\ &= \sqrt{(\vec{x} - \vec{x}_j)^T \cdot (\vec{x} - \vec{x}_j)} \\ &= \sqrt{(x - x_j)^2 + (y - y_j)^2 + (z - z_j)^2} \\ &= \sqrt{2(1 - \sin(\theta)\sin(\theta_j)\cos(\lambda - \lambda_j) - \cos(\theta)\cos(\theta_j))} \end{aligned} \tag{4.9}$$

4.1.3 Coordinate rotation

To faciliate the computation of the RBF on the sphere, the following theorem will be exploited repeatedly in the later sections.

Theorem IV.1. *(Coordinate Rotation) Suppose $f(\lambda, \theta)$ is a radially symmetric func-*

tion on the sphere in the sense that

$$f(\lambda, \theta; \lambda_j, \theta_j) = f(\| \vec{x} - \vec{x}_j \|) \tag{4.10}$$

for point $\vec{x}_j$; thus if the coordinate system is rotated to make $\vec{x}_j$ the new north pole, f is independent of longitude and may be written

$$f(\lambda, \theta; \lambda_j, \theta_j) = f'(\xi)$$

equation for some function f' that depends only on the new colatitude ξ. Then in terms of the original coordinate system,

$$f(\lambda, \theta; \lambda_j, \theta_j) = f'(\xi(\lambda, \theta; \lambda_j, \theta_j)) \tag{4.12}$$

where

$$\cos(\xi) = \cos(\theta)\cos(\theta_j) + \sin(\theta)\sin(\theta_j)\cos(\lambda - \lambda_j) \tag{4.13}$$

4.2 Hyperviscosity

4.2.1 Aliasing instability

In 1D, on a grid of N points, Fourier components $\exp(ikx)$ with $|k| > N/2$ appear as low wavenumbers. The high wavenumber is said to be *aliased* to the low wavenumber. This is because, to resolve a given wave k, we need to sample the wave at least *twice* as frequent as the wave, which means we must place at least *two* grid points within *one* wave distance, in order to resolve the wave(*Boyd*, 1999). However, the computational grid is finite, i.e., h. Therefore, the smallest wave that can be resolved by this h grid is $\lambda = 2h$. Larger wave numbers (correspond to smaller wave lengths via $k = \frac{2\pi}{\lambda}$) get *aliased* to smaller waves via the relation $k_A = k \pm 2mK, m = integer$,

where, $k_A \in [-K, K]$.

In fluid simulation, aliasing can cause numerical instability in the time integration of nonlinear dynamic equations. For example, the quadratically nonlinear term is quite typical in fluid dynamic equations,

$$\begin{aligned} uu_x &= (\sum_{p=-K}^{K} a_p \exp(ipx))(\sum_{q=-K}^{K} iqa_q \exp(iqx)) \\ &= \sum_{k=-2K}^{2K} b_k \exp(ikx) \end{aligned} \tag{4.14}$$

The nonlinear interaction has generated high wavenumbers which will be aliased into wavenumbers on the range $k \in [-K, K]$, on the finite computational grid, which is non-physical. The aliasing reflects energies into larger scales, which is not physically predicted by the nonlinear equations. The simulation will eventually blow up.

The aliasing instability was first observed by *Phillips* (1956) in his model of the earth's general circulation. Phillips tried to stabilized his model by greatly decreasing both the time step and spatial grid size, but the model blew up anyway. He noticed that the warning of the instability was the appearance of $2h$-waves. The reasoning behind this is, in the Fourier space, on a finite grid with spacing h, the largest wave number is $K = \frac{\pi}{h}$, which corresponds to the smallest wave length $\lambda = \frac{2\pi}{K} = 2h$. The aliasing has the largest perturbation at $K = \frac{\pi}{h}$. Therefore the $2h$ wave is the warning of the instability.

To suppress the aliasing instability, Phillips applied a filter, which filtered out completely wavenumbers that are greater than $|k| > \frac{K}{2}$, and keeps the smaller wavenumber intact, i.e.,

$$a_k \rightarrow \begin{cases} a_k & |k| < K/2 \\ 0 & |k| > K/2 \end{cases} \tag{4.15}$$

Orszag(1971) pointed out that Phillips' filtering scheme is wasteful. He was able to

completely eliminate the aliasing by filtering only the upper one third of the spectrum. The two filter schemes fall into the general *filtering coefficients* scheme, where we attach a weighting function to the coefficients of spectral basis. The *filtering coefficients* (*Boyd*, 1996) scheme will be described in the next section.

4.2.2 Filtering coefficients

Suppose the original unfiltered and slowly converging series is

$$u_N(x) = \sum_{k=0}^{N} a_k \phi_k(x) \tag{4.16}$$

where the $\phi_k(x)$ are the basis functions. A smoother and more physical approximation is the filtered partial sum

$$u(x) = \sum_{k=0}^{N} \sigma_k a_k \phi_k(x) \tag{4.17}$$

where σ is symmetric with respect to $x = 0$, that is, $\sigma(-x) = \sigma(x)$ for all x. We call this σ *weighting function*. The weighting function is designed such that, when the frequency of the basis approaches zero $\sigma \to 1$; when the frequency of the basis tends to ∞, $\sigma \to 0$. From Eqn. 4.17, we see this procedure effectively filters out the high frequency waves of the solutions.

We can then rewrite the two schemes as

$$a_k \to \sigma_k a_k, \qquad \begin{cases} \sigma_k = 1 & |k| < K/2 \\ \sigma_k = 0 & |k| > K/2 \end{cases} \qquad \text{Phillips' scheme} \tag{4.18}$$

and

$$a_k \to \sigma_k a_k, \qquad \begin{cases} \sigma_k = 1 & |k| < 2K/3 \\ \sigma_k = 0 & |k| > 2K/3 \end{cases} \qquad \text{Orszag's scheme} \tag{4.19}$$

The weighting functions σ_k can be varied, such that, when $k \to 0$, $\sigma_k = 1$, while $k \to k_{max}$, $\sigma_k = 0$. Another very frequently used weighting function is the Boyd-

Vandeven filter (*Boyd*, 1996). The Boyd-Vandeven filter of order p is

$$\sigma_k = \frac{1}{2} erfc(2\sqrt{p}(|\frac{k}{N}| - \frac{1}{2})\sqrt{-\frac{\log(1-4(|\frac{k}{N}|-\frac{1}{2})^2)}{4(|\frac{k}{N}|-\frac{1}{2})^2}}) \tag{4.20}$$

where,

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-r^2} dr \tag{4.21}$$

However, the general *filtering coefficients* schemes are only valid for filtering orthogonal basis. For a non-orthogonal basis, i.e., Radial Basis Functions, filtering one basis function means filtering the whole spectrum of waves in Fourier space, which is not desirable. Another general filtering scheme for spectral methods is to add a *hyperviscosity* term to the dynamic equations. Boyd(*Boyd*, 1994) showed that the hyperviscosity is equivalent to adding a weighting function to the solution in the spectral basis space.

4.2.3 Hyperviscosity on a plane

The hyperviscosity terms used for spectral methods are $\nu\frac{\partial^2 f}{\partial x^2}, -\nu\frac{\partial^4 f}{\partial x^4}, \nu\frac{\partial^6 f}{\partial x^6}, ...$ One way to understand the hyperviscosity is to assume $f(t,x) = a(t,k)e^{ikx}$, which can be generalized to any functions by Fourier expansion, and take a look at the equation,

$$\frac{\partial f(t,x)}{\partial t} = \nu\frac{\partial^2 f(t,x)}{\partial x^2}, -\nu\frac{\partial^4 f(t,x)}{\partial x^4}, \nu\frac{\partial^6 f(t,x)}{\partial x^6} \tag{4.22}$$

The above equations become, after we substitute $f(t,x)$,

$$\frac{\partial a(t,k)}{\partial t} = -\nu k^2 a(t,k), -\nu k^4 a(t,k), -\nu k^6 a(t,k), \tag{4.23}$$

the solutions to which are,

$$a(t,k) = a(0,k)e^{-\nu k^2 t}, a(0,k)e^{-\nu k^4 t}, a(0,k)e^{-\nu k^6 t} \tag{4.24}$$

The filtering effect of hyperviscosity is obvious by looking at the solutions. Every wave component decays so that $a(t,k) = a(0,k)e^{-\nu k^{2n} t}$. $a(0,k)$ decays faster for larger k and slower for smaller k, which effectively filters higher modes. If we take, $a_k = a(0,k), \sigma_k = e^{-\nu k^{2n} t}$, then the hyperdiffusion falls into the general *filtering coefficient* scheme.

To extend to higher dimensions, for example, on a 2D Cartesian coordinate, we recall that the basis functions $e^{i(kx+jy)}$ in 2D. The corresponding weighting functions are $e^{-\nu(k^{2q}+j^{2q})t}$ where $2q$ is the order of the hyperviscosity.

4.2.4 Hyperviscosity on a sphere

On the surface of a sphere, the Laplacian operator is

$$\nabla^2\psi = \frac{1}{\sin^2(\theta)}\frac{\partial^2\psi}{\partial\lambda^2} + \frac{1}{\sin(\theta)}\frac{\partial}{\partial\theta}(\sin(\theta)\frac{\partial\psi}{\partial\theta}) \tag{4.25}$$

The eigenfunction of the Laplacian operator on the sphere with eigenvalue $-n(n+1)$ is spherical harmonic $Y_n^m(\lambda,\theta)$, that is,

$$\nabla^2 Y_n^m(\lambda,\theta) = -n(n+1)Y_n^m(\lambda,\theta) \tag{4.26}$$

where m can be any integer between $[-n,n]$. The hyperdiffusion equation of order $2q$ on the surface of a sphere can be written as

$$\frac{\partial\psi(t,\lambda,\theta)}{\partial t} = (-1)^{q+1}\nu_{2q}\nabla^{2q}\psi(t,\lambda,\theta) \tag{4.27}$$

If we write $\psi(t,\lambda,\theta) = a_m(t,n)Y_n^m(\lambda,\theta)$, and apply operator ∇^2 q times,

$$\frac{\partial a_m(t,n)}{\partial t} = (-1)^{2q+1}\nu_{2q}[n(n+1)]^q a_m(t,n) \tag{4.28}$$

We can easily derive the solution to Eqn. 4.27

$$\psi(t,\lambda,\theta) = a_m(0,n)e^{-\nu_{2q}[n(n+1)]^q t}Y_n^m(\lambda,\theta) \tag{4.29}$$

Again, what it tells us is that the initial amplitude of the spherical harmonic $a_m(0,n)Y_n^m(\lambda,\theta)$ has been damped by a factor of $e^{-\nu_{2q}[n(n+1)]^q t}$ after a time interval of t. Here the weighting function $\sigma_{n,m} = e^{-\nu_{2q}[n(n+1)]^q t}$, which means all the spherical harmonic modes with the same n have the same weighting function.

4.3 Numerical considerations, hyperviscosity coefficients and energy decay

4.3.1 Numerical considerations

There is one numerical stability issue that needs to be addressed when hyperviscosity is applied in fluid simulation. When we apply an *explicit* time-marching scheme to solve the diffusion equation, the time step Δt is restricted by

$$\Delta t = \frac{c(\Delta x)^{2q}}{\nu_{2q}} \tag{4.30}$$

Eqn. 4.30 can be justified by doing a scale analysis on the following hyperdiffusion equation.

$$\frac{\partial \psi}{\partial t} = (-1)^{q+1}\nu_{2q}\nabla^{2q}\psi \tag{4.31}$$

Scale analysis goes as the following

$$c|\frac{\Delta\psi}{\Delta t}| = |\nu_{2q}\frac{\Delta\psi}{\Delta x^{2q}}| \tag{4.32}$$

We arrive at,

$$\Delta t = \frac{c\Delta x^{2q}}{\nu_{2q}} \tag{4.33}$$

Therefore, as the grid spacing Δx becomes smaller, Δx^{2q} gets smaller, too. And this requires a smaller hyperviscosity coefficients ν_{2q} to keep the simulation at a relative constant time step Δt. The following section will present some techniques to choose the hyperviscosity coefficients ν_{2q}.

4.3.2 Hyperviscosity coefficients

Jablonowski and Williamson (2011) give a very comprehensive review on the choice of hyperviscosity coefficients. In this section, we briefly discuss some of the techniques. From Eqn. 4.30, to keep a relative constant time step Δt for arbitrary order of hyperviscosity $2q$, we let $\nu_{2q} = \gamma\Delta x^{2q}$, then, Δt becomes

$$\Delta t = \frac{c\Delta x^{2q}}{\gamma\Delta x^{2q}} = \frac{c}{\gamma} = C \tag{4.34}$$

On a near-uniform grid, i.e., icosahedral grid, $\Delta x = \sqrt{\frac{4\pi}{N}}$. The assumption behind this equation is the N grid points cover the surface of the sphere with radius $a = 1$ and area $4\pi a^2 = 4\pi$. To see the scale-selectivity of hyperviscosity, we can set $\nu_{2q} = \gamma(\frac{1}{M(M+1)})^q$ on a spherical domain; and $\nu_{2q} = \gamma(\frac{1}{K^2})^q$ for a Cartesian coordinate domain, where M and K are the damping scale of the hyperviscosity. Fig. 4.1 shows the weighting functions of different orders of hyperviscosity. Clearly all the curves cross at the damping scale wavenumber M and K respectively. What we see here is as the order of hyperviscosity increases, the weighting functions become

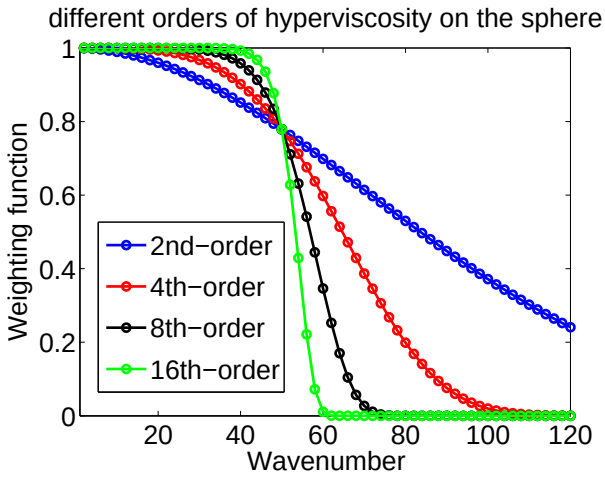

Figure 4.1: Weighting functions for different orders(2^{nd}-order, 4^{th}-order, 8^{th}-order and 16^{th}-order) of hyperviscosity on the sphere. In this plot, the damping scale $M = 50$ is used.

steeper. Therefore, wavenumbers smaller than M(or K) are preserved relatively well and wavenumbers larger than M(or K) are dissipated rapidly.

4.3.3 Energy decay

While hyperviscosity stabilizes the simulation of nonlinear dynamics, it does degrade the accuracy of the simulation. Hyperviscosity causes the energy to decay at a rate which depends on the time-marching schemes we employ and can be studied numerically too. In this part, I do an analysis of the energy decay of the hyperviscosity equation on the sphere using a spherical harmonic basis. Suppose we have an initial profile $\psi(0, \lambda, \theta) = a(0, n)\phi_n(\lambda, \theta)$, where $\phi_n(\lambda, \theta)$ is the spherical harmonic or combination of spherical harmonic of order n, and Forward Euler time-marching scheme is applied to the hyperdiffusion equation 4.31. We write $\psi(t, \lambda, \theta) = a(t, n)\phi_n(\lambda, \theta)$, then

$$a(\Delta t, n) = a(0, n) - \Delta t \nu_{2q}[n(n+1)]^q a(0, n) = \{1 - \Delta t \nu_{2q}[n(n+1)]^q\} a(0, n) \quad (4.35)$$

The damping factor $s_{\Delta t} = 1 - \Delta t \nu_{2q}[n(n+1)]^q$ is the first order Taylor expansion of the weighting function $e^{-\nu_{2q}[n(n+1)]^q \Delta t}$.

