Approximation by Translates of a Radial Basis Function

By

Stephen Hales

Department of Mathematics and Computer Science

December 2000

SUBMITTED TO THE UNIVERSITY OF LEICESTER FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
Acknowledgements

This work has been supported by the EPSRC. The facilities provided by the Mathematics and Computer Science department are gratefully acknowledged. I would also like to thank Jeremy Levesley for his guidance and counterexamples.
Contents

1 Introduction .......................................................... 1
   1.1 Interpolation .................................................. 1
   1.2 Solving partial differential equations ...................... 7
   1.3 Error analysis ............................................... 9

2 Computational methods .............................................. 14
   2.1 Introduction ................................................ 14
   2.2 Hierarchical refinement algorithm ...................... 25
      2.2.1 Local solutions and gridding data ................. 27
      2.2.2 Hierarchical iterative refinement .................. 29
      2.2.3 Pseudocode ............................................ 33
      2.2.4 Numerical results .................................... 33
      2.2.5 Conclusion .............................................. 48
   2.3 Multilevel polyharmonic splines ......................... 51
      2.3.1 Introduction .......................................... 51
      2.3.2 Iterative refinement .................................. 52
      2.3.3 Data configuration .................................... 54
      2.3.4 Error analysis ......................................... 55
      2.3.5 Polyharmonic splines ................................ 58
      2.3.6 Numerical results ..................................... 59
      2.3.7 Conclusion .............................................. 65
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Compactly supported functions</td>
<td>66</td>
</tr>
<tr>
<td>3.1</td>
<td>Outline</td>
<td>66</td>
</tr>
<tr>
<td>3.2</td>
<td>Introduction</td>
<td>66</td>
</tr>
<tr>
<td>3.3</td>
<td>Characterisation of positive definiteness</td>
<td>75</td>
</tr>
<tr>
<td>3.4</td>
<td>Smoothness of compactly supported basis functions</td>
<td>79</td>
</tr>
<tr>
<td>3.5</td>
<td>Polynomials of minimal degree</td>
<td>85</td>
</tr>
<tr>
<td>3.6</td>
<td>Conclusion</td>
<td>87</td>
</tr>
<tr>
<td>4</td>
<td>Creating a native space</td>
<td>88</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>88</td>
</tr>
<tr>
<td>4.2</td>
<td>Designing a basis function</td>
<td>90</td>
</tr>
<tr>
<td>4.3</td>
<td>Constructing a basis function</td>
<td>94</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Approximating $\phi$</td>
<td>95</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Bounding $F_d(\psi)$</td>
<td>106</td>
</tr>
<tr>
<td>4.4</td>
<td>Numerical results</td>
<td>108</td>
</tr>
<tr>
<td>4.5</td>
<td>Conclusion</td>
<td>112</td>
</tr>
</tbody>
</table>

Bibliography | 113 |
Approximation by Translates of a Radial Basis Function

By

Stephen Hales

Abstract

The aim of this work is to investigate the properties of approximations obtained by translates of radial basis functions. A natural progression in the discussion starts with an iterative refinement scheme using the strictly positive definite inverse multiquadric. Error estimates for this method are greatly simplified if the inverse multiquadric is replaced by a strictly conditionally positive definite polyharmonic spline. Such error analysis is conducted in a native space generated by the Fourier transform of the basis function. This space can be restrictive when using very smooth basis functions. Some instances are discussed where the native space of \( \Phi \) can be enlarged by creating a strictly positive definite basis function with comparable approximating properties to \( \Phi \), but with a significantly different Fourier transform to \( \Phi \). Before such a construction is possible however, strictly positive definite functions in \( \mathbb{R}^d \) for \( d < \infty \) with compact support must be examined in some detail. It is demonstrated that the dimension in which a function is positive definite can be determined from its univariate Fourier transform.

This work is biased towards the computational aspects of interpolation, and the theory is always given with a view to explaining observable phenomena.
Chapter 1

Introduction

1.1 Interpolation

Radial basis functions have become widely used for multivariate interpolation of scattered data. A radial function $\Phi$ can be expressed in terms of a univariate function $\phi$, via

$$\Phi(x) = \phi(||x||)$$

for $x \in \mathbb{R}^d$ and where $|| \cdot ||$ is the Euclidean norm. An interpolant is generated by a linear combination of basis functions centred at data locations, and in some cases appended by a polynomial. This yields an intuitively reasonable approach to interpolating scattered data, where the contribution from each node is radially symmetric. This interpolation technique has demonstrated numerical success for a range of applications from neural networks [18] to topographical data fitting [28]. See Golberg & Chen [24] for a comprehensive survey of applications, and Schaback gives a good survey in [51]. Radial basis function interpolation is also stable in high dimensions, a feature which is advantageous for problems in biomolecular dynamics; see Skeel [54].
The definition of (conditionally) positive definite functions varies slightly with different authors. The terminology here will conform to that of Schoenberg [52] and Micchelli [40].

**Definition 1.1** A function
\[ \Phi : \Omega \times \Omega \to \mathbb{R} \]
is (strictly) positive definite on \( \Omega \subseteq \mathbb{R}^d \) if and only if for any set of distinct points \( \{x_1, \ldots, x_N\} \subseteq \Omega \) then
\[ \sum_{i,j=1}^{N} \nu_i \nu_j \Phi(x_i - x_j) \]
is (positive) nonnegative for every vector \( \nu = (\nu_1, \ldots, \nu_N) \in \mathbb{R}^N \setminus \{0\} \). Let \( \text{PD} \) denote strictly positive definite functions.

**Definition 1.2** A function
\[ \Phi : \Omega \times \Omega \to \mathbb{R} \]
is (strictly) conditionally positive definite of order \( m \) on \( \Omega \subseteq \mathbb{R}^d \) if and only if for any set of distinct points \( \{x_1, \ldots, x_N\} \subseteq \Omega \) then
\[ \sum_{i,j=1}^{N} \nu_i \nu_j \Phi(x_i - x_j) \]
is (positive) nonnegative whenever the vector \( \nu = (\nu_1, \ldots, \nu_N) \in \mathbb{R}^N \setminus \{0\} \) has the property
\[ \sum_{i=1}^{N} \nu_i p(x_i) = 0 \]
for all \( d \)-variate polynomials \( p \) of degree \((m - 1)\) or less. Let \( \text{CPD}_m \) denote strictly conditionally positive definite of order \( m \) functions.

Consider data \( f(x_i) \in \mathbb{R} \) located at distinct points \( x_i \in \mathbb{R}^d \) for \( i = 1, \ldots, N \). In this setting, let the interpolant be a linear combination of translated basis functions thus
\[ s(x) = \sum_{i=1}^{N} \lambda_i \Phi(x - x_i) \]  

where \( s(x_i) = f(x_i) \) for \( i = 1, \ldots, N \), and \( \Phi(x) = \phi(||x||) \) is PD. Then \( \Phi \) is a radial function. This construction relies on the effective solution of a system of linear equations which can be both large and dense. The associated interpolation matrix \( A = (\Phi(x_i - x_j))_{1 \leq i, j \leq N} \) is then positive definite, and therefore nonsingular.

Major work has been done by Schoenberg in [52] to give the necessary and sufficient conditions to ensure that \( \phi \) is positive definite. This result will be examined and rewritten in the Fourier domain in Chapter 3.

**Definition 1.4** A function \( f \) is completely monotone on \([0, a)\) for \( a > 0 \), written \( f \in \text{CM}[0, a) \), if

(i) \( f \in C[0, a) \),

(ii) \( f \in C^\infty(0, a) \),

(iii) \((-1)^k f^{(k)}(r) \geq 0\) for \( 0 < r < a \) and \( k = 0, 1, 2, \ldots\)

**Theorem 1.5** The function \( \phi \) is positive definite if and only if \( \psi \) is completely monotone on \([0, \infty)\), where \( \psi(r) = \phi(\sqrt{r}) \).

The choice of basis function is both user and problem dependent, and the following list shows some popular examples of PD functions.

\[
\begin{align*}
    e^{-cr^2}, & \quad c > 0, \text{ gaussian}, \\
    (c^2 + r^2)^{\beta/2}, & \quad \beta < 0, \text{ generalised inverse multiquadric}, \\
    r^\nu K_\nu(r), & \quad \nu > 0, \text{ Sobolev spline}.
\end{align*}
\]

Compactly supported functions (which cannot be PD in all space dimensions) are dealt with at length in Chapters 3 and 4.

The more general class of \( \text{CPD}_m \) functions can also yield nonsingular interpolation matrices. In this case a number of extra conditions are imposed. Let the
interpolant now be a linear combination of translated basis functions appended by some polynomial thus

\[ s(x) = \sum_{i=1}^{N} \lambda_i \Phi(x - x_i) + p_{m-1}(x) \]  \hspace{1cm} (1.6)

where \( p_{m-1} \in \mathcal{P}_{m-1}^d \), \( \Phi(x) = \phi(\|x\|) \) is \( \mathbf{CPD}_m \), and \( s(x_i) = f(x_i) \) for \( i = 1, \ldots, N \) with the extra conditions

\[ \sum_{i=1}^{N} \lambda_i q(x_i) = 0 \quad \text{for all } q \in \mathcal{P}_{m-1}^d. \]  \hspace{1cm} (1.7)

Here \( \mathcal{P}_{m-1}^d \) is the space of all \( d \)-variate polynomials of degree \((m-1)\) or less, and is of dimension \( Q \), where

\[ Q = \begin{pmatrix} m - 1 + d \\ d \end{pmatrix}. \]

Then the additional polynomial is given by

\[ p_{m-1} = \sum_{j=1}^{Q} \alpha_j t_j \]  \hspace{1cm} (1.8)

where \( t_1, \ldots, t_Q \) is a basis for \( \mathcal{P}_{m-1}^d \). So the coefficients \( \lambda \) are chosen such that they annihilate all polynomials up to degree \((m-1)\), and combined with \( \alpha \) they together satisfy the interpolation conditions. The weights in (1.6), (1.7) and (1.8) form the solution vector to the following system

\[
\begin{bmatrix}
A & P \\
P^T & 0
\end{bmatrix}
\begin{bmatrix}
\lambda \\
\alpha
\end{bmatrix}
=
\begin{bmatrix}
f \\
0
\end{bmatrix}
\]

where \( P_{ij} = t_j(x_i) \) for \( 1 \leq i \leq N, 1 \leq j \leq Q \). In [40] Micchelli proves the nonsingularity of such systems when \( \Phi \) is \( \mathbf{CPD}_m \), and the data set \( X = \{x_1, \ldots, x_N\} \subseteq \Omega \) is \( \mathcal{P}_{m-1}^d \)-nondegenerate. Common choices for \( \phi \) in this setting are given in [43] as
CHAPTER 1. INTRODUCTION

\[ r^\beta, \quad \beta \notin 2\mathbb{N}, \quad m \geq \lfloor \beta/2 \rfloor, \quad \text{polyharmonic spline,} \]

\[ r^\beta \log r, \quad \beta \in 2\mathbb{N}, \quad m > \beta/2, \quad \text{polyharmonic spline,} \]

\[ (c^2 + r^2)^{\beta/2}, \quad \beta \notin 2\mathbb{N}, \quad m \geq \lfloor \beta/2 \rfloor, \quad \text{generalised multiquadric.} \]

In [40], Schoenberg’s theorem is extended for conditionally positive definite functions.

**Theorem 1.9** A function \( \phi \) is conditionally positive definite of order \( m \) whenever \( \psi \) is completely monotone, where \( \psi(r) = (-1)^m \phi^{(m)}(\sqrt{r}) \).

Note that the equivalence is lost when compared to Theorem 1.5.

The polyharmonic spline with \( \beta = 2 \) was introduced as the thin plate spline by Duchon in [15]. The interpolant generated using this basis function minimises the bending energy in a thin plate of infinite extent clamped at the data points \((x_i, f(x_i))\) for \( i = 1, \ldots, N \).

The multiquadric resulted from the work of Hardy [28] in the application of representing topography from sparse, scattered data. The theory surrounding this function is dealt with in the often cited papers of Madych and Nelson [38, 39] which is an extension of Duchon’s work without the use of Fourier transforms. For a history of theoretical progress, see the introduction to [42].

A major aspect concerning the multiquadrics and gaussian basis function is their shape parameter dependence. Put simply, as the shape increases the approximation accuracy improves but the solution becomes harder to obtain numerically. There is no theory which gives the optimal shape parameter. Whilst intuition might suggest a dependence on point density or location, Carlson and Foley found in [10] that the optimal value for the shape parameter was most strongly influenced by the magnitude of the data values. The following method is given in [10] for computing an effective value for \( c \) when using the multiquadric to interpolate data given at points \((x_i, y_i, z_i)\). Find the maximum and minimum values for the
CHAPTER 1. INTRODUCTION

points \((x_i, y_i, z_i)\) for \(i = 1, \ldots, N\), and scale the data to the unit cube using

\[
\begin{align*}
x_i &= \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \\
y_i &= \frac{y_i - y_{\min}}{y_{\max} - y_{\min}} \\
z_i &= \frac{z_i - z_{\min}}{z_{\max} - z_{\min}}
\end{align*}
\]

where the subscripts are self explanatory. Compute the least squares bivariate quadratic polynomial fit to the scaled data \((\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)\). Let \(q(\tilde{x}_i, \tilde{y}_i)\) denote this quadratic function. Then the variance or average residual is given by

\[
V = \frac{\sum_{i=1}^{N} (\tilde{z}_i - q(\tilde{x}_i, \tilde{y}_i))^2}{N}.
\]

The rather empirical recommendation for the value of the shape parameter is given as

\[
c = \frac{1}{(1 + 120V)^2}.
\]

The reason for keeping a fixed shape parameter for all basis functions is so that the interpolation matrix will be symmetric and strictly (conditionally) positive definite. Section 1.2 gives further evidence of the difficulties of non-symmetric matrices. It is easy to find examples where even a small collection of differently shaped basis functions results in a singular system. Schaback has shown this experimentally, and concludes that no general theory is possible to guarantee nonsingularity; see [31]. The argument can be explained by considering an interpolation matrix \(A_y \in \mathbb{R}^N \times \mathbb{R}^N\) where \(y \in \mathbb{R}^2\) denotes the location of the \(N^{th}\) point. The function centred at \(x_i\) has shape parameter \(c_i\) such that the shapes are not the same at every point. By moving the \(N^{th}\) point in \(\mathbb{R}^2\) it is sometimes possible to find two values \(y_1\) and \(y_2\) such that

\[
\det(A_{y_1}) \cdot \det(A_{y_2}) < 0.
\]

Since \(\det(A_y)\) is a continuous function of \(y\), then there exists a \(y_*\) such that \(\det(A_{y_*}) = 0\).
However, using different shape parameters does improve the approximation. In [33] Kansa gives a method for predicting suitable values for \( c_i \) at each node \( x_i \). It has been found that exponential variations in \( c_i \) provided a better conditioned matrix and reduced errors. The shape parameter allocation is quite arbitrary since the points can be reordered. The user input comes from determining values \( c_{\text{min}} \) and \( c_{\text{max}} \), after which

\[
 c_i = c_{\text{min}} \left( \frac{c_{\text{max}}}{c_{\text{min}}} \right)^{(i-1)/(N-1)}. 
\]

When interpolating smooth data, the root mean square error surface when viewed as a function of \( c_{\text{min}} \) and \( c_{\text{max}} \) appears convex in the region of the minimum. This allows the use of standard minimisation routines to find optimum values for \( c_{\text{min}} \) and \( c_{\text{max}} \). Curiously, Kansa found that sequentially sorting the data was detrimental to the approximation error, and that a random ordering of points was advantageous. Such disregard for point density, and initial data values, indicates that improvements to this scheme are necessary.