By time t, the initial profile has been damped N_t times

$$s_t = s_{\Delta t}^N = (1-\nu_{2q}[n(n+1)]^q \Delta t)^{N_t} \approx 1-\nu_{2q}[n(n+1)]^q N_t \Delta t = 1-\nu_{2q}[n(n+1)]^q t \quad (4.36)$$

In the above equation, we have used $N_t \Delta t = t$ and $(1-x)^N \approx 1 - Nx$ for $|x| << 1$. This damping factor is the first order approximation of $e^{-\nu_{2q}[n(n+1)]^q N_t \Delta t} = e^{-\nu_{2q} n(n+1)^q t} \approx 1-\nu_{2q}[n(n+1)]^q t$, where we have used $\lim_{N_t \to \infty}\{1-\nu_{2q}[n(n+1)]^q \frac{t}{N_t}\}^{N_t} = e^{-\nu_{2q}[n(n+1)]^q t}$. Using first order approximation, we can easily see the linear relationship between relative error and time. Therefore, at time t, the relative error is predicted to be

$$E_t = |\frac{a(0,n)\phi_n(\lambda,\theta) - (1-\nu_{2q}[n(n+1)]^q t)a(0,n)\phi_n(\lambda,\theta)}{a(0,n)\phi_n(\lambda,\theta)}| = \nu_{2q}[n(n+1)]^q t \quad (4.37)$$

From Eqn. 4.37, the relative error of a spherical harmonic on the sphere after a period of time t is linearly proportional to $\nu_{2q}[n(n+1)]^q$. This has been verified by the Rossby-Haurwitz wave test in Fig. 4.5.

4.4 Discretization of BVE with hyperviscosity using Gaussian RBF

4.4.1 Discretization of BVE

We repeat the vorticity equation here,

$$\frac{\partial \zeta}{\partial t} = -\frac{\partial \psi}{\partial \theta}\frac{1}{a\sin(\theta)}\frac{\partial \zeta}{\partial \lambda} + \frac{1}{a\sin(\theta)}\frac{\partial \psi}{\partial \lambda}\frac{\partial \zeta}{\partial \theta} - \frac{2\Omega}{a}\frac{\partial \psi}{\partial \lambda} \quad (4.38)$$

where, the relation between the relative vorticity ζ and streamfunction ψ satisfies,

$$\Delta\psi = \zeta \tag{4.39}$$

ζ and ψ can be interpolated using Gaussian Radial Basis Functions,

$$\psi(\lambda,\theta) = \sum_{j=1}^{N} a_j^{\psi}\phi_j(\lambda,\theta) \tag{4.40}$$

$$\zeta(\lambda,\theta) = \sum_{j=1}^{N} a_j^{\zeta}\phi_j(\lambda,\theta) \tag{4.41}$$

where $\phi_i(\lambda,\theta)$ is the Gaussian RBF corresponding to the center located at (λ_j,θ_j). First, assume that the RBF center is located at the north pole, and the corresponding GA RBF is

$$\phi_0(\xi;\epsilon) = \exp(-2\epsilon^2(1-\cos(\xi))) \tag{4.42}$$

Apply the Poisson operator, we have,

$$\Delta\phi_0(\xi;\epsilon) = 4\epsilon^2\{\epsilon^2 - \cos(\xi) - \epsilon^2\cos(\xi)^2\}\exp(-2\epsilon^2(1-\cos(\xi)) \tag{4.43}$$

Now, rotate the center from the north pole to the location (λ_j,θ_j), since the Poisson operator is invariant under rotation on the sphere, the coordinate rotation theorem implies

$$\Delta\phi_j(\lambda,\theta) = 4\epsilon^2\{\epsilon^2 - \Xi_j(\lambda,\theta) - \epsilon^2\Xi_j^2(\lambda,\theta)\}\exp(-2\epsilon^2(1-\Xi_j(\lambda,\theta))) \tag{4.44}$$

where,

$$\Xi_j(\lambda,\theta) = \cos(\theta)\cos(\theta_j) + \sin(\theta)\sin(\theta_j)\cos(\lambda-\lambda_j) \tag{4.45}$$

Therefore,

$$\Delta\psi(\lambda,\theta) = \Delta(\sum_{j=1}^{N} a_j^{\psi}\phi_j(\lambda,\theta)) = \sum_{j=1}^{N} a_j^{\psi}\Delta\phi_j(\lambda,\theta) \tag{4.46}$$

Usually, we take the collocation points the same as the RBF centers. Then Eqn. 4.40 and Eqn. 4.46 becomes,

$$\begin{aligned} \vec{\vec{D}}^0\vec{a}^{\psi} &= \vec{\psi} \\ \vec{\vec{D}}^0\vec{a}^{\zeta} &= \vec{\zeta} \\ \vec{\vec{P}}\vec{a}^{\psi} &= \vec{\zeta} \end{aligned} \tag{4.47}$$

and

$$\vec{a}^{\zeta} = (\vec{\vec{D}}^0)^{-1}\vec{\zeta} = \vec{\vec{Q}}\vec{\zeta} \tag{4.48}$$

$$\vec{a}^{\psi} = (\vec{\vec{P}})^{-1}\vec{\zeta} = \vec{\vec{G}}\vec{\zeta} \tag{4.49}$$

where, the interpolation matrices are given by,

$$\vec{\vec{D}}^0_{ij} = \phi_j(\lambda_i,\theta_i) = \exp\{-2\epsilon^2(1-\sin(\theta_i)\sin(\theta_j)\cos(\lambda_i-\lambda_j)-\cos(\theta_i)\cos(\theta_j))\} \tag{4.50}$$

$$\vec{\vec{P}}_{ij} = 4\epsilon^2\{\epsilon^2 - \Xi_j(\lambda_i,\theta_i) - \epsilon^2\Xi_j^2(\lambda_i,\theta_i)\}\exp(-2\epsilon^2(1-\Xi_j(\lambda_i,\theta_i))) \tag{4.51}$$

and,

$$\Xi_j(\lambda_i,\theta_i) = \cos(\theta_i)\cos(\theta_j) + \sin(\theta_i)\sin(\theta_j)\cos(\lambda_i-\lambda_j) \tag{4.52}$$

To evaluate the derivative terms in the Eqn. 4.38, we have,

$$\frac{\partial \zeta}{\partial \lambda} = \sum_{j=1}^{N} a_j^{\zeta} \frac{\partial \phi_j(\lambda, \theta)}{\partial \lambda} \tag{4.53}$$

$$\frac{\partial \zeta}{\partial \theta} = \sum_{j=1}^{N} a_j^{\zeta} \frac{\partial \phi_j(\lambda, \theta)}{\partial \theta} \tag{4.54}$$

where,

$$\begin{aligned} \frac{\partial \phi_j(\lambda, \theta)}{\partial \lambda} &= \exp\{-2\epsilon^2(1 - \sin(\theta)\sin(\theta_j)\cos(\lambda - \lambda_j) - \cos(\theta)\cos(\theta_j))\} \\ &\quad (-2\epsilon^2)\sin(\theta)\sin(\theta_j)\sin(\lambda - \lambda_j) \end{aligned} \tag{4.55}$$

$$\begin{aligned} \frac{\partial \phi_j(\lambda, \theta)}{\partial \theta} &= \exp\{-2\epsilon^2(1 - \sin(\theta)\sin(\theta_j)\cos(\lambda - \lambda_j)) - \cos(\theta)\cos(\theta_j)\} \\ &\quad (-2\epsilon^2)\{-\cos(\theta)\sin(\theta_j)\cos(\lambda - \lambda_j) + \sin(\theta)\cos(\theta_j)\} \end{aligned} \tag{4.56}$$

Again, we take the collocation points the same as the RBF centers, then, we can rewrite Eqn. 4.53 as,

$$\begin{aligned} \frac{\partial \zeta}{\partial \lambda} &\rightarrow \vec{\vec{D}}^{\lambda} \vec{a}^{\zeta} = \vec{\vec{D}}^{\lambda} (\vec{\vec{D}}^{0})^{-1} \vec{\zeta} = \vec{\vec{D}}^{\lambda} \vec{\vec{Q}} \vec{\zeta} \\ \frac{\partial \zeta}{\partial \theta} &\rightarrow \vec{\vec{D}}^{\theta} \vec{a}^{\zeta} = \vec{\vec{D}}^{\theta} (\vec{\vec{D}}^{0})^{-1} \vec{\zeta} = \vec{\vec{D}}^{\theta} \vec{\vec{Q}} \vec{\zeta} \end{aligned} \tag{4.57}$$

Here, matrices $\vec{\vec{D}}^{\lambda}$ and $\vec{\vec{D}}^{\theta}$ are given by,

$$\begin{aligned} \vec{\vec{D}}^{\lambda}_{ij} &= \frac{\partial \phi_j(\lambda_i, \theta_i)}{\partial \lambda} = \vec{\vec{D}}^{0}_{ij}\{-2\epsilon^2 \sin(\theta_i)\sin(\theta_j)\sin(\lambda_i - \lambda_j)\} \\ \vec{\vec{D}}^{\theta}_{ij} &= \frac{\partial \phi_j(\lambda_i, \theta_i)}{\partial \theta} = \vec{\vec{D}}^{0}_{ij}\{\cos(\theta_i)\sin(\theta_j)\cos(\lambda_i - \lambda_j) - \sin(\theta_i)\cos(\theta_j)\} \end{aligned} \tag{4.58}$$

Similarly, we have,

$$\begin{aligned} \frac{\partial \psi}{\partial \lambda} &\rightarrow \vec{\vec{D}}^{\lambda} \vec{a}^{\psi} = \vec{\vec{D}}^{\lambda} \vec{\vec{G}} \vec{\zeta} \\ \frac{\partial \psi}{\partial \theta} &\rightarrow \vec{\vec{D}}^{\theta} \vec{a}^{\psi} = \vec{\vec{D}}^{\theta} \vec{\vec{G}} \vec{\zeta} \end{aligned} \tag{4.59}$$

Then it immediately follows that the discrete velocities are

$$\begin{aligned} \vec{u} &= \vec{\vec{D}}^{\theta} \vec{\vec{G}} \vec{\zeta} \\ \vec{v} &= -\frac{1}{\sin(\vec{\theta})} .* \vec{\vec{D}}^{\lambda} \vec{\vec{G}} \vec{\zeta} \end{aligned} \tag{4.60}$$

Here, the MATLAB *element-wise multiplication* notation $.*$ is used.

4.4.2 Discretization of hyperviscosity term

We applied the 4^{th}-order diffusion operator to a Gaussian RBF with center located at the north pole,

$$\Delta^2 \phi_0(\xi; \epsilon) = -\Delta^2 (e^{-2\epsilon^2(1-\cos(\xi))}) = -(8\epsilon^2 e^{-2\epsilon^2(1-\cos(\xi))}) f1(\xi) \tag{4.61}$$

where $f1(\xi) = \{2\epsilon^6 \cos(\xi)^4 + 8\epsilon^4 \cos(\xi)^3 + (7\epsilon^2 - 4\epsilon^6)\cos(\xi)^2 + (1 - 8\epsilon^4)\cos(\xi) - 3\epsilon^2 + 2\epsilon^6\}$. Again, due to the invariance of operator $-\Delta^2$ on the sphere, we can apply the rotation theorem so that the north pole is rotated to (λ_j, θ_j), and $\cos(\xi)$ becomes

$$\cos(\xi) = \Xi_j(\lambda, \theta) = \cos(\theta)\cos(\theta_j) + \sin(\theta)\sin(\theta_j)\cos(\lambda - \lambda_j) \tag{4.62}$$

Therefore,

$$-\Delta^2 \phi_j(\lambda, \theta) = -\Delta^2 (e^{-2\epsilon^2(1-\Xi_j)}) = -(8\epsilon^2 e^{-2\epsilon^2(1-\Xi_j)}) f2(\Xi_j) \tag{4.63}$$

where $\text{f2}(\Xi_j) = \{2\epsilon^6 \Xi_j^4 + 8\epsilon^4 \Xi_j^3 + (7\epsilon^2 - 4\epsilon^6)\Xi_j^2 + (1 - 8\epsilon^4)\Xi_j - 3\epsilon^2 + 2\epsilon^6\}$.

$$
\begin{aligned}
-\Delta^2\zeta(\lambda,\theta) &= -\Delta^2\sum_{j=1}^{N} a_\zeta^j\phi_j(\lambda,\theta) = \sum_{j=1}^{N} a_\zeta^j(-\Delta^2\phi_j(\lambda,\theta)) \\
&= -\sum_{j=1}^{N} a_\zeta^j(8\epsilon^2 e^{-2\epsilon^2(1-\Xi_j)})f3(\Xi_j)
\end{aligned}
\tag{4.64}
$$

where $f3(\Xi_j) = \{2\epsilon^6\Xi_j^4 + 8\epsilon^4\Xi_j^3 + (7\epsilon^2 - 4\epsilon^6)\Xi_j^2 + (1 - 8\epsilon^4)\Xi_j - 3\epsilon^2 + 2\epsilon\}$. Choosing the collocation points to be the same as the RBF centers, Eqn. 4.64 can be written as

$$
-\Delta^2\zeta(\lambda,\theta) \rightarrow \vec{\vec{D}}^4\vec{a}_\zeta = \vec{\vec{D}}^4\vec{\vec{Q}}\vec{\zeta} \tag{4.65}
$$

where

$$
\vec{\vec{D}}^4_{ij} = -(8\epsilon^2 e^{-2\epsilon^2(1-\Xi_{ij})})\{2\epsilon^6\Xi_{ij}^4 + 8\epsilon^4\Xi_{ij}^3 + (7\epsilon^2 - 4\epsilon^6)\Xi_{ij}^2 + (1-8\epsilon^4)\Xi_{ij} - 3\epsilon^2 + 2\epsilon^6\} \tag{4.66}
$$

with

$$
\Xi_{ij} = \cos(\theta_i)\cos(\theta_j) + \sin(\theta_i)\sin(\theta_j)\cos(\lambda_i - \lambda_j) \tag{4.67}
$$

4.5 Simulating BVE with hyperviscosity

In this section, we describe two versions of the algorithm to simulate the Barotropic vorticity equation with hyperviscosity: *Eulerian* and *Lagrangian*.