### 1.2 Solving partial differential equations

This section is given to highlight further difficulties of non-symmetric matrices, with respect to solving partial differential equations (PDEs) by collocation. Radial basis functions have been used to solve a wide range of partial differential equations, with interesting applications in relativistic mechanics [13]. Consider the problem where \( L \) is a linear partial differential operator, and interpolatory conditions are prescribed on the boundary \( \partial \Omega \) of the domain \( \Omega \). The solution \( u \) must then satisfy

\[
 Lu = f \quad \text{in} \quad \Omega \subseteq \mathbb{R}^d \\
 u = g \quad \text{in} \quad \partial \Omega. 
\]

Let \( s_1 \) be an approximation to \( u \) of the form

\[
 s_1(x) = \sum_{i=1}^{N} \lambda_i \phi(\|x - x_i\|). 
\]
Assume there are \( b \) boundary points and \((N - b)\) interior points, which are ordered such that \( \{x_1, \ldots, x_{N-b}\} \in \Omega \setminus \partial \Omega \). For clarity it will be assumed that \( \phi \) is \( \text{PD} \) so that a polynomial need not be added to \( s_1 \). Then \( \lambda \) must be found such that

\[
L s_1(x_i) = f(x_i) \quad \text{for} \quad i = 1, \ldots, N - b
\]
\[
s_1(x_i) = g(x_i) \quad \text{for} \quad i = N - b + 1, \ldots, N.
\]

The resulting system of linear equations is of the form

\[
C \lambda = \begin{pmatrix} L \Phi \\ \Phi \end{pmatrix} \lambda = \begin{pmatrix} f \\ g \end{pmatrix}
\]

where \( L \Phi \) denotes the matrix \( (L \phi(\|x_i - x_j\|))_{1 \leq i \leq N - b, 1 \leq j \leq N} \), and \( \Phi = (\phi(\|x_i - x_j\|))_{1 \leq i \leq N - b + 1, 1 \leq j \leq N} \). The problem with this formulation is that the matrix \( C \) is clearly unsymmetric, and possibly singular. To circumvent this difficulty, the following approximation to the solution is proposed by Fasshauer [19]

\[
s_2(x) = \sum_{i=1}^{N-b} \alpha_i L \phi(\|x - x_i\|) + \sum_{j=1}^{b} \beta_j \phi(\|x - x_{N-b+j}\|).
\]

Therefore the arising system of equations is of the form

\[
E \lambda = \begin{pmatrix} L^2 \Phi & L \Phi \\ L \Phi & \Phi \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}.
\]

Now \( E \) is symmetric, and nonsingularity depends on the choice of \( \phi \). Experimentation has shown that there is little difference in the errors \( \|s_1 - u\|_\infty \) and \( \|s_2 - u\|_\infty \). For further theoretical treatment of such solutions to partial differential equations see Franke [23].

In [14] Dubal provides a domain decomposition iterative refinement method for obtaining multiquadric solutions to PDEs in 1D. The relatively poor results highlight the delicacies of solving PDEs by collocation. The boundary nodes can be viewed as anchoring the solution to fixed points, and the remaining nodes are required to communicate this information to the interior. Unlike pure interpolation, the clustering of points rarely brings local improvement. However, since the problem is set on a domain \( \Omega \) and not a finite number of points, the number of nodes employed by \( s \) is only limited by the solver.
1.3 Error analysis

It is necessary to be able to bound the approximation error \( (f - s) \) on some domain. Assume that the function \( f \) lies in a space \( \mathcal{G}_\Phi \) endowed with an inner product \( (\cdot, \cdot)_\Phi \), with null space \( \mathcal{P} \). Then there exists the following bound

\[
|f(x) - s(x)| \leq P(x)\|f\|_\Phi
\]  

(1.10)

where \( P \) is known as the power function, and \( \| \cdot \|_\Phi \) is the seminorm on \( \mathcal{G}_\Phi \). In [48] Schaback shows that by making further assumptions on \( f \), the error bound (1.10) can be improved by a factor of approximately two. The objective is to create a stronger seminorm \( \| \cdot \|_{\Phi^*} \) on a subspace \( \mathcal{G}_{\Phi^*} \) of \( \mathcal{G}_\Phi \), then

\[
|f(x) - s(x)| \leq P(x)\|P\|_{L_2(\Omega)}\|f\|_{\Phi^*}
\]

for \( f \in \mathcal{G}_{\Phi^*} \).

The aim of this section is to outline how the power function can be found explicitly in terms of the basis function \( \Phi \), and the location of data points. It will then be possible to find asymptotic bounds on \( P \), and therefore on the approximation error. It is also necessary to obtain a description of the native space \( \mathcal{G}_\Phi \) in terms of the Fourier transform \( \hat{\Phi} \).

The interpolation problem presented in Section 1.1 uses point evaluations \( \delta_j(f) = f(x_j) \) for \( j = 1, \ldots, N \) to construct an approximation. These interpolation conditions can be generalised by the action of linear functionals \( \lambda_j(f) \) for \( j = 1, \ldots, N \). Assume that these linear functions are continuous on \( \mathcal{G}_\Phi \). To implement this generalisation, consider the interpolation process \( S \) such that

\[
S(f)(x) = \sum_{j=1}^{N} u_{j,S}(x)\lambda_j(f)
\]  

(1.11)

where \( u_{1,S}, \ldots, u_{N,S} \) are functions on \( \Omega \). The aim is to determine the functions \( u_{j,S} \) such that the error \( \|f - S(f)\|_\Phi \) is minimised. The action of \( S \) should preserve
the null space under the recovery process, so

\[ S(p) = p \text{ for all } p \in \mathcal{P}. \]

Then for every \( f \in \mathcal{G}_\Phi \) there exists a pointwise error bound on the interpolant \( S(f) \) given by

\[ |f(x) - S(f)(x)| \leq P_u(x)\|f\|_\Phi \]

where \( P_u \) is the power function.

**Definition 1.13** The power function with respect to \( \Phi \) is given by

\[ P_u(x) = \sup_{\substack{f \in \mathcal{G}_\Phi \\
\|f\|_\Phi \neq 0}} \frac{|(f - S(f))(x)|}{\|f\|_\Phi} \in \mathbb{R} \cup \{\infty\}. \]

This amounts to finding the worst possible error when interpolating \( f \) by the process \( S \). It is essential to obtain \( P_u \) explicitly. Let \( \delta_{x,u,s} \) be the pointwise error evaluation functional vanishing on \( \mathcal{P} \) such that

\[ \delta_{x,u,s}(f) = f(x) - S(f)(x) = f(x) - \sum_{j=1}^{N} u_j(x)\lambda_j(f). \]

Then

\[ P^2_u(x) = \|\delta_{x,u,s}\|_\Phi^2. \]

The evaluation of \( \|\delta_{x,u,s}\|_\Phi \) requires the mapping \( F \)

\[ F(\delta_{x,u,s})(\cdot) = \Phi(x - \cdot) - \sum_{j=1}^{N} \lambda_j^2 \Phi(z - \cdot)u_j(x) \]

and the identity

\[ (\delta_{y,u,s},\delta_{x,u,s})_\Phi = \delta_{y,u,s}(F(\delta_{x,u,s})). \]

Then define \( P^2_u(x) = P_u(x, x) \) where

\[ P_u(x, y) = (\delta_{y,u,s},\delta_{x,u,s})_\Phi \]
\[ P_\delta(x, u, S) = \Phi(x - y) - \sum_{j=1}^{N} \lambda_j \Phi(z - \cdot) u_j(x) \]

\[ - \sum_{k=1}^{N} \lambda_k \Phi(x - z) u_k(y) + \sum_{j,k=1}^{N} \lambda_j \lambda_k \Phi(z - t) u_j(x) u_k(y) \]

for \( x, y \in \Omega \). The recovery process \( S \) uses the generalised interpolation conditions given by \( \lambda_j(f) \). In the setting of Section 1.1, where point evaluations of \( f \) are employed, the application of \( \lambda_j \) yields \( \lambda_j(f) = f_j \), therefore

\[ \lambda_j \Phi(z - y) = \Phi(x_j - y), \]

where the superscript indicates which argument the operator is applied to. Therefore

\[ P^2_\delta(x) = \Phi(0) - 2 \sum_{j=1}^{N} \Phi(x - x_j) u_j(x) + \sum_{j,k=1}^{N} \Phi(x_j - x_k) u_j(x) u_k(x) \quad (1.14) \]

for \( x \in \Omega \). Recall that the aim is to find functions \( u_j \) such that \( P_\delta \) is minimised. Differentiation of \( P^2_\delta \) with respect to \( u_k \) shows that the minimum is attained when \( u_k \) satisfies the Lagrange type interpolation conditions

\[ u_j(x_i) = \delta^i_j. \]

Let \( P_* \) denote the minimised power function. This representation can be used to obtain asymptotic error bounds in terms of the fill distance \( h \).

**Definition 1.15** The fill distance \( h \) is given by

\[ h = \sup_{x \in \Omega} \min_{1 \leq i \leq N} \| x - x_i \|_2. \]
CHAPTER 1. INTRODUCTION

<table>
<thead>
<tr>
<th>$\phi(r)$</th>
<th>$L_\infty$ bound on $\mathcal{P}_*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r^\beta$</td>
<td>$h^{\beta/2}$</td>
</tr>
<tr>
<td>$r^\beta \log(r)$</td>
<td>$h^{\beta/2}$</td>
</tr>
<tr>
<td>$(c^2 + r^2)^\beta$</td>
<td>$\exp(-\delta/h), \delta &gt; 0$</td>
</tr>
<tr>
<td>$e^{-cr^2}$</td>
<td>$\exp(-\delta/h^2), \delta &gt; 0$</td>
</tr>
<tr>
<td>$r^\nu K_\nu(r)$</td>
<td>$h^\nu$</td>
</tr>
</tbody>
</table>

Table 1.1: Bounds on the power function for small $h$.