4.5.1 Eulerian version

In Eulerian coordinates, we fix the interpolation points in time. The Eulerian version of the Barotropic Vorticity Equation immediately follows,

$$
\frac{\partial\zeta}{\partial t} = -\frac{\partial\psi}{\partial\theta}\frac{1}{a\sin(\theta)}\frac{\partial\zeta}{\partial\lambda} + \frac{1}{a\sin(\theta)}\frac{\partial\psi}{\partial\lambda}\frac{\partial\zeta}{\partial\theta} - \frac{2\Omega}{a}\frac{\partial\psi}{\partial\lambda} - \nu\Delta^2\zeta \tag{4.68}
$$

The discretized form of Eqn. 4.68 is

$$\frac{\partial \vec{\zeta}}{\partial t} = -\vec{u}. * \frac{1}{\sin(\vec{\theta})}. * (\vec{\vec{D}}^{\lambda}\vec{\vec{Q}}\vec{\zeta}) - \vec{v}. * \vec{\vec{D}}^{\theta}\vec{\vec{Q}}\vec{\zeta} - \vec{\vec{D}}^{\lambda}\vec{\vec{G}}\vec{\zeta} + \nu\vec{\vec{D}}^{4}\vec{\vec{Q}}\vec{\zeta} \tag{4.69}$$

Of course, we can solve Eqn. 4.69 directly using 4^{th}-order Runge-Kutta time-marching scheme. However, since hyperviscosity is not physical, we can do it separately using cheaper schemes, i.e., Forward Euler method. That is, we solve

$$\frac{\partial \vec{\zeta}^{*}}{\partial t} = -\vec{u}. * \frac{1}{\sin(\vec{\theta})}. * (\vec{\vec{D}}^{\lambda}\vec{\vec{Q}}\vec{\zeta}^{n}) - \vec{v}. * \vec{\vec{D}}^{\theta}\vec{\vec{Q}}\vec{\zeta}^{n} - \vec{\vec{D}}^{\lambda}\vec{\vec{G}}\vec{\zeta}^{n} \tag{4.70}$$

using $RK4$ scheme. Before moving to the next time step, we do a hyperdiffusion on $\vec{\zeta}^{n+1}$.

$$\vec{\zeta}^{n+1} = \vec{\zeta}^{*} + \Delta t\nu\vec{\vec{D}}^{4}\vec{\vec{Q}}\vec{\zeta}^{*} \tag{4.71}$$

4.5.2 Lagrangian version

The Barotropic Vorticity Equation

$$\frac{D(\zeta + \cos(\theta))}{Dt} = 0 \tag{4.72}$$

tells us that as a fluid element moves with the flow, the absolute vorticity $\zeta + \cos(\theta)$ on the element is conserved. The Lagrangian approach is: we initialize a set of vortices with strength $\vec{\zeta}$ on the sphere, where the locations of the vortices $(\vec{\lambda}, \vec{\theta})$ are the RBF centers, too. The $\vec{\zeta}$ is used to compute the velocity field $\vec{u}$ and $\vec{v}$. The absolute vorticity on the vortex particles are $\vec{\zeta}_{abs} = \vec{\zeta} + \cos(\vec{\theta})$. The vortices are advected by the velocity field, i.e., $\vec{u}$ and $\vec{v}$, using 4^{th}-order Runge Kutta scheme to new locations, i.e., $(\vec{\lambda}_{new}, \vec{\theta}_{new})$. Due to the conservation of the absolute vorticity $\vec{\zeta}_{abs}$ and the change of Coriolis forces on the vortex particles, the new relative vorticity becomes $\vec{\zeta}_{new} = \vec{\zeta}_{abs} - \cos(\vec{\theta}_{new})$, which will induce new velocity field, $\vec{u}_{new}$ and $\vec{v}_{new}$.

Fig. 4.2 shows the flowchart for simulating the BVE using RBF-Vortex method with hyperviscosity.

The hyperdiffusion process occurs before we move on to next time step. That is, we do a standard hyperdiffusion using the vortex configuration we have now. For simulating the Navier-Stokes Equations, Chorin(*Chorin*, 1973) proposed that the diffusion of vorticity can be simulated via the Brownian motion of vortex elements. A widely used method to generate a Brownian motion (Gaussian random numbers) in Monte Carlo simulations is called *Box-Muller* method(*Ahrens et al.*, 2013). However, this scheme is of low accuracy and is applied to slightly viscous flow only.

$$\frac{\partial \zeta(\lambda,\theta)}{\partial t} = -\nu\Delta^2\zeta(\lambda,\theta) \tag{4.73}$$

Usually, we can just use a lower order time-marching scheme for Eqn. 4.73, i.e., Forward Euler, which is cheap and easy to program. That is,

$$\zeta^{n+1} = \zeta^* - \nu\Delta t\Delta^2\zeta^* \tag{4.74}$$

4.6 Numerical experiments

4.6.1 Rossby-Haurwitz waves

4.6.1.1 Rossby-Haurwitz waves test

Theorem IV.2. *: Let $Y_n^m(\lambda,\theta)$ denote the spherical harmonics. If the initial condition for the streamfunction is a spherical harmonic $Y_n^m(\lambda,\theta)$ or a superposition of spherical harmonics with arbitrary amplitudes a_m, but the same subscript(degree) n,*

$$\Psi(\lambda,\theta,0) = H(\lambda,\theta) = \sum_{m=-n}^{m} a_m Y_n^m(\lambda,\theta) \tag{4.75}$$

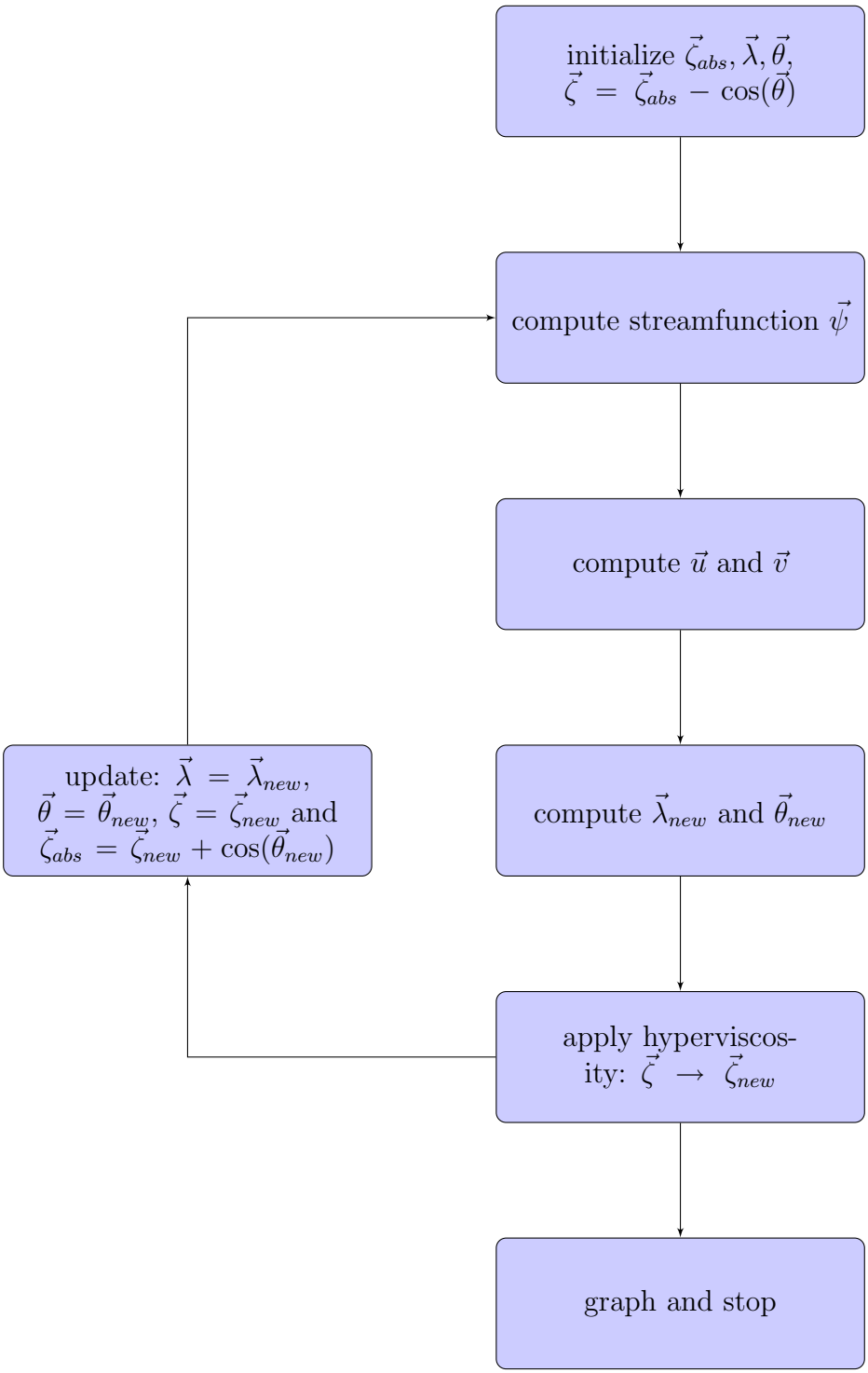

Figure 4.2: Flowchart: Simulating the Barotropic Vorticity Equation using RBF-Vortex Method

this is an exact, steadily-translating finite amplitude solution to the Barotropic Vorticity Equation. The different zonal wavenumber components propagate with the longitudinal phase speed

$$\begin{aligned} c &= -\frac{2\Omega}{a}\frac{1}{n(n+1)} \\ \Psi(\lambda,\theta,t) &= H(\lambda - ct,\theta) \end{aligned} \tag{4.76}$$

The sum of the advective terms in the vorticity equation is zero for all time:

$$N = -\frac{\partial\Psi}{\partial\theta}\frac{1}{a\sin(\theta)}\frac{\partial\zeta}{\partial\lambda} + \frac{1}{\sin(\theta)}\frac{\partial\Psi}{\partial\lambda}\frac{1}{a}\frac{\partial\zeta}{\partial\theta} = 0 \tag{4.77}$$

Proof. Because Ψ is an eigenfunction of the Laplacian operator, $\zeta = -n(n+1)\Psi$ for all times. The vorticity equation can be written in terms of Ψ as

$$-n(n+1)\frac{\partial\Psi}{\partial t} = n(n+1)\frac{\partial\Psi}{\partial\theta}\frac{1}{a\sin(\theta)}\frac{\partial\Psi}{\partial\lambda} - n(n+1)\frac{1}{\sin(\theta)}\frac{\partial\Psi}{\partial\lambda}\frac{1}{a}\frac{\partial\Psi}{\partial\theta} - \frac{2\Omega}{a}\frac{\partial\Psi}{\partial\lambda} \tag{4.78}$$

It is then obvious by inspection that the two nonlinear terms, the first two to the right of the equal sign, cancel identically when Ψ and ζ are proportional to one another. The vorticity equation collapses to the linear wave equation

$$\frac{\partial\Psi}{\partial t} - \frac{1}{n(n+1)}\frac{2\Omega}{a}\frac{\partial\Psi}{\partial\lambda} = 0 \tag{4.79}$$

We can easily check that the solution to the above equation is $\Psi = H(\lambda - ct)$.

□

4.6.1.2 Legendre Polynomial Waves

Consider two unit vectors $\vec{x}$ and $\vec{y}$, respectively, with spherical coordinates (λ, θ) and (λ', θ'). The addition theorem states

$$P_n(\vec{x}, \vec{y}) = \frac{4\pi}{2n+1} \sum_{m=-n}^{n} Y_{nm}^*(\theta', \lambda') Y_{nm}(\theta, \lambda) \tag{4.80}$$

For a fixed rotation angle $\theta' = \theta_p$, $\lambda' = \lambda_p$, $Y_{nm}^*(\theta_p, \lambda_p)$ is a function of m only. We can easily see that the Legendre polynomial is simply a combination of the spherical harmonics with constant coefficients, the same n and m ranging from $-n$ to n. Then, Theorem IV.2 applies. Thus, a steadily-translating Rossby-Haurwitz wave can be generated by a Legendre polynomial. Given the non-dimensionalization condition we have assumed, the phase velocity is

$$c = -\frac{1}{n(n+1)} \tag{4.81}$$

$P_n(\vec{x} \cdot \vec{y}) = P_n(\cos(\xi))$, where ξ is the angle between $\vec{x}$ and $\vec{y}$, which satisfies,

$$\cos(\xi) = \cos(\theta)\cos(\theta_p) + \sin(\theta)\sin(\theta_p)\cos(\lambda - \lambda_p) \tag{4.82}$$

The Legendre polynomial wave $P_n(\cos(\xi))$ becomes

$$P_n(\cos(\xi)) = P_n(\cos(\theta)\cos(\theta_p) + \sin(\theta)\sin(\theta_p)\cos(\lambda - \lambda_p)) \tag{4.83}$$

For example, the Legendre polynomial

$$P_2(\mu) = (3/2)\mu^2 - 1/2 \tag{4.84}$$

with $n = 2, c = -\frac{1}{n(n+1)} = -1/6$ gives the traveling wave

$$\psi(\lambda - ct, \theta) = (3/2)(\cos(\theta)\cos(\theta_p) + \sin(\theta)\sin(\theta_p)\cos(\lambda - ct - \lambda_p))^2 - 1/2 \quad (4.85)$$

Similarly, the Legendre polynomial of tenth degree is

$$P_{10}(\mu) = \frac{1}{256}(46189\mu^{10} - 109395\mu^8 + 90090\mu^6 - 30030\mu^4 + 3465\mu^2 - 63) \quad (4.86)$$

. The corresponding traveling wave is

$$P_{10}(\mu(t)) = \frac{1}{256}(46189\mu(t)^{10} - 109395\mu(t)^8 + 90090\mu(t)^6 - 30030\mu(t)^4 + 3465\mu(t)^2 - 63) \quad (4.87)$$

where $\mu(t) = \cos(\theta)\cos(\theta_p) + \sin(\theta)\sin(\theta_p)\cos(\lambda - ct - \lambda_p)$. This wave is travelling with $c = -\frac{1}{110}$.

In Fig. 4.3, we simulate the second degree Legendre polynomial wave using Lagrangian and Eulerian approaches. Small scales develop in Eulerian coordinate due to the small hyperdiffusion coefficient we used; however, as we increase the hyperdiffusion coefficient 5 times, the solution becomes smooth. In Lagrangian coordinate, even for small $\nu = 2 \times 10^{-6}$, the numerical result is smooth. In the Lagrangian coordinate, the particles follow the flow and the numerical error of approximating $\frac{Du}{Dt} = \frac{\partial u}{\partial t} + u \cdot u_x$ comes from the discretization of the velocities only. In the Eulerian coordinate, the error for approximating the advection term $u \cdot u_x$ comes from two parts: the discretization of u and the approximation of u_x, which introduces more error in many cases. In the Rossby-Haurwitz wave test, we use $N = 1442$ grid points, that is, we divided each icosahedral edge into 12 equal intervals on the sphere.