**Definition 1.16** The maximum separation distance $q$ is given by

$$q = \frac{1}{2} \max_{x,y \in \Omega} \|x - y\|_2.$$  

It is assumed that the domain $\Omega \subset \mathbb{R}^d$ satisfies an interior cone condition, then $\mathcal{P}_*$ can be asymptotically bounded in $L_\infty$; see Table 1.1.

**Definition 1.17** A closed domain $\Omega \subset \mathbb{R}^d$ satisfies an interior cone condition if there is a fixed positive angle $\gamma$ and fixed height $\delta$ such that for any boundary point $x$ there is a cylindrical cone within $\Omega$ that has vertex $x$, angle $\gamma$ at the vertex, and height $\delta$.

A bound on $\mathcal{P}_*$ can further be established from the Fourier transform of $\Phi$. Let $\hat{\Phi}$ be the Fourier transform of $\Phi$ given by

$$\hat{\Phi}(w) = \int_{\mathbb{R}^d} \Phi(x)e^{-ix^T w} dx$$

and assume that

$$\hat{\Phi}(w) = \mathcal{O}(\|w\|_2^{-d-\rho}) \quad \text{as} \quad \|w\|_2 \to \infty. \quad (1.18)$$

The error bound (1.12) can then be written as

$$\|f - s\|_{L_\infty(\Omega)}^2 \leq C \cdot h^\rho \cdot \|f\|_\Phi^2 \quad (1.19)$$

for sufficiently small $h$; see [47] and [50]. The order of convergence with respect to $h$ is determined by the asymptotic behaviour of $\hat{\Phi}$ from (1.18).
CHAPTER 1. INTRODUCTION

It is also necessary to obtain a description of the native space $G_\Phi$. Every
$\text{CPD}_m$ basis function $\Phi$ is the reproducing kernel of a native space $G_\Phi$. This
native space can be described by

$$G_\Phi = \left\{ f : \hat{f}/\sqrt{\hat{\Phi}} \in L_2(\mathbb{R}^d) \right\}. \quad (1.20)$$

If $\Phi$ is $\text{PD}$ (that is $\text{CPD}_0$) then

$$||f||_\Phi = ||\hat{f}/\sqrt{\hat{\Phi}}||_2 \quad (1.21)$$

is the norm on $G_\Phi$. If $\Phi$ is $\text{CPD}_m$ for $m \geq 1$ then (1.21) is a seminorm on $G_\Phi$
with kernel $\mathcal{P}^d_{m-1}$.

Chapter 2 is concerned with the numerical implementation of this technique.
It will be seen that domain decomposition methods offer significant computational advantages. Error estimates obtained for such schemes require the use of polyharmonic splines as the basis functions. Existing error analysis is conducted in a native space generated by the Fourier transform of the basis function; see (1.20). If the basis function is very smooth, then the native space becomes restrictively small, and therefore limits the application of the error estimates. An approach is provided in Chapter 4 to create approximations, of finite continuity, to infinitely smooth basis functions. Use of these approximate basis functions effectively enlarges the native space. Such construction is possible as a result of the freedom to manipulate functions outside the domain on which the interpolation problem is set. This promotes the use of compactly supported functions, which are studied in Chapter 3.
Chapter 2

Computational methods

The difficulty of solving large linear systems which arise from radial basis function interpolation are discussed, and possible solutions are provided. Summaries of key algorithms in the field are given. An effective hierarchical domain decomposition iterative refinement method using inverse multiquadrics is then presented. Error analysis of this scheme however has not been possible. If the inverse multiquadric is replaced by a polyharmonic spline, then certain derivatives in the direct form of the new seminorm allow simple error bounds to be obtained. It is shown that the pointwise error bound decreases linearly at each level.

2.1 Introduction

As described in the previous chapter, a key aspect of obtaining an accurate interpolant is the successful solution of a system of linear equations. The choice of solver is determined by the size and conditioning of the system. The issue here is one of accuracy verses cost. Various iterative methods are available, which offer economical solutions, but often suffer from poor rates of convergence; see [9] and [34]. In PD cases conjugate gradient methods are possible, but only effective when the system is sparse. Gauss elimination however, is a computationally ex-
pensive direct method, which produces accurate results until machine precision becomes a problem. By taking $O(N^3)$ flops this is unacceptable for large interpolation problems. These difficulties motivate a search for improved computational procedures. The aim is to achieve the required accuracy at a reasonable cost.

One approach is to manipulate the interpolation matrix. This can be viewed as an instance of recasting the same interpolation problem in a more favourable way. The use of preconditioners is well described by Golub and Van Loan [25]. Let $\kappa(A)$ be the $L_2$ condition number of matrix $A$, obtained as the ratio of the smallest and largest eigenvalues of $A$. Let the linear problem be of the form $A\lambda = b$. A preconditioning matrix $S$ is sought such that

$$\kappa(SA) \ll \kappa(A)$$

then $SA\lambda = Sb$ is easier to solve than the original problem.

Interpolants using functions which look almost constant over the domain produce good approximations. Such systems can be so badly conditioned that preconditioning is not effective. For example, consider the configuration of points which lie on a finite regular grid $X$, and a slightly perturbed set $Y$. Let $A_X$ and $A_Y$ be the associated interpolation matrices respectively arising from inverse multiquadric interpolation. It was hoped that $A_X^{-1}$ would be a good preconditioner for $A_Y$. This preconditioner is only effective when the perturbations giving rise to $Y$ are small, or when the shape of the basis function is very narrow. This highlights the instability of some interpolation problems, which generally lead to good approximations if only a solution can be found. Schaback [46] explains this phenomenon by means of an “Uncertainty Relation” between upper bounds on errors for interpolants and lower bounds on the smallest eigenvalue of the interpolation matrix. An extreme case is when the basis function looks like the kronecker delta - easy to solve for, but which provides useless approximations. Alternatively, a constant basis function would yield an excellent approximation, but suffers from a singular interpolation matrix. These two opposing forces must
be played off against each other to achieve a successful approximation at reasonable cost. For example, the shape parameter in the gaussian and multiquadric cases can be adjusted to improve the diagonal weight of the matrix. The best approximation is usually found using the largest shape before machine precision breakdown occurs.

Rather than changing the shape of the basis function, applications of the discretised Laplace operator can improve the matrix conditioning. In [16] Dyn and Levin note that in the case of the multiquadric and polyharmonic splines

$$\lim_{r \to \infty} \Delta^m \phi(r) = 0$$

and

$$\lim_{r \to 0} \Delta^m \phi(r) = \begin{cases} \infty & \text{for polyharmonic splines} \\ \gg 1 & \text{for multiquadrics} \end{cases}$$

where $\Delta^m$ is the $m$-iterated Laplacian. A discrete analogue of the Laplacian is applicable on a regular $N \times N$ grid. The basis function at each interior node is replaced by a linear combination of its nearest neighbours. When $m = 1$, the new basis function $\psi$ at location $x_{i,j}$ for $2 \leq i, j \leq N - 1$ is given by

$$\psi_{i,j} = -4\phi_{i,j} + \phi_{i-1,j} + \phi_{i+1,j} + \phi_{i,j-1} + \phi_{i,j+1}$$

where $\phi_{i,j}$ is the original basis function centred at $x_{i,j}$. The resulting $\psi$ is much steeper than $\phi$ and looks peaked, thus increasing the weight of the interpolation diagonal. An iterative method for solving $A\lambda = b$ is then given where most of the rows in $A$ are diagonally dominant.

Work done by Beatson, Light and Billings [3] shows that the linear system arising from $\text{CPD}_m$ functions can be rewritten such that a new matrix has much improved qualities. If $\Phi$ is $\text{CPD}_m$, then a reproducing Hilbert space $\mathcal{H}$ naturally arises. By the Riesz Representation Theorem, there exists a unique reproducing kernel $K$ such that $f(y) = \langle f, K(\cdot, y) \rangle$ for all $f \in \mathcal{H}$ and $y \in \mathbb{R}^n$. This kernel can
be written explicitly thus

\[ K(x, y) = \Phi(x, y) - \sum_{i=1}^{Q} p_i(x)\Phi(x_i, y) - \sum_{j=1}^{Q} p_j(y)\Phi(x, x_j) \]

\[ + \sum_{i=1}^{Q} \sum_{j=1}^{Q} p_i(x)p_j(y)\Phi(x_i, x_j) + \sum_{i=1}^{Q} p_i(x)p_i(y) \]

for \( x, y \in \mathbb{R}^n \). Here \( \{p_1, \ldots, p_Q\} \) is the Lagrange basis for \( \mathbb{P}^n_{m-1} \) with respect to the points \( \{x_1, \ldots, x_Q\} \), which are also assumed to be \( \mathbb{P}^n_{m-1} \)-nondegenerate. It is then shown that \( \{K(\cdot, x_1), \ldots, K(\cdot, x_N)\} \) is a basis for the space containing interpolants of the form (1.6), and therefore interpolation can be accomplished via

\[ \sum_{i=1}^{N} \nu_i K(x_i, x_j) = f(x_j) \quad \text{for} \quad 1 \leq j \leq N. \]

The ensuing matrix \((K(x_i, x_j))_{1 \leq i, j \leq N}\) is positive definite and therefore offers a potentially favourable alternative to the original problem. Unfortunately this recasting has an undesirable scale dependence property. To overcome this, a new basis

\[ \{p_1, \ldots, p_Q, H(\cdot, x_{Q+1}), \ldots, H(\cdot, x_N)\} \]

is created where

\[ H(x, y) = K(x, y) - \sum_{i=1}^{Q} p_i(x)p_i(y) \quad \text{for} \quad x, y \in \mathbb{R}^n. \]

This formulation yields the following linear system

\[
\begin{pmatrix}
I & 0 \\
E^T & C
\end{pmatrix}
\begin{pmatrix}
c \\
\gamma
\end{pmatrix} = d
\]

(2.1)

where \( I \) is the \( Q \times Q \) identity, \( C \) is the matrix \((H(x_i, x_j))_{Q+1 \leq i, j \leq N}\), and \( E^T \) is the matrix \((p_j(x_i))_{i,j}\) where \( Q + 1 \leq i \leq N \) and \( 1 \leq j \leq Q \). Denote the vectors \((c_1, \ldots, c_Q)^T\), \((\gamma_{Q+1}, \ldots, \gamma_N)^T\) and \((d_1, \ldots, d_N)^T\) by \( c, \gamma \) and \( d \) respectively. The key here is that \( C \) can be shown to be strictly positive definite, so that (2.1) is solved by setting \( c = (d_1, \ldots, d_Q)^T \), leaving

\[ C\gamma = (-E^T I)d \]
to be tackled with Cholesky decomposition.

An outright change of basis is offered by compactly supported PD functions. These functions can yield good approximations whilst having a matrix which is sparse, allowing conjugate gradient methods to be effective. The computational gain is usually at the expense of approximation power and smoothness. Methods for obtaining compactly supported functions with specific continuity are discussed in Chapter 3. The notion of compactly supported functions brings a localisation of the problem. Refinement to a coefficient at a particular point will not disturb the approximation outside the support of the basis function. An extreme case of this is demonstrated by Schaback’s Greedy Method [49] where the current residual is evaluated, and the node with the greatest error is adjusted to interpolate the data exactly. Wendland polynomials [56] are used as the basis function, with decreasing support as the method progresses. Such a process can continue until the effects of the refinements cause more error elsewhere than that which is being refined. This method converges slowly, after some good progress in the first few iterations.

Multilevel iterative refinement processes offer another method of circumventing the problem of ill conditioning. Consider a sequence of subsets $X_k \subseteq X = \{x_1, x_2, \ldots, x_N\}$ each containing $N_k$ points. Note that these subsets need not be nested. At the $k^{th}$ level, a refinement to the approximation is made by interpolating the error of the previous level $f - (s_1 + s_2 + \cdots + s_{k-1})$ at the points in $X_k$ with $s_k$. Formally

\[
\begin{align*}
  s_1|_{x_1} &= f|_{x_1} \\
  s_2|_{x_2} &= (f - s_1)|_{x_2} \\
  & \vdots \\
  s_k|_{x_k} &= (f - \sum_{i=1}^{k-1} s_i)|_{x_k}.
\end{align*}
\]
For clarity, let the error at each level be

\[ e_k = f - \sum_{j=1}^{k} s_j. \]

Then \( s_k \) interpolates \( e_{k-1} \) on \( X_k \).

Iterative refinement is central to the work of Floater and Iske [20]. Compactly supported radial basis functions are used at each level, with reducing support as the algorithm progresses. The exact support at each level is determined by the uniformity of the data. The philosophy of this approach is that the main features of the data \( f \) are captured at the outset, and finer details are resolved as the levels increase. A method for ordering the points into the set \( Y = \{ y_1, \ldots, y_N \} \) is given such that for any \( N_k \), the subset \( X_k = \{ y_1, \ldots, y_{N_k} \} \) is optimally uniform; see [32] for details. The uniformity is measured as the ratio between the minimum separation distance and the radius of the largest empty sphere. These quantities are easily obtained from the Voronoi diagram; see [6] and [12]. Figure 2.1 shows the Voronoi diagram for the set of 50 random points \( X = \{ x_1, \ldots, x_N \} \) on the unit square. Each polygon centred at \( x_i \) contains all of the points which are closer to \( x_i \) than any other node. Two points are called Voronoi neighbours if they share an edge of a polygon. Figure 2.2 shows the Delaunay triangulation of the same set \( X \), where Voronoi neighbours are connected. Delaunay triangulation takes \( O(N \log N) \) operations, and due to the continual updating required in [20], the final algorithm is \( O(N^2) \). It seems strange that such effort is expended for thinning the data before any interpolation takes place.

Data thinning is an obvious way of overcoming excessively large systems. If the initial data is relatively smooth, then much of the information can be discarded without detriment to the approximation. A purely random selection may not prove sufficient since this cannot guard against two points being selected which are very close to each other. The precision of the thinning process in [20] however is not necessary for the work presented in Section 2.2.

The thin plate spline has certain localisation properties which allow an effect-
Figure 2.1: Voronoi diagram of $X$.

ive use of iterative refinement as described by Powell in [44]. A global solution is required of the form

$$s(x) = \sum_{i=1}^{N} \lambda_i \|x - x_i\|^2 \log(\|x - x_i\|_2) + \text{linear polynomial}.$$  

The ordering of the points $\{x_1, \ldots, x_N\}$ is crucial to the success of this algorithm. The points $\{x_{N-2}, x_{N-1}, x_N\}$ are chosen such that they lie on the boundary of the convex hull of the interpolation points, and are as far apart from each other as possible. Let $k$ take the values in the sequence $1, 2, \ldots, N - q$, where $q = 30$ is a typical value. The freedom to reorder points $x_j$ for $j = k, k+1, \ldots, N-3$ is then
used to minimise the numbers \( \{\|x_j - x_k\|_2 : j = k + 1, k + 2, \ldots, N\} \). So for any given point, the remaining nodes are reordered such that the following member in the sequence is the closest available neighbour. Let \( \mathcal{L}_k \subset \{k, k + 1, \ldots, N\} \) be the set containing the indices of the next \( q \) points which follow the \( k^{th} \) node in the ordered sequence. Now let

\[
\chi_k(x) = \sum_{j \in \mathcal{L}_k} \nu_{kj}\phi(\|x - x_j\|_2) + \text{linear polynomial}
\]

be the thin plate spline solution to the Lagrange conditions

\[
\chi_k(x_i) = \delta_i^k \quad \text{for} \quad i \in \mathcal{L}_k.
\]


Each coefficient $v_k$ involves only $q$ equations, and is calculated explicitly before the algorithm commences. Refinement of the approximation comes from considering the $k^{th}$ point, where the value

$$
\mu_k = \sum_{j \in \mathcal{C}_k} \lambda_{kj} (f(x_j) - s(x_j))
$$

is obtained and the coefficient $\lambda_k$ is replaced by $\lambda_k + \mu_k$. The reason for this technique is that $\mu_k$ is the coefficient of $\phi(||x - x_k||_2)$ in the thin plate spline function that interpolates the residuals $(f(x_i) - s(x_i))$ for $i \in \mathcal{L}_k$. So if the errors at all points $x_i$ for $i \in \mathcal{L}_k$ were matched by the cluster of thin plate splines $\phi(||x - x_i||_2)$ for $i \in \mathcal{L}_k$, then $\mu_k$ would be the coefficient ascribed to $\phi(||x - x_k||_2)$. This therefore makes $\mu_k$ a good candidate for an update to the current coefficient.

In one iteration, the values for $\lambda_k$ are updated in this way for $k = 1, \ldots, N - q$. Since this technique cannot be used for the final $q$ points, the last stage is to adjust the approximation by interpolating the current residual over the final $q$ points directly. This explains the necessity of the last 3 points being well spread; to give a global influence to the final stage.