Fig. 4.4 shows two smooth looking graphs using the two approaches. The upper panel shows relative error, streamfunction and relative vorticity using Eulerian coordinate with $\nu = 10^{-5}$; and the lower panel shows the corresponding Lagrangian case

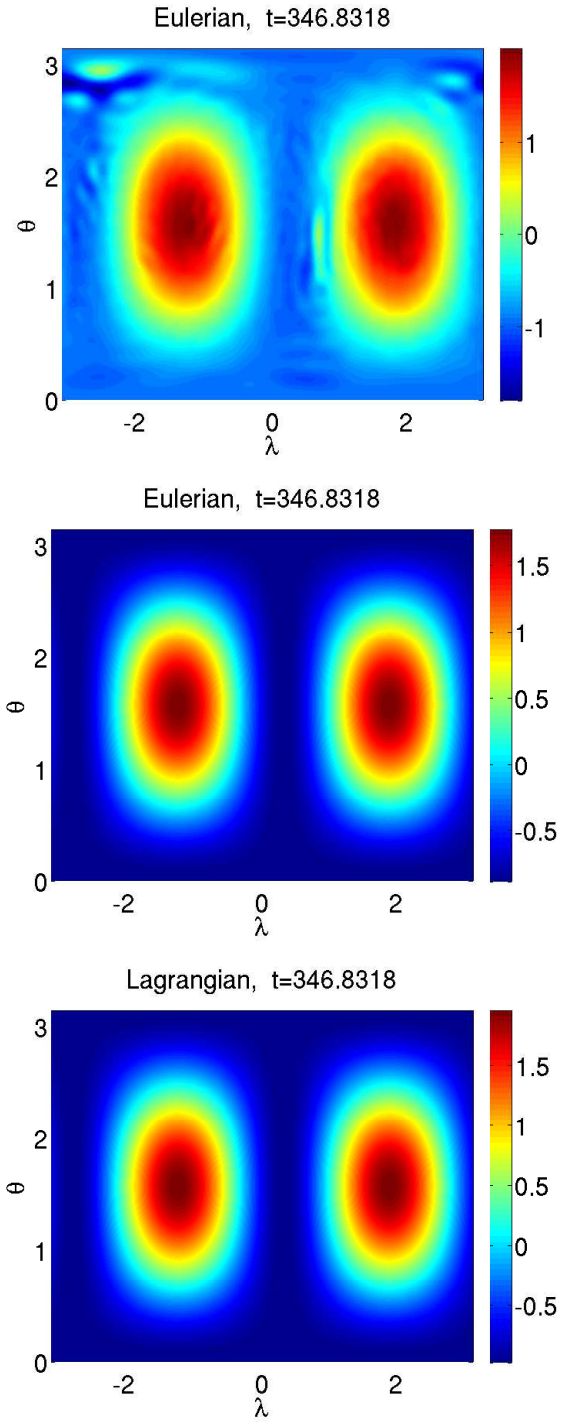

Figure 4.3: Rossby-Haurwitz wave test: using Eulerian and Lagrangian approaches. We use icosahedral grid with 1442 points, RBF shape parameter $\alpha = 1/3$ and 4^{th}-order hyperviscosity. From up to down the three graphs are: Eulerian coordinate with $\nu = 2 \times 10^{-6}$, Eulerian coordinate with $\nu = 10^{-5}$ and Lagrangian coordinate with $\nu = 2 \times 10^{-6}$.

with $\nu = 2 \times 10^{-6}$. The relative error of Eulerian case is about 5 times larger than the Lagrangian, which may result from the fact that we are using a larger ν for the Eulerian. We track the error of the simulation, and the error is plotted in Fig. 4.3. The relative errors can be justfied by the analysis in Sec. 4.3.3. The slope of the relative error using Forward Euler method at time t is predicted to be

$$E_t = \nu_{2q}[n(n+1)]^q t \tag{4.88}$$

In our simulation, the Rossby-Haurwitz wave we are simulating is the second degree Legendre polynomial with $\nu = 2 \times 10^{-6}$ and $q = 2$; therefore, by Eqn. 4.88, we have the time-dependent relative error is

$$E_t = 2 \times 10^{-6} \times [2(2+1)]^2 t = 7.2 \times 10^{-5} t \tag{4.89}$$

The theoretically predicted slope of relative error 7.2×10^{-5} agrees well with the numerical slope of the curve of relative error 7.1341×10^{-5} in Fig. 4.5. This means that the RBF-Vortex method is accurate, and the total error is dominated by the artificial dissipation. Similarly we can predict the ratio of relative errors using different values of hyperviscosity coefficients, i.e., 10^{-5} for Eulerian and 2×10^{-6} for Lagrangian

$$ratio = \frac{E_t^{eul}}{E_t^{lag}} = \frac{10^{-5} \times [2(2+1)]^2 t}{2 \times 10^{-6} \times [2(2+1)]^2 t} = 5 \tag{4.90}$$

From Fig. 4.4, the Eulerian relative error at T(final time) is 0.12689, while the final Lagrangian relative error is 0.026832, the ratio follows

$$ratio = \frac{0.12689}{0.026832} = 4.7291 \tag{4.91}$$

Again, the theoretically calculated and numerically computed error ratios are very

close.

Fig. 4.6 shows the relative error with zero hyperviscosity after a short time interval. There are two reasons that we consider a short time interval and zero hyperviscosity in this case. First, to quantify the merits of the two approaches, we need to eliminate the effect of hyperviscosity and also aliasing error; a longer time period will accumulate more aliasing error if there is no hyperviscosity. Second, there is one obvious advantage for vortex method, that is, the vortex distortion after a longer time has been shown to be a disadvantage of the vortex methods, and by measuring such a short time, we are able to eliminate the effect of the vortex distortion. Fig. 4.6 shows that at $N(<\approx 100)$, the Eulerian coordinate is superior; while as we increase the number of grid points, the advantage of Lagrangian approach takes over. For $N(<\approx 100)$, our result shows that the Lagrangian RBF-Vortex model is of about 5^{th}-order accuracy.

Fig. 4.7 gives the time per 4^{th}-order Runge-Kutta in our Lagrangian and Eulerian models. The time scales as $N^{2.2}$ for Lagrangian model. The Eulerian case is plotted for comparison.

4.6.2 A single vortex on a rotating sphere

In this section, we simulated the dynamics of a Gaussian vortex on a rotating sphere using BVE. The same kind of numerical experiment via BVE has been studied by Levy, Nair and Tufo(*Levy et al.*, 2009). In their paper in 2009, they solved the BVE equation on the β-plane (the coriolis force is approximated by Taylor expansion to the first order, that is, a plane tangent to the spherical surface), using a discontinuous Galerkin method. Our numerical experiments show that a vortex on a rotating sphere tends to shift to the northwest which agrees with Levy's work. The single vortex perturbs the atmosphere. The perturbation propagates westward as Rossby waves. The reasoning behind this is: the rotating vortex drags the surrounding initially

stationary fluid elements, resulting a displacement, either to the north or to the south; since we are solving an absolute-vorticity-conserving BVE, the fluid to the north will experience a larger Coriolis force and therefore obtain a negative relative vorticity (a clockwise vorticity), which turns the fluid to the south; the fluid to the south would undergo a smaller Coriolis force and therefore acquires a positive relative vorticity to compensate, indicating a counter-clockwise vorticity. This perturbation propagates westward.

4.6.3 Vortex merger on a rotating sphere

Two vortices on a rotating experiment has been performed by (*Levy et al.*, 2009; *Shin et al.*, 2006), either on a β-plane or f-plane. The general numerical results in these papers are: the merger or separation between two vorticies is largely determined by the sign of the initial relative vorticity and relative distance between the two vortices. When the initial relative vorticity between two vortices is negative, the two vortices separate. However, the initial positive relative vorticity between the two vortices does not guarantee their merger. Shin's research (*Shin et al.*, 2006) suggests, only when the distance between the two vorticies is smaller than a certain critical separation distance, can the merger occur. Numerical experiments suggested that the critical separation distance of binary vortices is slightly smaller than twice the radius at which the relative vorticity of one vortex becomes zero (*Shin et al.*, 2006). The existence of the *critical separation distance* is obvious: we do not expect a typhoon in the East China Sea to interact with a tropical cyclone in the Indian ocean. In the atmosphere, the merger of two cyclonic vortices is referred to as *Fujiwhara effect*, which is named after Sakuhei Fujiwhara who initially described it in a 1921 paper about the motion of vortices in water(*Fujiwhara*, 1921).

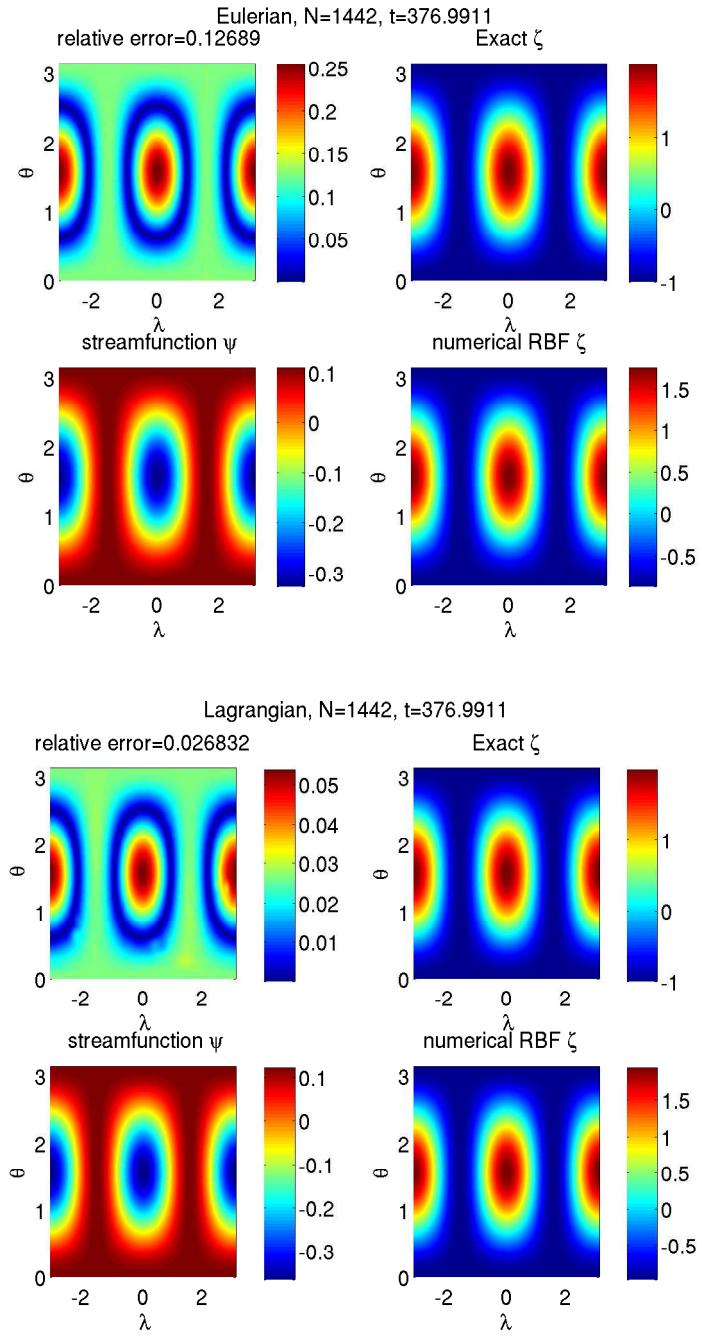

Figure 4.4: Rossby-Haurwitz wave test: relative error with different hyperdiffusion coefficients–Lagrangian coordinate with $\nu = 2 \times 10^{-6}$ and Eulerian coordinate with $\nu = 10^{-5}$. The simulation is performed on a icosalhedral grid with $N = 1442, \Delta t = 0.0015$. The upper panel is from a Eulerian coordinate simulation with $\nu = 2 \times 10^{-5}$. The lower panel is from a Lagrangian coordinate simulation with 4^{th}-order hyperdiffusion with $\nu = 2 \times 10^{-6}$.

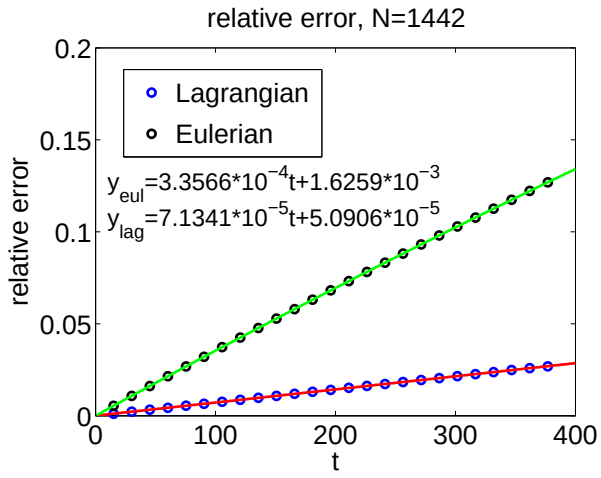

Figure 4.5: Rossby-Haurwitz wave test: track of relative error with different hyperdiffusion coefficients–Lagrangian coordinate with $\nu = 2 \times 10^{-6}$ and Eulerian coordinate with $\nu = 10^{-5}$.

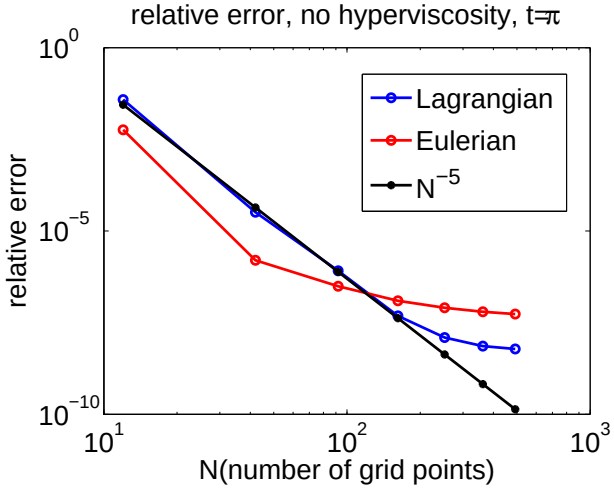

Figure 4.6: Rossby-Haurwitz wave test: relative error with no hyperviscosity. The simulational parameters are the same as Fig. 4.4

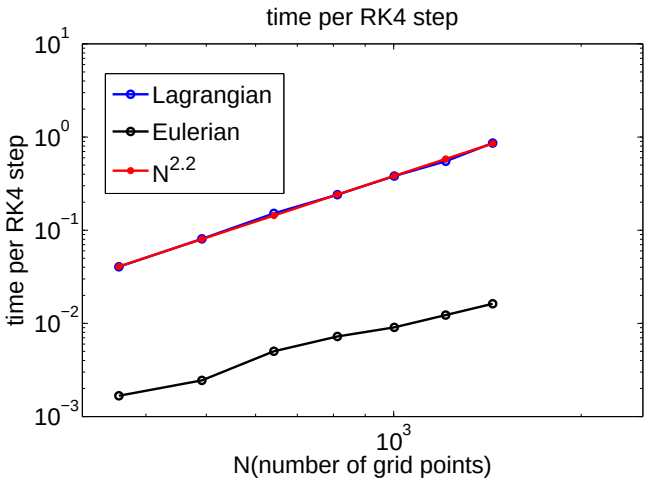

Figure 4.7: Rossby-Haurwitz wave test: time per 4^{th}-order Runge-Kutta step.

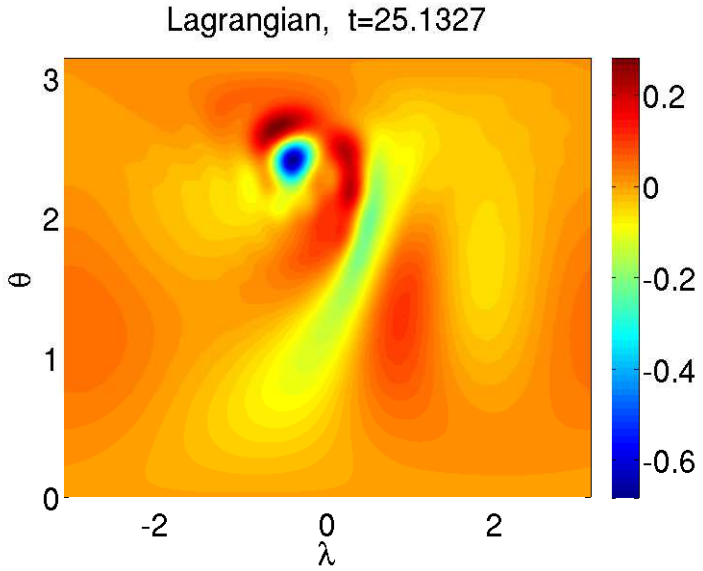

Figure 4.8: A single vortex simulation on a rotating sphere without hyperviscosity. The simulation uses $N = 1442, \triangle t = 0.0015, \alpha = 3.6153$ on a icosahedral grid. The initial Gaussian vortex is located at $(\lambda, \theta) = (0, 2.000)$.

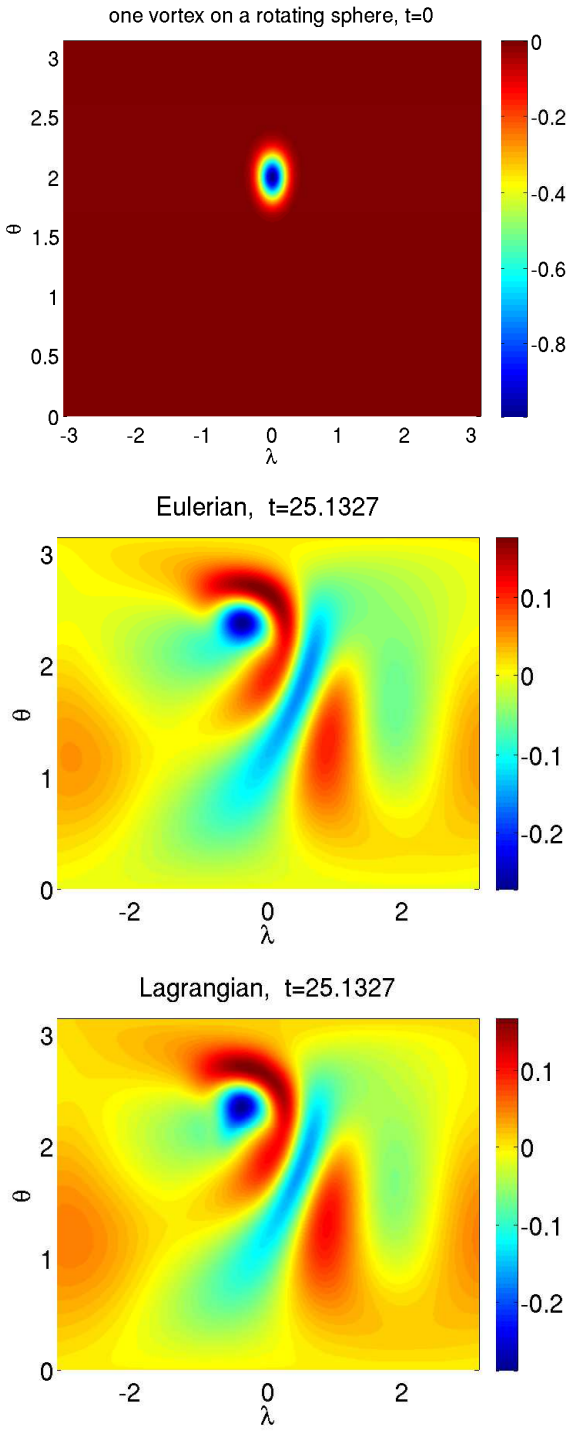

Figure 4.9: A single vortex on a rotating sphere using Eulerian and Lagrangian approaches. The simulations are performed using $N = 1442, \triangle t = 0.0015, \alpha = 3.6153$ on a icosahedral grid. The initial Gaussian vortex is located at $(\lambda, \theta) = (0, 2.000)$.

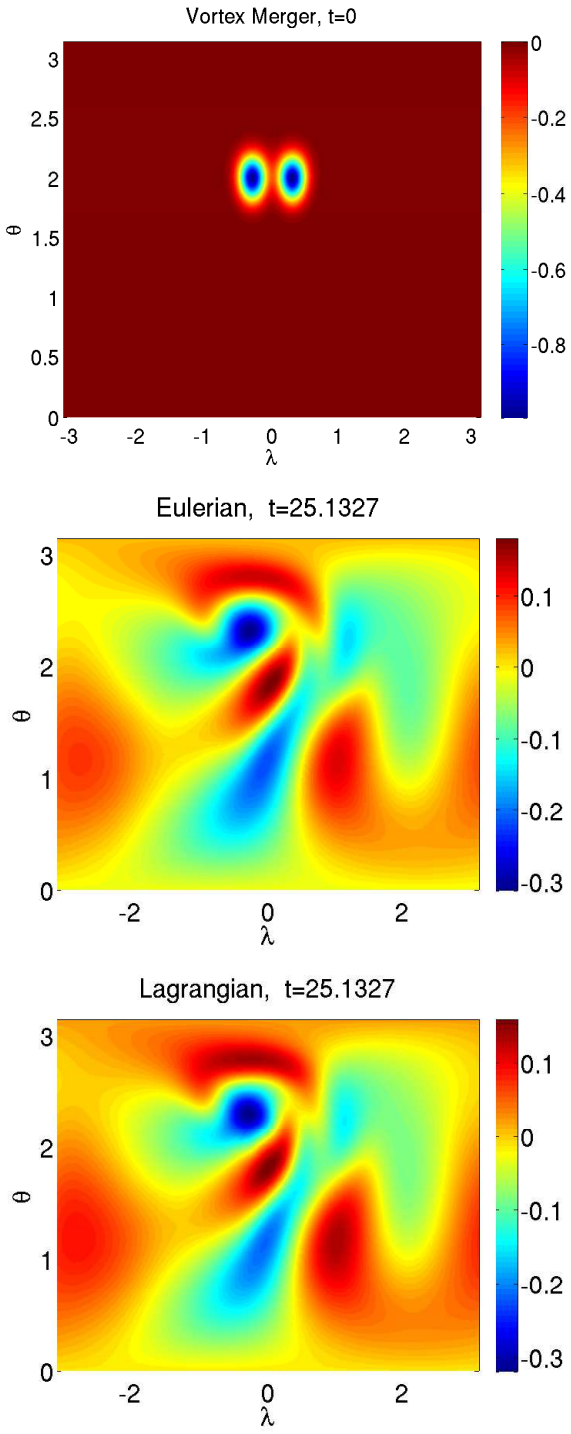

Figure 4.10: Vortex merger using Eulerian and Lagrangian approaches with the number of grid points $N = 1442$ and time step $\triangle t = 0.0015, \alpha = 3.6153$ on a icosahedral grid.

4.7 Grid generation

4.7.1 Latitude-Longitude Grid

A longitude-colatitude grid with uniform grid spacings $\delta\lambda$ and $\delta\theta$ has long been deprecated, although it is easy to generate, because the east-west grid spacing $\delta x = \sin(\theta)\delta\lambda$ shrinks to 0 as $\theta \to 0, \pi$, and this implies a severe timestep restriction when the forecasting model is integrated using an explicit method. The remedy is to thin the grid near the poles so that the number of points on a circle of constant colatitude shrinks as $\theta \to 0, \pi$.

For RBFs, there is an added problem. When the RBFs are narrow compared to the local grid spacing, RBFs are very inaccurate. When the RBFs are very wide compared to the local grid spacing, the condition number of the RBF interpolation matrix is huge. Because the uniform longitude-colatitude grid is so highly nonuniform, it is not possible to find a single absolute width ϵ that is both accurate and well-conditioned everywhere. Even a variable ϵ that is large near the poles and small near the equator is unsatisfactory. Fig. 4.11 shows that near the poles, the longitude-colatitude grid has a small grid spacing in the east-west direction but a large north-south spacing; and choice of ϵ that is satisfactory in one direction is poor in the other. Rather than accept the complexities of this near-polar thinning grid, we prefer the icosahedral grid.

4.7.2 Icosahedral Grid

The grid is generated by subdividing the triangular faces of the icosahedron into smaller triangles and placing a grid point at each vertex.

One distinction from most of this previous work is that after the grid points are defined, one must connect the points so as to make triangular or hexahedral cells to apply finite element or finite volume methods. The RBF/vortex method is gridded,

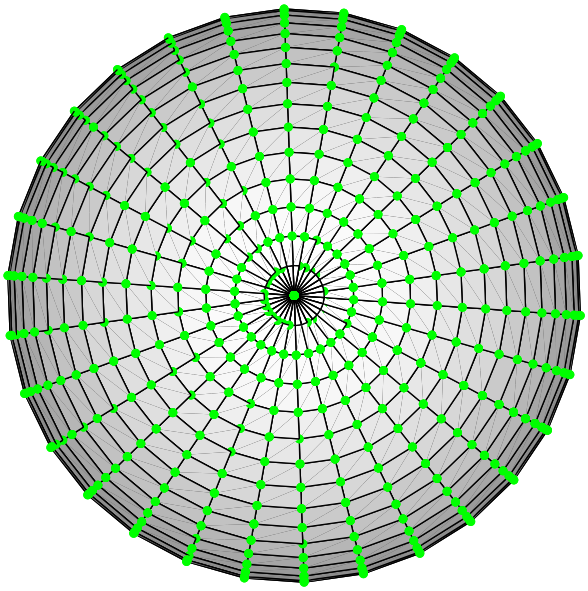

Figure 4.11: Latitude-Longitude grid

but meshless: the issue of triangular versus hexahedral elements is irrelevant.

Fig. 4.12 illustrates a plane icosahedron; the spherical harmonic is obtained by projecting the edges of the equilateral triangles onto the surface of the sphere. If two vertices are located at the north and south poles, then the remaining ten vertices are located in two rings of five at latitudes of ± 26.5651, which correspond to colatitude $\theta = \arccos(1/\sqrt{5}), \pi - \arccos(1/\sqrt{5})$. The lower right is rotated by a tenth of a rotation relative to the upper ring. The coordinates of these vertices are given in Table 4.1.

To avoid coordinate singularities at the poles, it is convenient to rotate the twelve

vertices by a small angle χ about the x-axis through

$$\begin{vmatrix} x' \\ y' \\ z' \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos(\chi) & -\sin(\chi) \\ 0 & \sin(\chi) & \cos(\chi) \end{vmatrix} \begin{vmatrix} x \\ y \\ z \end{vmatrix} \tag{4.92}$$

We chose $\chi = 2(1/\sqrt{5})$ although the choice is arbitrary.

The next step is to unfold the plane icosahedron and assign numerical labels to each of the twelve vertices, twenty faces and thirty edges. The 20×3 matrix $\vec{\vec{F}}$ and 30×2 matrix $\vec{\vec{E}}$, combined for brevity into the single Table 4.2

Table 4.1: The twelve vertices of a Icosahedron in Cartesian Coordinates and in Longitude and Colatitude

vertex no.	x	y	z	λ	θ
1	0	0	1	-	0
2	0.8506	0.2764	0.4472	$\pi/10$	0.3524π
3	0	0.8945	0.4472	$5\pi/10$	0.3524π
4	-0.8506	0.2764	0.4472	$9\pi/10$	0.3524π
5	-0.5257	0.2764	0.4472	$-7\pi/10$	0.3524π
6	0.5257	-0.7236	0.4472	$-3\pi/10$	0.3524π
7	0.5257	0.7256	-0.4472	$3\pi/10$	0.6476π
8	-0.5257	0.7236	-0.4472	$7\pi/10$	0.6476π
9	-0.8506	-0.2764	-0.4472	$-9\pi/10$	0.6476π
10	0	-0.8945	-0.4472	$-5\pi/10$	0.6476π
11	0.8506	-0.2761	-0.4472	$-\pi/10$	0.6476π
12	0	0	1	-	π

The faces of the icosahedron are equilateral plane triangles. Such a triangle of side S can always be subdivided into ν^2 equilateral triangles of side S/ν for arbitrary non-negative integer ν. The vertices of the little triangles give a grid which is uniform on the plane with the face of side S, and these vertices can be projected onto the sphere via $\vec{x} \rightarrow \vec{x}/ \parallel \vec{x} \parallel_2$. This almost what we do. However, the projected-little-plane-triangle grid is not uniform on the sphere.

For example, if one of the edges of the plane icosahedron is divided into thirds, the spherical icosahedron is divided into angles(0.350,0.757) but dividing the spherical arc directly into thirds gives interior angles of (0.369,0.738). This is not a huge nonuniformity, but most authors have tried to avoid it.

To do so, a basic operation, repeated over and over, is to subdivide the geodesic arc connecting two points arbitrarily labeled "A" and "B" on the sphere into p equal angles. This is nontrivial because the geodesic is not straight but curved in the longitude-colatitude plane. Dividing the line that passed through the sphere between the two points into even parts and projecting the line onto the geodesic arc gives a *non-uniform* subdivision of the arc.

Sadourny *et al.* (*Sadourny et al.*, 1968) showed that the Cartesian coordinates of the $(p-1)$ interior points could be calculated by solving a 3×3 matrix equation for each point. The first step is to determine Θ, the central angle subtended by the arc. Let $(\lambda_A, \theta_A))$ denote the coordinates of one endpoint and (λ_B, θ_B) denote the longitude and colatitude of the other. Then

$$\Theta = 2\arcsin(\sqrt{\sin^2(\frac{\theta_B - \theta_A}{2}) + \cos(\theta_A)\cos(\theta_B)\sin^2(\frac{\lambda_B - \lambda_A}{2})}) \tag{4.93}$$

The Cartesian coordinates of the points on the interior of the arc are found by looping over the index $j = 1, ..., (p-1)$. For each j, solve the 3×3 system

$$\begin{vmatrix} x_A & y_A & z_A \\ x_B & y_B & z_B \\ (y_A z_B - z_A y_B) & (z_A x_B - z_A x_B) & (x_A y_B - y_A x_B) \end{vmatrix} \begin{vmatrix} x_j \\ y_j \\ z_j \end{vmatrix} = \begin{vmatrix} \cos(\frac{j}{p}\Theta) \\ \cos((1-\frac{j}{p})\Theta) \\ 0 \end{vmatrix} \tag{4.94}$$

The result is the Cartesian coordinate of the j-th interior point on the arc. Note that the matrix elements depend only on the endpoints (A, B) whereas the righthand side vector is different for each j.

The justification for this matrix is as follows. We have three unit vectors $\vec{v}_A = (x_A, y_A, z_A)$, $\vec{v}_B = (x_B, y_B, z_B)$ and $\vec{v}_j = (x_j, y_j, z_j)$. The scalar product between $\vec{v}_A$ and $\vec{v}_B$ is the cosine of the angle between the two vectors, i.e., $\vec{v}_A \cdot \vec{v}_B = x_A x_j + y_A y_j + z_A z_j = \cos(\frac{j}{p}\Theta)$. Similarly, $\vec{v}_B \cdot \vec{v}_j = x_B x_j + y_B y_j + z_B z_j = \cos(1 - \frac{j}{p}\Theta)$. Lastly, the vector $\vec{v}_j = (x_j, y_j, z_j)$ lies in the plane expanded by vectors $\vec{v}_A$ and $\vec{v}_B$. The orthonormal of this plane is $\vec{v}_A \times \vec{v}_B = (y_A z_B - z_A y_B)\vec{i} + (z_A x_B - z_A x_B)\vec{j} + (x_A y_B - y_A x_B)\vec{k}$, and therefore $\vec{v}_j \cdot (\vec{v}_A \times \vec{v}_B) = x_j(y_A z_B - z_A y_B) + y_j(z_A x_B - z_A x_B) + z_j(x_A y_B - y_A x_B) = 0$.

To apply the "Sadourny partition" of an arc, we first employ it to subdivide each of the thirty faces of the spherical icosahedron into ν parts for some user-chosen non-negative integer ν. To find these interior points of a face, we pick two edges–it does not matter which two–and then apply Sadourny's procedure to arc of varying lengths connecting pairs of edge points. If we index points along the edges by k such that $k = 1$ is closed to the common vertex of the two vertices, and call the edge points A_k and B_k on the two edges "in play", then divide $A_1 - B_1$ into one part(no interior points; do nothing), $A_2 - B_2$ into two parts(one interior point) and so on until the geodesic $A_{\nu-1} - B_{\nu-1}$ is divided into $\nu - 1$ parts, yielding $\nu - 2$ interior points, a total of $\frac{(\nu-1)(\nu-2)}{2}$ interior points per icosahedral face.