Instead of tackling the global problem which may consist of a large number of points, the domain can be decomposed into smaller systems and solved individually. This approach can heavily exploit parallel processing since the subdomains may not need to communicate during the solve. Once all the subinterpolation problems have been solved, if necessary the tiles can be patched together with some smoothing operator at the boundaries to ensure a specific degree of continuity. However, if the tiles are specified to overlap, then evaluating the approximation at particular points does not require the construction of a global solution.

There may be instances when a complete global solution is required, possibly if the interpolant is to have some differential operator applied to it. Domain decomposition has been used to iteratively obtain a solution using globally supported functions; see Beatson et al [3]. The original domain is divided into panels
$P_1, P_2, \ldots, P_k$ and an interpolation problem is set on each panel. The coefficients $\lambda_1, \lambda_2, \ldots, \lambda_k$ obtained on each panel are then used as an estimate for the coefficients of the whole system. Since the basis functions are globally supported then the first iteration obviously bears little resemblance to the initial data. For example, the accuracy obtained on panel $P_1$ is destroyed by the influence of the surrounding panels in the evaluation. This is remedied by taking a few points from each panel and performing global iterative refinement once again. The complete process is repeated and the coefficients continually updated until a specified tolerance has been reached. Such a method has been demonstrated to solve problems with $10^4$ points in seconds.

A critical aspect of this algorithm is the ability to quickly evaluate the interpolant at numerous points. By direct calculation this procedure takes $O(N)$ operations, but fast-evaluation methods can accomplish this in $O(1)$. From Beaton [4], consider the weighted sum $E_y$ of a cluster of thin plate splines at points centred on $y$. By using a far-field expansion, an approximation for $E_y(x)$ can be obtained for large $||x - y||_2$. This approximation can then be expanded as a Taylor series, where only the first $n$ terms are required to estimate $E_y(x)$ to within a specified tolerance. Other moment based fast-evaluation methods are available for a wider range of basis functions; see [5].

A theoretical foundation for multilevel methods is given by Narcowich, Schaback and Ward [42]. Inspired by the numerics of Floater and Iske, the theory requires that basis functions of decreasing smoothness are used at each level. Consider the problem of finding the best approximation $f^* \in W$ to a given $f \in W$, where $W$ is a closed linear subspace of the normed linear space $W$. This is equivalent to solving the following minimisation problem

$$
\inf_{g \in W} \|f - g\|_W = \|f - f^*\|_W.
$$

To obtain a useful error bound, the function $f$ must be restricted to a subspace $W_0$ of $W$, such that the norm associated with $W_0$ carries more information concerning
There exists a bound of the form
\[ \|f - f^*\|_W \leq K(W, W_0)\|f\|_{W_0} \]
with the constant \( K \) being significantly less than one. The multilevel approach is stimulated by wanting to improve the above error bound, but with a reluctance to change \( W_0, W, W \) or \( f \). The technique is to use a sequence of nested spaces
\[ W_0 \subset W_1 \subset W_2 \cdots \subset W_m = W \]
that connect \( W_0 \) and \( W \). At each level the task is to approximate \( f_k \in W_k \) from an element in \( W_k \subset W_k \), where \( f_k \) is the residual of the previous level. Therefore
\[ f_k = f_{k-1} - f_{k-1}^* \in W_{k-1} \text{ for } 2 \leq k \leq m \text{ and } f_1 = f. \]
Then the final approximation at level \( m \) is given by
\[ g_m^* = \sum_{k=1}^{m} f_k^* \]
and is in the space
\[ V_m = \bigcup_{k=1}^{m} W_k. \]
As the approximation problems are solved in each subspace \( W_k \), the error bound (2.2) is applicable thus
\[ \|f_k - f_k^*\|_{W_k} \leq K_k\|f_k\|_{W_{k-1}} \text{ for } 1 \leq k \leq m \]
where the constant \( K_k = K(W_k, W_{k-1}, W_k) \). This error bound can then be applied recursively so that
\[ \|f - g_m^*\|_W = \|f_1 - g_m^*\|_{W_m} \leq \left( \prod_{k=1}^{m} K_k \right) \|f_1\|_{W_0}. \]
Since the constants \( K_k \) are expected to be less than one, the benefit of a multilevel approach is evident when
\[ \left( \prod_{k=1}^{m} K(W_k, W_{k-1}, V_k) \right) < K(W_m, W_0, V_m). \]
The subspaces $\mathcal{W}_k$ are determined by the smoothness of the function $f_k$. An interesting example is given where cubic splines are used to approximate $f$ and then the residual is refined by piecewise linears. Conditions are given for when this particular multilevel approach provides an improvement over the single step method.

Hartmann also provides some work on the linear decay of error between levels of a stationary multilevel process; see [29]. A bandlimiting operator is introduced by

$$\mathcal{F}_d(B_1g)(w) = \begin{cases} \mathcal{F}_d(g)(w) & \text{for } w \leq \frac{1}{h}C \\ 0 & \text{for } w \geq \frac{1}{h}C \end{cases}$$

where $C = [-\pi, \pi]^d$. If basis functions $\phi_k$ are used at the $k^{th}$ level, which induce a (semi)norm $\| \cdot \|_k$, then the error of the current approximation is given by

$$\|f - s_0 - \cdots - s_k\|_k \leq \left\{ h^{-d/2} \left( \frac{2}{\sqrt{K}} + \frac{\sqrt{C}}{\sqrt{L}} \right) \right\}^k \cdot \left[ 2\|f - B_0f\|_0 + \sqrt{C}\|B_0f\|_0 \right]$$

where $h$ is the point separation, with constants $C, K$ and $L$ depending on the basis function $\phi$. It then remains to show that the term

$$\left\{ h^{-d/2} \left( \frac{2}{\sqrt{K}} + \frac{\sqrt{C}}{\sqrt{L}} \right) \right\}$$

is significantly less than unity for a selection of basis functions. Examples are given for the gaussian, multiquadric and Wendland polynomials. In the latter case however, the value of (2.3) is given as 0.9911, suggesting that further work is required.

### 2.2 Hierarchical refinement algorithm

A method of finding local approximations is used to thin data before a hierarchical iterative refinement scheme is employed in conjunction with domain decomposition. The interpolation problem on each subdomain is solved using the same
stored inverse. The approximation power of the inverse multiquadric is exploited whilst overcoming the computational difficulties associated with globally supported basis functions.

The interpolation problem is given on a set of points \( X = \{x_1, \ldots, x_N\} \) with data \( f(x_i) \). An interpolant is constructed of the form (1.3), using the inverse multiquadric basis function. This parameter dependent function is good at approximating data for certain values of \( c \), but these cause inherent ill conditioning in \( A \). Large values of \( c \) achieve good initial approximations to smooth data, whilst smaller values produce functions capable of resolving fine detail. Ideally, such properties could be exploited without having to solve the whole system directly. This statement is not restricted to inverse multiquadrics, but these are used throughout due to their approximating power.

Since \( s \) is evaluated at points \( y \neq x_i \) where an approximation is required, a global solution incorporating all \( N \) centres may be inappropriate. Further, it is unnecessary to find \( \tilde{\lambda} \) such that \( \|f - A\tilde{\lambda}\|_{\infty} \ll |f(y) - s(y)| \), since the accuracy of \( s(y) \) is limited by the approximating power of \( \phi \). Rather than searching for a complete global solution, this shows that attention may be focused on small regions around evaluation points. Moreover, the aim is to obtain a solution such that the residual from the interpolation equations and approximation accuracy are comparable. For little is to be gained by having a small residual, while the approximation power of the basis functions limits the final accuracy. For example, an extremely small residual for a piecewise linear approximation to some smooth function is wasted since the reproduction quality is poor.

In Section 2.2.1, local approximations are used to convert irregular data to a regular mesh of approximate function values. Whilst the method can be generalised to \( \mathbb{R}^d \), the description and examples are given in \( \mathbb{R}^2 \). The system of equations associated with the gridded data is inverted and used to solve subsequent systems.
Section 2.2.2 describes the present hierarchical iterative refinement algorithm, and explains the computational advantages of domain decomposition and the use of a stored inverse.

2.2.1 Local solutions and gridding data

A smooth data set can be significantly thinned whilst retaining general information about its behaviour. A method of finding uniform approximate data is presented.

An approximation to $f$ at a point $y \in \mathbb{R}^2$ is achieved by solving a small interpolation problem centred on $y$. The closest $q$ points in $X$ to $y$ are interpolated by inverse multiquadrics with shape parameter $c_{\text{local}}$, and evaluated at $y$. Since $q$ can be as low as 20, $c_{\text{local}}$ can be relatively large before the matrix ill-conditioning becomes unacceptable, thus yielding a good approximation. This method is highly parallelizable, and large data sets can be managed without the need for assembling the interpolation matrix.