The union of icosahedral vertices plus interior points on edges, i.e., $30(\nu - 1)$ plus interior points on the faces, i.e., 12 yields a total number of point in the Sadourny icosahedral grid of

$$N = 12 + 30(\nu - 1) + 20(\nu - 2)(\nu - 1)/2 \tag{4.95}$$

Fig. 4.12 shows 4 icosahedral grids with: $\nu = 3, N = 92$, $\nu = 6, N = 362$, $\nu = 9, N = 912$, $\nu = 12, N = 1442$.

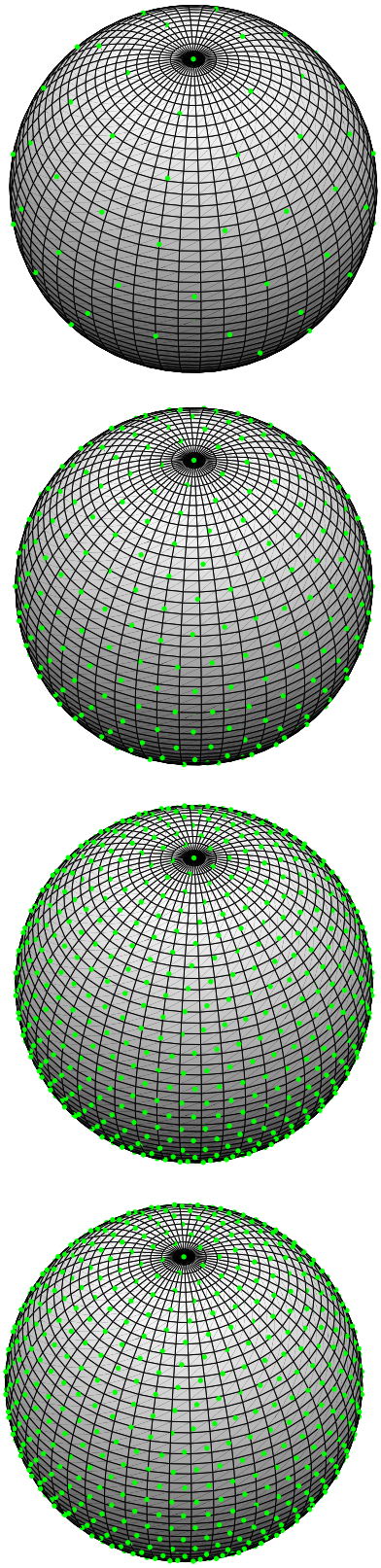

Figure 4.12: Icosahedral grid: level-3, level-6, level-9 and level-12 (from up to down). The corresponding number of grid points for the 4 levels are: $N = 92$, $N = 362$, $N = 912$, $N = 1442$.

Table 4.2: Twenty faces and Thirty edges of a Icosahedron

Face No.	F(1:20, 1:3)	Edge No.	E(1:30,1:2)
1	(1, 2, 3)	1	(1, 2)
2	(1, 3, 4)	2	(1, 3)
3	(1, 4, 5)	3	(1, 4)
4	(1, 5, 6)	4	(1, 5)
5	(1, 6, 2)	5	(1, 6)
6	(2,11, 7)	6	(2, 3)
7	(2, 3, 7)	7	(3, 4)
8	(3, 7, 8)	8	(4, 5)
9	(3, 4, 8)	9	(5, 6)
10	(4, 8, 9)	10	(6, 2)
11	(4, 5, 9)	11	(2, 11)
12	(5, 9, 10)	12	(2, 7)
13	(5, 6, 10)	13	(3, 7)
14	(6,10, 11)	14	(3, 8)
15	(6, 2, 11)	15	(4, 8)
16	(12,11, 7)	16	(4, 9)
17	(12, 7, 8)	17	(5, 9)
18	(12, 8, 9)	18	(5, 10)
19	(12, 9,10)	19	(6, 10)
20	(12,10,11)	20	(6, 11)
-	-	21	(11, 7)
-	-	22	(7, 8)
-	-	23	(8, 9)
-	-	24	(9, 10)
-	-	25	(10,11)
-	-	26	(11,12)
-	-	27	(7, 12)
-	-	28	(8, 12)
-	-	29	(9, 12)
-	-	30	(10,12)

4.8 Vortex Regridding

Vortex methods are notorious for the "vortex distorsion problem". The lattice of vortex positions, initially almost uniform or otherwise nicely aligned, become more and more distorted with time. This distortion has been shown to heavily degrade the accuracy of vortex method(*Perlman*, 1985; *Barba et al.*, 2003). *Perlman* (1985) showed that the error comes from the fact that as the configuration gets distorted, the overlap of the Gaussian particles decreases. Note that the accuracy of the Gaussian vortex method has a super-exponential dependence on the parameter h/σ, where h is the separation of the vortex particles and σ is the width of the Gaussian vortex. Radial basis functions methods are "meshfree" method in the sense of yielding high accuracy even when the grid has become highly distorted.

The procedure for vortex regridding is that after every n time steps we reinterpolated the vorticity field at the initial well-posed configuration using the *present* vortex configuration. The velocities of the vortices at the well-posed location are computed to advect the vortex particles. Fig. 4.14 gives the plot of the relative error with a 10-steps regridding (regrid every 10 steps). Before regridding is performed the relative error has a jump at around $t = 5$. The error increases 4 orders in a very short time period, which severely degrades the accuracy of RBF-Vortex method. However, after the implementation of regridding, the relative error doesnot have a sharp increase. The RBF-Vortex method stays highly accurate.

In Fig. 4.15, a 4^{th}-order hyperviscosity is added to the simulation at $t = 4\pi$. As we have seen, the relative error of the regridding increases rapidly and converges to the non-regridding curve. This means the error of the regridding RBF-Vortex method is dominated of the hyperviscosity.

The RBF-Vortex method is self-adaptive, which means the frequency of the regridding matters. Frequent regridding will eliminate the adaptive nature of the RBF-Vortex method. While sparse regridding will lead to a very distorted vortex con-

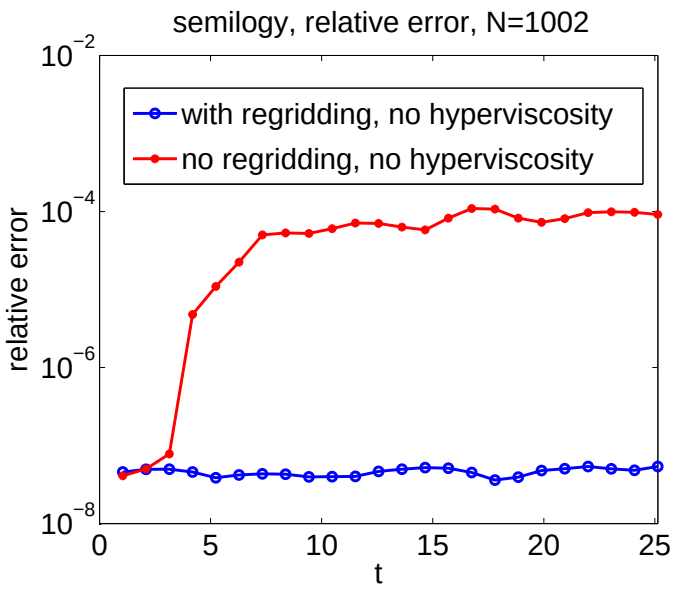

Figure 4.13: Regridding for RBF-Vortex method: relative error for Rossby-Haurwitz wave test with no hyperviscosity. Regridding is performed every 10 steps, with $\triangle t = 0.0015$.

figuration and the accuracy degraded too. Fig. 4.15 shows that there is an optimal frequency of regridding. Away from the optimal frequency, we see an increase of the relative error. Fig. 4.17 shows that a Lagrangian method adds more damping to the simulation than a Eulerian method.

Fig. 4.8 shows the evolution of one vortex on a rotating sphere after a long time of integration. At $t = 16\pi$ of integration, the center of the vortex has been distorted. Fig. 4.8 shows the figures the regridded Lagrangian case. The Eulerian case is plotted for comparison. Vortex merger is also studied with regridding RBF-Vortex method. Fig. 4.8 and Fig. 4.8 show the vortex merger of using regridded Lagrangian RBF-Vortex and non-regridded version. The Eulerian graph is plotted for comparison.

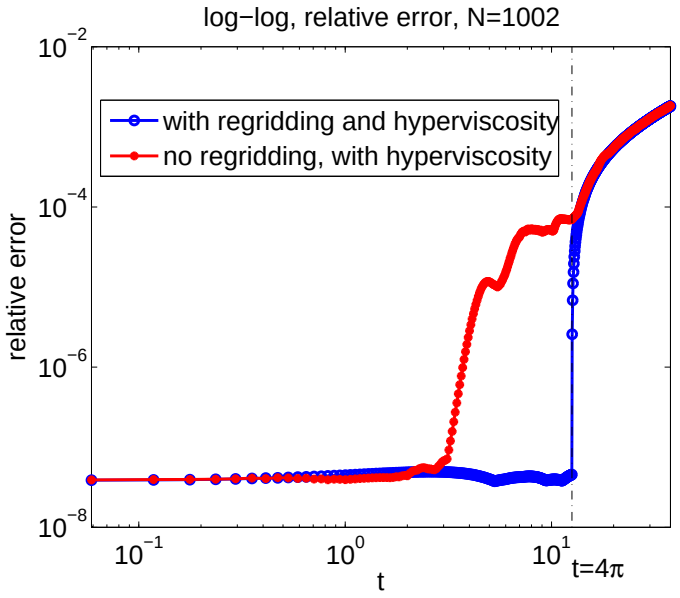

Figure 4.14: Regridding for RBF-Vortex method: relative error for Rossby-Haurwitz wave test with 4^{th}-order hyperviscosity turned on at $t = 4\pi$. Regridding is performed every 10 steps, with $\triangle t = 0.0015$.

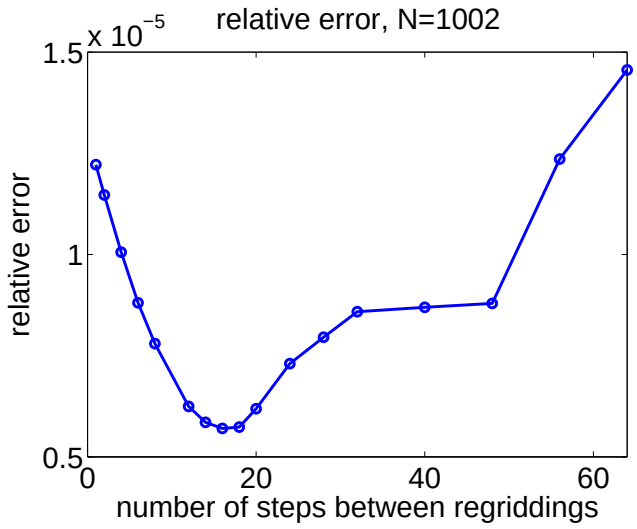

Figure 4.15: Regridding for RBF-Vortex method: relative error versus the number of time steps between regridding, with $\triangle t = 0.0015, \alpha = 3.6153$.

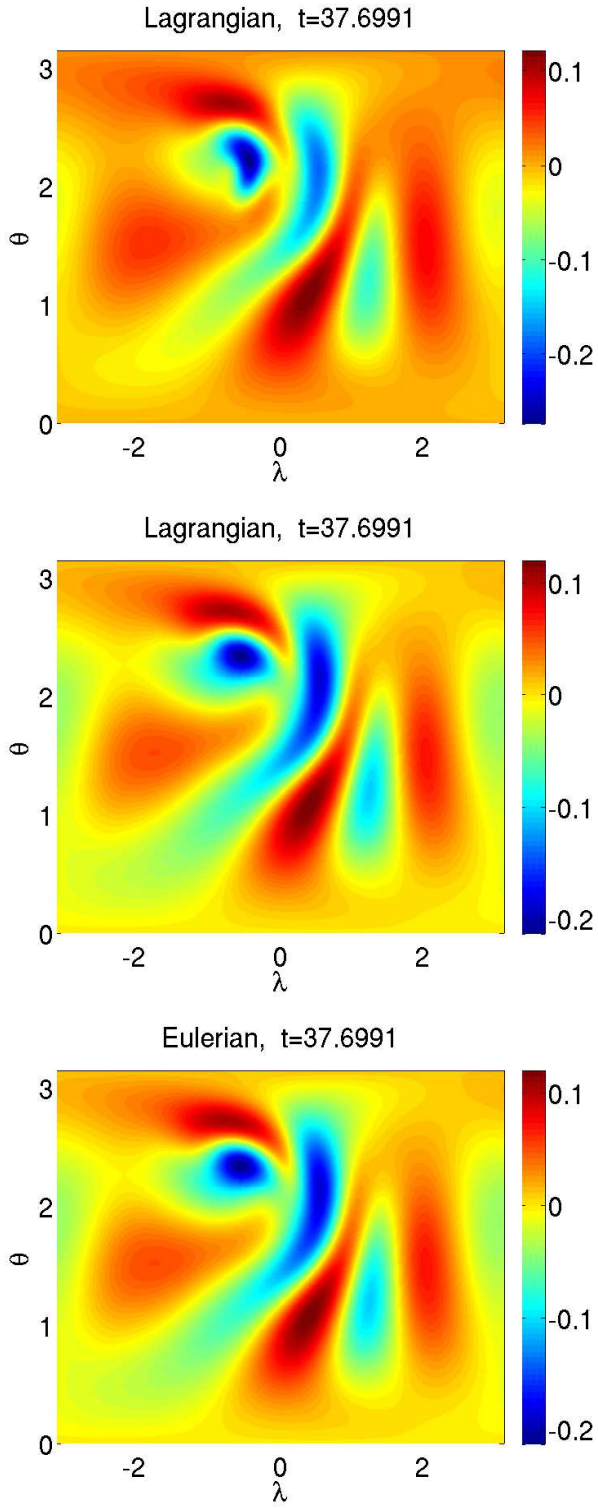

Figure 4.16: One vortex on a rotating sphere using: Lagrangian coordinate without regridding, Lagrangian coordinate with regridding and Eulerian coordinate (from up to down). The Gaussian vortex was initially located at $(\lambda, \theta) = (0, 2.000)$. The three simulations were done using the same set of parameters: $N = 1442, \triangle t = 0.0015$ on a icosahedral grid.

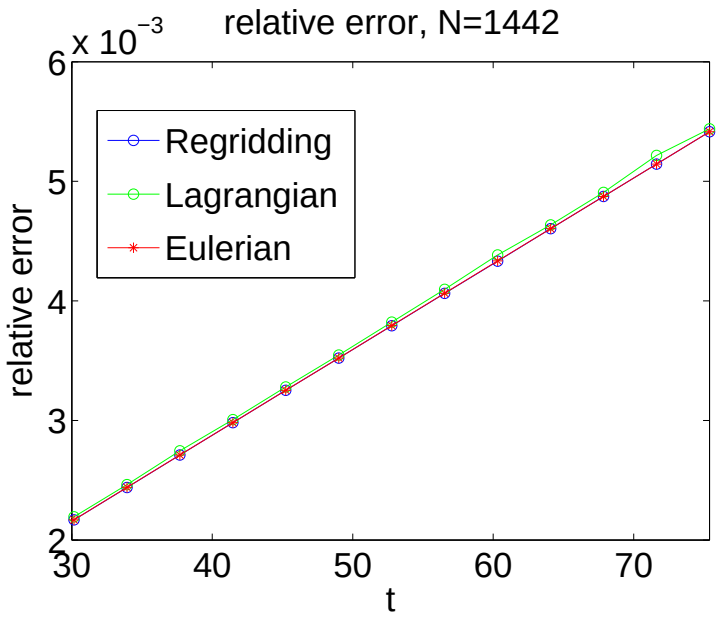

Figure 4.17: Relative error of Lagrangian, Eulerian and regridding Lagrangian method. Vortex regridding is performed every 10 time steps. The three simulations were performed using the same set of parameters: $N = 1442, \triangle t = 0.0015, \alpha = 3.6153$ on a icosahedral grid.

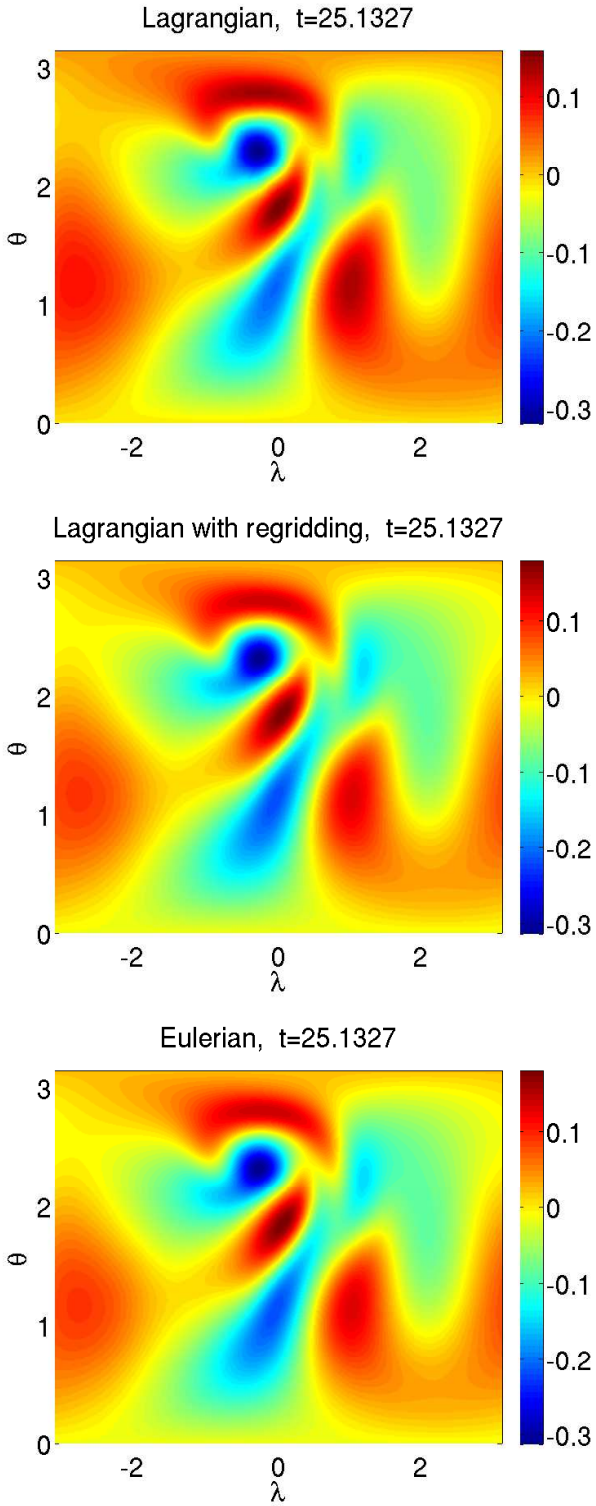

Figure 4.18: Vortex merger experiment using: Lagrangian coordinate without regridding, Lagrangian coordinate with regridding and Eulerian coordinate (from up to down). The simulations were performed on a icosahedral grid with $N = 1442, \triangle t = 0.0015, \alpha = 3.6153$. The two Gaussian vortices were initially located at $(\lambda, \theta) = (-0.325, 2.000)$ and $(\lambda, \theta) = (0.325, 2.000)$.

CHAPTER V

Summary and Future Work

5.1 Summary

This thesis consists of three parts. First, the theory of Periodized Radial Basis Functions (PRBFs) has been extended. Although PRBFs have been used before, in this thesis the theoretical background is extended with proofs and numerical experiments. Interpolation experiments show that the PRBFs are spectrally accurate. An application to solve the Mathieu equation has shown that PRBFs require a much smaller number of grid points to achieve the same accuracy than trigonometric basis(*Boyd and Xiao*, 2013). Second, an analysis of the PRBFs interpolation with Tikhonov regularization is given. In this chapter I analytically calculated the eigenvalues of the PRBFs interpolation matrices and showed that the Tikhonov regularization can be a filter in PRBFs interpolation. The explicit form of the damping factor is given. The related numerical tests were performed in both Cartesian coordinates and spherical coordinates. The Poisson operator that relates the streamfunction and vorticity is discretized using PRBFs in both one and two dimensional spaces. The eigenvalues of the resulting matrices are analytically computed(*Xiao*, 2014). Finally, the Lagrangian RBF-Vortex method is applied to solve the Barotropic Vorticity Equation (BVE). The Rossby-Haurwitz wave test, a single vortex and vortex merger experiments have been performed using Lagrangian RBF-Vortex method. The numer-

ical results agree well with the corresponding simulation using Eulerian coordinate. Although the Lagrangian RBF-Vortex method suffers from grid distortion problem, regridding technique performs well and improves the accuracy by several orders of magnitude(*Xiao et al.*, 2014).

5.2 Future work

5.2.1 Extension of PRBFs and Tikhonov regularization

Most of our discussion on PRBFs is focused on one dimensional space. Extension to higher dimension needs further study. For Gaussian PRBF, extension to higher dimensions are quite straightforward as we have shown in Chapter 3.4, supposing we work on a tensor product grid. However, no numerical experiments have been done on solving PDEs on higher dimensional periodic domain. For other PRBFs, further study is needed to apply them to higher dimensions both for interpolation and solving PDEs.

Tikhonov regularization has been shown to be a filter on a periodic domain analytically and on a spherical domain numerically. However, there are no numerical examples on applications of Tikhonov regularization to nonlinear equations, i.e., inviscid Burgers' equation $u_t + uu_x = 0$. A future direction on Tikhonov filter is to apply it to solve nonlinear dynamic equations on a periodic domain. Comparisions of the Tikhonov filter with other numerical dissipation techniques will be helpful.

5.2.2 Accelerate the RBF-Vortex method

In RBF-Vortex method, a dense matrix problem $\vec{\vec{A}}\vec{x} = \vec{b}$ has to be solved every time step. Iterative methods have to be employed to solve $\vec{\vec{A}}\vec{x} = \vec{b}$. While solving $\vec{\vec{A}}\vec{x} = \vec{b}$ using iterative methods we need to evaluate $\vec{\vec{A}}\vec{a}$ frequently. For the Gaussian RBFs, the computation of $\vec{\vec{A}}\vec{a}$ needs evaluating $\sum_{j=1}^{N} \vec{a}_j e^{-\epsilon^2(\vec{x}_i - \vec{x}_j)^2}$ N times. The eval-

uation of $\sum_{j=1}^{N} \vec{a}_j e^{-\epsilon^2(\vec{x}_i-\vec{x}_j)^2}$ is equivalent to computing the interaction of a two-body problem. This has greatly slowed down the RBF-Vortex method. However, there are several strategies to accelerate the RBF-Vortex code. First, Krasny (*Li et al.*, 2009; *Krasny and Wang*, 2011) have developed a tree code, which is closely connected to (*Greengard and Rokhlin*, 1987; *Beatson and Greengard*, 1997), to evaluate the Coulomb interactions and Multiquadratic RBF sums. With minor modifications, the tree code can be applied to Gaussian RBF sums, and the related Taylor expansion for Gaussian RBFs is given in Wang's thesis(*Wang*, 2010). Second, Fornberg, Flyer and Wright have developed the RBF-FD method to solve a variety of problems(*Flyer et al.*, 2012; *Fornberg and Lehto*, 2011). The main advantage of this method is to transform the original dense matrix into a sparse matrix by localization. The initial $O(N^3)$ problem has been reduced to an $O(N^2)$ problem. In the RBF-FD method, the kd-tree algorithm is used to find the nearest n neighbors of the target particle–the stencil of the target particle. Each particle has its own FD weights, which has to be computed separately for each particle(*Fornberg and Lehto*, 2011). Third, the interpolation of Gaussian RBFs on the surface of a sphere is very ill-conditioned, which results in a slow convergence of iterative methods and inaccuracy of the solutions. Barba(*Torres and Barba*, 2009) proposed a preconditioner to solve an ill-conditioned RBF interpolation matrix by localization and iteration.

BIBLIOGRAPHY

BIBLIOGRAPHY

Abe, Y., and Y. Iiguni (2006a), Fast computation of RBF coefficients using FFT, *Signal Processing*, *86*, 3264–3274.

Abe, Y., and Y. Iiguni (2006b), Interpolation capability of the periodic radial basis function network, *IEEE Porc-Vision, Image and Signal Processing*, *6*, 785–794.

Ahrens, B., J.-P. Xiao, A. K. Hartmann, and H. G. Katzgraber (2013), Diluted antiferromagnets in a field seem to be in a different universality class than the random-field Ising model, *Phys. Rev. B*, *88*(174408).

Barba, L., A. Leonard, and C. Allen (2005), Advances in viscous vortex methods-meshless spatial adaption based on radial basis function interpolation, *Int. J. Numer. Meth. Fluid*, *47*, 387–421.

Barba, L. A., and L. F. Rossi (2010), Global field interpolation for particle methods, *J. Comput. Phys.*, *229*, 1292–1310.

Barba, L. A., A. Leonard, and C. B. Allen (2003), Numerical investigations on the accuracy of the vortex method with and without remeshing, in *16th AIAA Computational Fluid Dynamics Conference*, AIAA 2003-3426, Orlando, Florida.

Baxter, B. (2002), Preconditioned conjugate gradients, radial basis functions, and toeplitz matrices, *Computers Mathematics with Aapplications*, *43*(3-5), 305–318.

Beale, J. T. (1988), *On the accuracy of vortex methods at large times*, Computational Fluid Dynamics and Reacting Gas Flows, 19-22 pp., Springer-Verlag, New York.

Beale, J. T., and A. Majda (1982a), Vortex methods. I. Convergence in three dimensions, *Math. Comput.*, *39*, 1–27.

Beale, J. T., and A. Majda (1982b), Vortex methods. II. Higher order accuracy in two or three dimensions, *Math. Comput.*, *39*, 29–52.

Beale, J. T., and A. Majda (1985), High order accurate vortex methods with explicit velocity kernels, *Journal of Computational Physics*, *58*, 188–208.

Beatson, R., and L. Greengard (1997), A short course on fast multipole methods, in *Wavelets, Multilevel Methods and Elliptic PDEs*, pp. 1–37, Oxford University Press.

Beatson, R., J. Cherrie, and C. Mouat (1999), Fast fitting of radial basis functions: Methods based on preconditioned GMRES iteration, *Advances in Computational Mathematics*, *11*(2-3), 253–270.

Binaghi, E., V. Pedoia, A. Guidali, and M. Guglielmin (2013), Snow cover thickness estimation using radial basis function networks, *The Cryosphere*, *7*, 841–854.

Bogomolov, V. A. (1977), Dynamics of vorticity on a sphere, *Fluid Dyn.*, *6*, 863–870.

Bogomolov, V. A. (1979), 2-dimensional hydrodynamics on a sphere, *Atmos. Ocean. Phys.*, *15*, 29–36.

Bosler, P., L. Wang, C. Jablonowski, and R. Krasny (2014), A lagrangian particle/panel method for the barotropic vorticity equations on a rotating sphere, *Fluid Dynamics Research*, accepted.

Bosler, P. A. (2013), Particle Methods for Geophysical Flow on the Sphere, PhD dissertation, University of Michigan, Ann Arbor, Department of Mathematics.

Boyd, J. P. (1989), New directions in solitons and nonlinear periodic waves: Polycnoidal waves, imbricated solitons, weakly non-local solitary waves and numerical boundary value algorithms, *Advances in Applied Mechanics*, pp. 1–82.

Boyd, J. P. (1992), The arctan/tan and Kepler-Burger mappings for periodic solutions with a shock, front, or internal boundary layer, *J. Comput. Phys.*, *98*, 181–193.

Boyd, J. P. (1994), Hyperviscous shock layers and diffusion zones: Monotonicity, spectral viscosity, and pseudospectral methods for very high order differential equations, *Journal of Scientific Computing*, *9*(1).

Boyd, J. P. (1996), The erfc-log filter and the asymptotics of the Euler and Vandeven sequence accelerations, *Proceedings of the Third International Conference on Spectral and High Order Methods*, pp. 267–276, published by the Houston Journal of Mathematics, Houston, TX.

Boyd, J. P. (1997), Construction of Lighthill's unitary functions: The imbricate series of unity, *Appl. Math. Comput.*, *86*, 1–10.

Boyd, J. P. (1999), *Chebyshev and Fourier Spectral Methods*, Dover, New York, 2nd edition.

Boyd, J. P. (2011), The near-equivalence of five species of spectrally-accurate radial basis functions(RBFs): Asymptotic approximations to the RBF cardinal functions on a uniform, unbounded grid, *J. Comput. Phys.*, *230*, 1304–1318.

Boyd, J. P., and K. W. Gildersleeve (2011), Numerical experiments on the condition number of the interpolation matrices for radial basis functions, *Applied Numerical Mathematics*, *61*, 443–459.

Boyd, J. P., and L. Wang (2009), An analytic approximation to the cardinal functions of Gaussian radial basis functions on a one-dimensional infinite uniform lattice, *Appl. Math. Comput.*, *215*, 2215–2223.

Boyd, J. P., and J.-P. Xiao (2013), Periodized radial basis functions, part I: theory, *Appl. Numer. Math*, submitted.

Brown, D., I. Ling, E. Kansa, and et al. (2005), On approximate cardinal preconditioning methods for solving pdes with radial basis functions, *Engineering Analysis With Boundary Elements*, *4*, 343–353.

Buhmann, M. D. (2000), Radial basis functions, *Acta Numerica*, *9*, 1–38.

Buhmann, M. D. (2003), *Radial Basis Functions: Theory and Implementations*, *Cambridge monographs on applied and computational mathematics*, vol. 12, Cambridge University Press.

Calvetti, D., L. Morigi, L. Reichel, and F. Sgallari (2000), Tikhonov regularization and the L-curve for large discrete ill-posed problems, *Journal of Computational and Applied Mathematics*, *123*, 423–446.

Charney, J., R. Fjortoft, and J. Von Neumann (1950), Numerical integration of the barotropic vorticity equation, *Tellus*, *2*, 237–254.

Chorin, A. J. (1973), Numerical study of slightly viscous flow, *J. Fluid Mech.*, *57*, 785–796.

Chorin, A. J., and P. S. Bernard (1973), Discretization of a vortex sheet, with an example of roll-up, *Journal of Computational Physics*, *13*, 363–379.

Cottet, G. H., and P. D. Koumoutsakos (2000), *Vortex Methods: Theory and Practice*, Cambridge University Press.

Degond, P., and S. Mas-Gallic (1989), The weighted particle method for convection-diffusion equations. part 1. the case of an isotropic viscosity, *Mathematics of Computation*, *53*, 485–.

Diago, L., M. Kitago, and I. Hagiwara (2004), Image interpolation from scattered data - an iterative approach using CSRBF, *Proceedings: Communications, Information Technologies and Computing*, *1*, 120–125.

Dibattista, M., and L. Polvani (1998), Barotropic vortex pairs on a rotating sphere, *J. Fluid Mech.*, *358*, 107–133.

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

El-Shafie, A., A. Noureldin, M. Taha, A. Hussain, and M. Mukhlisin (2012), Dynamic versus static neural network model for rainfall forecasting at Klang River Basin, Malaysia, *Hydrology and Earth System Sciences*, *16*, 1151–1169.

Fasshauer, G. E. (2007), *Meshfree Approximation Methods with MATLAB*, *Interdisciplinary Mathematical Sciences*, vol. 6, World Scientific.