The task of finding the $q$-nearest neighbours in the set $X = \{x_1, \ldots, x_N\}$ requires some attention. Note that in the given localisation procedure, the $q$ chosen points do not have to be precisely the nearest neighbours, but it is advantageous. The most elementary approach, used by Powell [44] in the thin plate spline algorithm, is to calculate the distance between every pair of points. Although this $O(N^2)$ operation can be computed by parallel processors, a more efficient technique is required. The $O(N \log N)$ Delaunay triangulation has potential since the immediate neighbours are immediately identified. Structures such as the nD-tree can also be invoked to find all points within a given rectangular box, which can then be easily extended to finding $q$ neighbours; see [53].

Finding the optimum shape parameter on regular or scattered data remains an open problem, as shown in [26]. There is no obvious correlation between point spacing and a good choice of $c_{\text{local}}$. The best shape parameter is generally found
by increasing the value of $c_{local}$ until just prior to machine precision breakdown.

This localisation approach can now be employed to thin data. Let $Y = \{y_1, \ldots, y_{n^2}\}$ be the set of $(n \times n)$ regular grid points. If the previous local approximation technique is applied to each $y_i$, then the irregular data can be transformed to a regular grid with approximate function values $\tilde{f}(y_i)$. The aim is now to find a global approximation using the new data at the grid points.

After converting scattered data to a regular grid, certain approximation techniques become available which would otherwise have been difficult to implement. Polynomial tensor product splines can be efficiently employed to approximate a solution from the given gridded data. To find such an approximation at a point $z$, then $z$ must lie inside a $(d + 1) \times (d + 1)$ subgrid of the regular points, where $d$ is the degree of the Lagrange polynomials to be used. Let the points of such a subgrid be labelled $\xi_{ij}$ and have function values $\tilde{f}_{ij}$ for $i, j = 1, \ldots, d + 1$. The univariate Lagrange polynomials $L_i$ and $L^j$ are constructed such that

$$L_i(\xi_k) = \begin{cases} 1 & \text{for } k = i \\ 0 & \text{for } k \neq i \end{cases}$$

$$L^j(\xi_k) = \begin{cases} 1 & \text{for } k = j \\ 0 & \text{for } k \neq j. \end{cases}$$

The polynomial tensor product spline $\varphi_{ij}$ is defined to be

$$\varphi_{ij}(z) = L_i(z) \cdot L^j(z).$$

The approximation at $z$ is given by

$$\sum_{i=1}^{d+1} \sum_{j=1}^{d+1} \varphi_{ij}(z) \tilde{f}_{ij}.$$ 

Alternatively, a global interpolant to the approximate data $\tilde{f}(y_i)$ can be constructed thus

$$s(x) = \sum_{i=1}^{n^2} \mu_i \phi(||x - y_i||),$$
where \( s(y_i) = \tilde{f}(y_i) \) for \( i = 1, \ldots, n^2 \). This leads to the following system of linear equations

\[
B\mu = \tilde{f},
\]

where \( B_{i,j} = \phi(||y_i - y_j||) \), \( \mu = [\mu_1 \cdots \mu_{n^2}]^T \), and \( \tilde{f} \) is the vector containing the approximate function values \( \tilde{f}(y_i) \) for \( i = 1, \ldots, n^2 \). This amounts to finding an interpolant \( s \) to a thinned approximation of the initial data. The local approximation errors \( |f(y_i) - \tilde{f}(y_i)| \) limit the final accuracy of \( s \) when compared to the original data \( f \).

The inverse of \( B \) need only be computed once, and then stored for future use. All scattered data problems can then be scaled and transformed to the regular grid \( Y \), whereupon \( \mu \) is given by the matrix–vector product \( \mu = B^{-1}\tilde{f} \). Only half of the entries of \( B^{-1} \) need to be stored since \( B^{-1} = (B^{-1})^T \).

If the matrix \( B \) is too large then storage becomes a problem, and it is difficult to calculate the inverse. As \( c \) increases the approximation improves, but the residual \( \|\tilde{f} - B\mu\|_\infty \) grows. A value for \( c \) is chosen before the approximation begins to deteriorate, due to the rise in the residual.

**Example 2.5**

The function \( f \equiv 1 \) is approximated on the unit square using inverse multiquadrics by interpolating \( \tilde{f}_i = 1 \) for \( i = 1, \ldots, 400 \) using (2.4). The approximation is evaluated at 1000 random points. The results in Figure 2.3 are typical for smooth functions, but the consequent choice of \( c \) is only a guide and does not guarantee success for all \( f \).

## 2.2.2 Hierarchical iterative refinement

Hierarchical iterative refinement is employed as described in Section 1.1. The technique of gridding data in Section 2.2.1 is applied to the data sets \( X_k \) to find
approximate function values at regular points. Then the current approximation $s_k$ is updated via $s_{k+1} = s_k + t_k$ where $t_k$ interpolates the current residual on the regular grid. This interpolant of the residual is given by

$$t_k(x) = \sum_{i=1}^{N_k} \gamma_i \phi_k(||x - y_i||)$$

(2.6)

where $\phi_k(r) = (c_k^2 + r^2)^{-1/2}$. The value $c_1$ can be relatively large to give a good initial approximation. As $N_k$ increases, $c_k$ has to be reduced to ensure computational solvability. The decrease in $c_k$ introduces tighter basis functions which improve the resolution of the approximation.

The dense systems arising from (2.6) have to be solved directly, but this is impractical for large $N_k$. To overcome such complications, domain decomposition is applied to each $X_k$.

The levels of the hierarchy have to be computed sequentially, but by using
domain decomposition each subdomain can be dealt with in parallel. Moreover, each solution only requires a single matrix–vector product.

To put this in the current context, each $X_k$ is covered by $m_k$ overlapping domains $\alpha_k Y$, where $\alpha \in \mathbb{R}$ and $Y$ consists of $(n \times n)$ regular points on $[0, 1]^2$. Each subdomain $\alpha_k Y$ consists of an inner region, where the approximation is finally evaluated, and an overlap; see Figure 2.4. Special attention has to be given to subdomains whose edges coincide with the boundary of $X$; see Figure 2.5.

![Figure 2.4: Anatomy of typical interior subdomain.](image)

At the $k^{th}$ level, $m_k$ subdomain interpolation problems need to be solved. Since $B$ is invariant under shifts and rotations of the centres $y_i$, the stored $B^{-1}$ can be invoked. If the centres are scaled $y_i \mapsto \alpha y_i$, this amounts to a change in
CHAPTER 2. COMPUTATIONAL METHODS

the shape parameter. Recall

\[ B_{1 \leq i,j \leq n} = \phi(\|y_i - y_j\|) \text{ where } y_i \in [0, 1]^2. \]

Now

\[ \phi(\|y - y_i\|) = (c^2 + \|y - y_i\|^2)^{-1/2} \]
\[ = \alpha(\alpha^2 c^2 + \|\alpha y - \alpha y_i\|^2)^{-1/2}. \]

Let \( w_i = \alpha y_i \) and define

\[ \psi(\|w - w_i\|) = \alpha(\alpha^2 c^2 + \|w - w_i\|^2)^{-1/2}. \]

Then

\[ B_{1 \leq i,j \leq n} = \phi(\|y_i - y_j\|) = \psi(\|w_i - w_j\|) \text{ where } w_i \in [0, \alpha]^2. \]

Therefore by using the matrix \( B \), a new inverse multiquadric is created at scaled points with shape parameter \( \alpha c \).

Each of the thinned global interpolation problems (2.6) can be decomposed and solved by multiple applications of the stored inverse \( B^{-1} \). Continued use of the same inverse naturally introduces tighter basis functions suitable for approximating typical residuals.
2.2.3 Pseudocode

Iterative refinement is performed on subsets $X_k \subseteq X$ as follows.

for $k = 1, 2, \ldots$

Divide $X_k$ into $m_k$ scaled patches $\alpha_k Y$.

for $p = 1, \ldots, m_k$.

Perform localisation on $\alpha_k Y$ to obtain grid points $\alpha_k y_i$.

Calculate residual at points $\alpha_k y_i$.

Interpolant residual on $\alpha_k Y$ with shape parameter $\alpha_k c$.

Solve interpolation problem using stored inverse $B^{-1}$.

Update total approximation $s_k$ restricted to patch $\alpha_k Y$.

end

end

Note that $s_k$ is difficult to express meaningfully since it consists of a variety of scaled compactly supported functions.