Fedoseyev, A., M. Friedman, and E. Kansa (2002), Improved multiquadric method for elliptic partial differential equations via PDE collocation on the boundary, *Comput. Math.*, *43*, 439–455.

Flyer, N., and B. Fornberg (2011), Radial basis function: Developments and applications to planetary scale flows, *Computers and Fluids*, *46*, 23–32.

Flyer, N., and E. Lehto (2010), Rotational transport on a sphere: local node refinement with radial basis functions, *Journal of Computational Physics*, *229*(6), 1954–1969.

Flyer, N., and G. B. Wright (2007), Transport schemes on a sphere using radial basis functions, *Journal of Computational Physics*, *226*, 1059–1084.

Flyer, N., and G. B. Wright (2009), A radial basis function method for the shallow water equations on a sphere, *Proc. R. Soc. A*, *465*(2106), 1949–1976.

Flyer, N., E. Lehto, S. Blaise, G. B. Wright, and A. St-Cyr (2012), A guide to RBF-generated finite differences for nonlinear transport: Shallow water simulations on a sphere, *J. Comput. Phys.*, *231*(11), 4078–4095.

Fornberg, B., and E. Lehto (2011), Stabilization of RBF-generated finite difference methods for convective PDEs, *Journal of Computational Physics*, *230*, 2270–2285.

Fornberg, B., and C. Piret (2008), On choosing a radial basis function and a shape parameter when solving a convective PDE on a sphere, *Journal of Computational Physics*, *227*, 2758–2780.

Fornberg, B., and G. Wright (2004), Stable computation of multiquadric interpolants for all values of the shape parameter, *Comput. Math. Appl.*

Fornberg, B., E. Larsson, and N. Flyer (2011), Stable computations with Gaussian radial basis functions, *SIAM J. Sci. Comput.*, *33*, 869–892.

Fujiwhara, S. (1921), The natural tendency towards symmetry of motion and its application as a principle in meteorology, *Quarterly Journal of The Royal Meteorological Society*, *47*(200), 287–293.

Greengard, C. (1985), The core spreading vortex method approximations the wrong equation, *Journal of Computational Physics*, *61*, 345–348.

Greengard, L., and V. Rokhlin (1987), A Fast Algorithm for Particle Simulation, *J. Comput. Phys.*, *73*, 325–348.

Gryanik, V. M. (1986), Singular geostrophic vortices in the beta plane as a model of synoptic vortices, *Okeanologiya*, *26*, 174–179.

Gryanik, V. M. (1988a), Dynamics of singular geostrophic vortices in a two-level model of the atmosphere or ocean, *Izv. Atmos. Oceanic Phys.*, *19*, 227–240.

Gryanik, V. M. (1988b), Localized vortices in the equatorial atmosphere: Structure and dynamics, *Izv. Atmos. Oceanic Phys.*, *24*, 13–19.

Hald, O. (1979), Convergence of vortex methods for Euler's equations II, *SIAM J. Num. Anal.*, *16*, 726–755.

Hald, O. (1987), Convergence of vortex methods for Euler's equations III, *SIAM J. Num. Anal.*, *24*, 538–582.

Hald, O., and V. M. Del Prete (1978), Convergence of vortex methods for Euler's equations, *Math. Comput.*, *32*, 791–801.

Hansen, P. C. (1992), Analysis of discrete ill-posed problems by means of the l-curve, *SIAM REVIEW*, *34*(4), 561–580.

Hansen, P. C., and D. P. O'Leary (1993), The Use of the L-Curve in the Regularization of Discrete Ill-Posed Problems, *SIAM J. Sci. Comput.*, *14*(6), 1487–1503.

Hardy, R. (1971), Multiquadric equations of topography and other irregular surfaces, *Journal of Geophysical Research*, *76*(8), 1905–1915.

Hickernell, F., and Y. Hon (1998), Radial basis function approximation of the surface wind field from scattered data, *Internat. J. Appl. Sci. Comput.*, *4*, 221–247.

Hickernell, F., and Y. Hon (1999), Radial basis function approximation as smoothing splines, *Appl. Math. Comput.*, *102*(1), 1–24.

Hogg, N. G., and H. M. Stommel (1985), The heton, and elementary interaction between discrete baroclinic geostrophic vortices, and its implications concerning eddy heat-flow, *Proc. R. Soc. London*, *A397*, 1–20.

Huang, Z., J.-P. Xiao, and J. P. Boyd (2014), Adaptive Radial Basis Function and Hermite Function Pseudospectral Methods for Computing Eigenvalues of the Prolate Spheroidal Wave Equation for Very Large Bandwidth Parameter, *Journal of Computational Physics*, submitted.

Hubbert, S., and S. Mueller (2006), Interpolation with circular basis functions, *Numerical Algorithms*, *42*, 75–90.

Iske, A. (2004), *Multiresolution Methods in Scattered Data Modelling*, *Lecture Notes in Computational Science and Engineering*, vol. 37, Springer, Heidelberg.

Jablonowski, C., and D. L. Williamson (2011), *The Pros and Cons of Diffusion, Filters and Fixers in Atmospheric General Circulation Models*, *Lecture Notes in Computational Science and Engineering*, vol. 80, 381-493 pp., Springer, Heidelberg.

Jamaloodeen, M. I., and P. K. Newton (2007), The N-vortex problem on a rotating sphere. II. Heterogeneous Ppltonic solid equilibria, *Proc. R. Soc. London A*, *462*, 3277–3299.

Kansa, E. (1990a), Multiquadrics - A scattered data approximation scheme with applications to computational fluid-dynamics, I. Surface approximations and partial derivative estimates, *Comput. Math. Appl.*, *19*(8/9), 127–145.

Kansa, E. (1990b), 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*(8/9), 147–161.

Kidambi, R., and P. K. Newton (1998a), Point vortex motion on a sphere with solid boundaries, *Phys. Fluids*, *12*, 581–588.

Kidambi, R., and P. K. Newton (1998b), Streamline topologies for integrable vortex motion on a sphere, *Physica D*, *140*, 95–125.

Kidambi, R., and P. K. Newton (1998c), Motion of three point vortices on a sphere, *Physica D*, *116*, 143–175.

Kimura, Y., and H. Okamoto (1987), Vortex on a sphere, *J. Phys. Soc. Japan*, *56*, 4203–4206.

Klyatskin, K. V., and G. M. Reznik (1989), Point vortices on a rotating sphere, *Oceanology*, *29*, 12–16.

Koumoutsakos, P. (1997), Inviscid axisymmetrization of an elliptical vortex, *J. Comp. Phys.*, *138*, 821–857.

Koumoutsakos, P. (2005), Flow simulations using particles, *Annual Review of Fluid Mechanics*, *37*, 457–487.

Koumoutsakos, P., and A. Leonard (1995), High resolution simulations of the flow around an impulsively started cylinder using vortex methods, *J. Fluid Mech.*, *296*, 1–38.

Krasny, R. (1986), Disingularization of periodic vortex sheet roll-up, *Journal of Computational Physics*, *65*, 291–313.

Krasny, R., and L. Wang (2011), Fast evaluation of multiquadratic RBF sums by a cartesian treecode, *SIAM J. Sci. Comput.*, *33*(5), 2341–2355.

Kuhlbrodt, T., and P. Nevic (2000), Low-order point vortex models of atmospheric blocking, *Meterology and Atmos. Phys.*, *73*, 127–138.

Lawson, C. L., and R. J. Hanson (1974), *Solving Least Squares Problems*, Prentic-Hall, Englewodd Cliffs, NJ.

Leonard, A. (1980), Vortex methods for flow simulation, *Journal of Computational Physics*, *37*, 289–335.

Levy, M. N., R. D. Nair, and H. M. Tufo (2009), A high-order element-based Galerkin method for the barotropic vorticity equation, *Int. J. Numer. Meth. Fluids*, *59*, 1369–1387.

Li, P., H. Johnston, and R. Krasny (2009), A Cartesian treecode for screened coulomb interactions, *Journal of Computational Physics*, *228*, 3858–3868.

Light, W., and E. Cheney (1992), Interpolation by periodic radial basis functions, *J. Math. Anal. Appl.*, *168*, 110–130.

Ling, L., and E. Kansa (2004), Preconditioning for radial basis functions with domain decomposition methods, *Mathematical and Computer Modelling*, *40*(13), 1413–1427.

Majda, A. J., and A. L. Bertozzi (2002), *Vorticity and Incompressible Flow*, Cambridge University Press.

McLachlan, N. W. (1947), *Theory and Application of Mathieu Functions*, Clarendon Press, Oxford.

McWilliams, J. C., and N. J. Zabusky (1982), Interactions of isolated vortices. 1. modons colliding with modons, *Geophys. Astro. Fluid Dyn.*, *19*, 207–227.

Moroney, T., and I. Turner (2007), A three-dimensional finite volume method based on radial basis functions for the accurate computational modeling of nonlinear diffusion equations, *J. Comput. Phys.*, *225*, 1409–1426.

Morse, P. M., and D. W. Feshbach (1953), *Methods of Theoretical Physics*, McGraw-Hill, New York.

Newton, P. K., and S. D. Ross (2006), Chaotic advection in the restricted four-vortex problem on a sphere, *Physics D*, *223*, 36–63.

Newton, P. K., and T. Sakajo (2006), The N-vortex problem on a rotating sphere. I. Multi-frequency configurations, *Proc. R. Soc. A.*, *462*, 149–169.

Newton, P. K., and T. Sakajo (2007), The N-vortex problem on a rotating sphere. III. Ring configurations coupled to a background field, *Proc. R. Soc. A.*, *463*, 961–977.

Newton, P. K., and T. Sakajo (2008), Interacting dipole pairs on a rotating sphere, *Proc. R. Soc. A.*, *464*, 1525–1541.

Olver, F., D. Lozier, R. Boisvert, and C. Clark (2010), *NIST Handbook of Mathematical Functions*, Cambridge University Press, New York.

Paes, F. F., H. F. Campos Velho, and F. M. Ramos (2011), *Artificial Neural Networks for Estimating the Atmospheric Pollutant Sources*, *Computational and Analytic Aspects*, vol. Integral Methods in Science and Engineering, 261-271 pp., Birkhauser, 422p.

Panakkat, A., and H. Adeli (2007), Neural network models for earthquake magnitude prediction using multiple seismicity indicators, *Int. J. Neural Syst.*, *17*(1), 13–33.

Pekarsky, S., and J. Marsden (1998), Point vortices on a sphere: Stability of relative equilibria, *J. Math. Phys.*, *39*, 5894–5907.

Perlman, M. (1985), On the Accuracy of Vortex Methods, *Journal of Computational Physics*, *59*, 200–223.

Phillips, N. J. (1956), The general circulation of the atmosphere: A numerical experiment, *Quart. J. Roy. Met. Soc.*, *82*, 123–164.

Platte, R. (2011), How fast do radial basis function interpolants of analytic functions converge?, *IMA J. Numer. Anal.*, *31*, 1578–1597.

Platte, R., and T. Driscoll (2005), Polynomials and potential theory for Gaussian radial basis function interpolation, *SIAM J. Numer. Anal.*, *43*, 750–766.

Platte, R., and T. Driscoll (2006), Eigenvalue stability of radial basis function discretizations for time-dependent problems, *Comput. Math. Appl.*, *51*, 1251–1268.

Ploumhans, P., and G. S. Winckelmans (2000), Vortex methods for high-resolution simulations of viscous flow past bluff bodies of general geometry, *J. Comp. Phys.*, *165*, 354–406.

Reznik, G. M. (1986), Point vortices on the beta plane and Rossby solitons, *Okeanologiya*, *26*, 165–173.

Reznik, G. M. (1992), Dynamics of singular vortices on the beta plane, *J. Fluid Mech.*, *240*, 405–432.

Reznik, G. M., and Z. Kizner (2007), Two-layer quasi-geostrophic singular vortices embedded in a regular flow. Part 1. Invariants of motion and stability of vortex pairs, *J. Fluid Mech.*, *584*, 185–202.

Rosenhead, L. (1932), The point vortex approximation of a vortex sheet, *Proc. R. Soc. London Ser. A*, *134*(823), 170–192.

Sadourny, R., A. Arakawa, and Y. Mintz (1968), Integration of the nondivergent barotropic equation with an icosahedral hexagonal grid for the sphere, *Mon. Wea. Rev.*, *96*, 351–356.

Sakajo, T. (2004), Motion of a vortex sheet on a sphere with pole vortices, *Phys. Fluids*, *16*, 717–727.

Sakajo, T. (2007a), Integrable four-vortex motion on sphere with zero moment of vorticity, *Phys. Fluids*, *19*, Art. No. 017,109.

Sakajo, T. (2007b), Invariant dynamical systems embedded in the n-vortex problem on a sphere with pole vortices, *Physica D*, *225*(2), 235–236.

Santhanam, T., and A. Subhajini (2011), An efficient weather forecasting system using radial basis function neural network, *Journal of Computer Science*, *7*(7), 962–966.

Schaback, R., and H. Wendland (2006), Kernel techniques: From machine learning to meshless methods, *Acta Numerica*, *15*, 543–639.

Shen, J., and L.-L. Wang (2009), On spectral approximations in elliptical geometries using mathieu functions, *Math. Comput.*, *78*, 815–844.

Shiguemori, E. H., H. F. d. Campos Velho, and J. D. S. d. Silva (2008), Atmospheric temperature retrieval from satellite data: New non-extensive artificial neural network approach, in *Proceedings of the 2008 ACM symposium on Applied computing*, pp. 1688–1692, New York, NY, USA.

Shin, S.-E., J.-Y. Han, and J.-J. Baik (2006), On the critical separation distance of binary vortices in a nondivergent barotropic atmosphere, *Journal of the Meteorological Society of Japan*, *84*(5), 853–869.

Torres, C. E., and L. Barba (2009), Fast radial basis function interpolation with Gaussians by localization and iteration, *Journal of Computational Physics*, *228*, 4976–4999.

Trefethen, L. N., and D. Bau (1997), *Numerical Linear Algebra*, SIAM, Philadelphia.

Wang, L. (2010), Radial basis functions and vortex methods and their application to vortex dynamics on a rotating sphere, PhD dissertation, University of Michigan, Ann Arbor, Department of Mathematics.

Wang, L. L., and J. Zhang (2011), An improved estimate of PSWF approximation and approximation by Mathieu functions, *J. Math. Anal. Applic.*, *379*, 35–47.

Wendland, H. (2005), *Scattered Data Approximation*, Cambridge University Press.

Wong, S., Y. Hon, and M. Golberg (2002), Compactly supported radial basis functions for shallow water equations, *Appl. Math. Comput.*, *127*, 79–101.

Wright, G., N. Flyer, and D. Yuen (2010), A hybrid radial basis function-pseudospectral method for thermal convection in a 3-D spherical shell, *Geochemistry, Geophysics, Geosystems*, *11*(7).

Xiao, J.-P. (2014), An analysis of Periodic Radial Basis Functions interpolation with Tikhonov regularization, *Journal of Computational Physics*, submitted.

Xiao, J.-P., L. Wang, and J. P. Boyd (2014), RBF-Vortex Methods for the Barotropic Vorticity Equation on a Sphere, *To be submitted to Journal of Computational Physics*.

Xu, Y., and E. Cheney (1992), Interpolation by periodic radial functions, *Comput. Math. Appl.*, *24*, 201–215.

Zhang, M., K. Wang, C. Zhang, H. Chen, H. Liu, Y. Yue, I. Luffman, and X. Qi (2011), Using the radial basis function network model to assess rocky desertification in northwest Guangxi, China, *Environ Earth Sci.*, *62*, 69–76.