2.2.4 Numerical results

Example 2.7

An example is given where the above scheme is used to approximate Franke's function $f$ from $10^4$ scattered points in the unit $\mathbb{R}^2$ domain; see [22]

$$f(u, v) = 0.75e^{-0.25(9u-2)^2 - 0.25(9v-2)^2} + 0.75e^{-(9u-2)^2/49-(9v-2)^2/10} + 0.5e^{-0.25(9u-7)^2 - 0.25(9v-3)^2 - 0.2e^{-(9u-4)^2-(9v-7)^2}}.$$

The localised interpolation problems are solved directly using Gauss elimination, with the number of nearest neighbours $q = 20$, and $c_{local} = 0.2$. The square subdomain grids $Y$ comprise of $21 \times 21$ equally spaced points. For ease of implementation, the subdomains used for a particular level are of equal size. The overlaps between subdomains therefore consist of one or two mesh points, depending on position. The key interpolation matrix $B$ is constructed from inverse
Table 2.1: Results of Example 2.7.

<table>
<thead>
<tr>
<th>Level</th>
<th>No. of domains $m_k$</th>
<th>Shape parameter $c_k$</th>
<th>Overlap</th>
<th>Max. error in data at grid points $|f - \tilde{f}|_\infty$</th>
<th>Max. error in solution $|f - s_k|_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0.25</td>
<td>0</td>
<td>$8.241 \times 10^{-5}$</td>
<td>$1.371 \times 10^{-4}$</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0.138</td>
<td>1/36</td>
<td>$2.424 \times 10^{-6}$</td>
<td>$1.838 \times 10^{-5}$</td>
</tr>
<tr>
<td>3</td>
<td>16</td>
<td>0.0688</td>
<td>1/72</td>
<td>$3.189 \times 10^{-6}$</td>
<td>$3.189 \times 10^{-6}$</td>
</tr>
</tbody>
</table>

multiquadrics with $y_i \in [0, 1]^2, c = 0.25$, and $B^{-1}$ is generated using Matlab. The domain decomposition is straightforward on the unit square with $m_k = 4^{k-1}$; see Figure 2.5 for $m_3$. The thinned global interpolants $s_k$ are evaluated at $161^2$ points $t_i \in [0, 1]^2$. Figures 2.7, 2.9 and 2.11 show the error in the approximated data at the regular grid points $|f(y_i) - \tilde{f}(y_i)|$, and Figures 2.8, 2.10 and 2.12 show the error in the approximation $|f(t_i) - s_k(t_i)|$ for each level. The results are also given in Table 2.1.
The error from Level 1 clearly demonstrates the ability of the inverse multi-quadric to approximate smooth data. The error near the boundary is scaled by an order of magnitude at each level, but has the same general behaviour. The final iteration leaves error near the boundary, aggravated by test points being outside the original scattered data set. Such evaluation points ought to be included since, although they require an extrapolation of $s_k$ to evaluate, the experiment was specified to be conducted on the unit domain. The original aim of finding a solution where the residual is comparable to the approximation accuracy is fulfilled at Level 3.

Figure 2.7: Example 2.7 Level 1 local approximation error.
Figure 2.8: Example 2.7 Level 1 global approximation error.

Figure 2.9: Example 2.7 Level 2 local approximation error.
Figure 2.10: Example 2.7 Level 2 global approximation error.

Figure 2.11: Example 2.7 Level 3 local approximation error.
Figure 2.12: Example 2.7 Level 3 global approximation error.
Table 2.2: Results for Example 2.7 with polynomial splines.

Example 2.8

Example 2.7 is repeated as far as the regularization of data, then polynomial tensor product splines are used to find the final approximation, as described in Section 2.2.1. Such splines cannot replace the inverse multiquadric approximation on the regular grid without an increase in error. Such an error is then propagated to the next level where the discrepancy is amplified. However, if the hierarchical refinement procedure is abandoned, then these basis functions efficiently yield good approximations. Table 2.2 shows the approximation accuracy for such splines of different polynomial degree without iterative refinement. The grid sizes are comparable to those used in Example 2.7.

Example 2.9

To demonstrate the implementation of the scheme on more complex domains, the Peaks function \( f \) given at 7500 random points in an L-shaped domain is approximated.

\[
f(u, v) = 3(4 - 6u)^2e^{-(6u - 3)^2-(6v-2)^2} - 1/3e^{-(6u-2)^2-(6v-3)^2} \\
-10((6u - 3)/5 - (6u - 3)^3 - (6v - 3)^5)e^{-(6u-3)^2-(6v-3)^2}.
\]

There are many ways of decomposing an L-domain into combinations of squares and rectangles. In order that a stored inverse can be used repeatedly, the subdomains must be similar to the original domain; see Figures 2.14 and 2.15.
The parameters are the same as Example 2.7, but the subdomains are now L-shaped without overlaps. Each subdomain is formed by removing the appropriate points from a $21 \times 21$ regular square grid. The inverse matrix obtained from the first level is stored and used to compute the solutions in Level 2. Figures 2.16 and 2.17, and Table 2.3 show the relative error of the approximation at each level.
CHAPTER 2. COMPUTATIONAL METHODS

Figure 2.14: Level 1 L-domain.

Figure 2.15: Level 2 L-domain decomposition.

Figure 2.16: Example 2.9 Level 1 global approximation error.
Figure 2.17: Example 2.9 Level 2 global approximation error.
Example 2.10

It would be ideal if one configuration of grid points could be successfully implemented to thin all scattered data interpolation problems. However, this example demonstrates the difficulty of the scheme in its most general setting. The previous example is repeated with the alteration that the domain decomposition invokes the square subdomains from Example 2.7. This requires approximate function values to be found for points well away from any initial data. Even though the extensions to the given function are smooth, the accuracy can be severely impaired. The reduced accuracy after Level 1 is to be expected due to a quarter of the data on the regular grid being fictitious. The Level 2 decomposition is clearly suited to the L-domain, where only 3 square panels are required. Table 2.4 and Figures 2.18 and 2.19 shows the global approximation error for each level.
Figure 2.18: Example 2.10 Level 1 global approximation error.

Figure 2.19: Example 2.10 Level 2 global approximation error.
CHAPTER 2. COMPUTATIONAL METHODS

<table>
<thead>
<tr>
<th>$c_{\text{local}}$</th>
<th>Local error</th>
<th>Global error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>on square</td>
<td>on disc</td>
</tr>
<tr>
<td>0.2</td>
<td>$1.7 \times 10^{-1}$</td>
<td>$1.2 \times 10^{-2}$</td>
</tr>
<tr>
<td>0.3</td>
<td>$1.1 \times 10^{-1}$</td>
<td>$4.9 \times 10^{-3}$</td>
</tr>
<tr>
<td>0.4</td>
<td>$3 \times 10^{-2}$</td>
<td>$1.2 \times 10^{-3}$</td>
</tr>
</tbody>
</table>

Table 2.5: Changing $c_{\text{local}}$ in Example 2.11.

Example 2.11

To further demonstrate the problem of grid points outside the original domain, Frankes function is given at 10000 points now on the unit disc, and solved for using the regular square grid configuration as in Example 2.7. The remaining variables are unchanged. The global approximation achieved on the disc after Level 1 is not promising; see Figure 2.20. Further levels of refinement in this case prove pointless in that the approximate data used for the global solution never recovers from this initial problem. The prominent boundary error comes from poor solutions to the local interpolation problems. Such errors are compounded by the existence of grid points with erroneous data being outside of the unit disc. To alleviate this first difficulty, the value of $c_{\text{local}}$ can be increased in order to achieve better pointwise evaluations on the regular grid. Such a strategy is beneficial for all of the given examples. Table 2.5 clearly shows the advantage of increasing $c_{\text{local}}$ at Level 1. The problem of grid points being outside the original domain is not easily overcome and requires more complex subdomain geometries.
Figure 2.20: Example 2.11 Level 1 global approximation error.
Example 2.12

An algorithm is presented to decompose the unit circle into square subdomains, where the iterative refinement takes place. The unit interval is divided into equal squares $W_1, \ldots, W_{p^2}$. The largest empty square is calculated, and replaces most of the panels. The remaining panels are tested and only included if there exists an $x \in W_k$ such that $x$ lies inside the circle; see Figure 2.21. The algorithm then sweeps across the remaining subdomains and searches for 4 adjacent panels which are then replaced by a single square of twice the size. This process is repeated until no further improvements can be made; see Figures 2.22 and 2.23. Care is taken such that when 6 adjacent panels are discovered, that priority for grouping is given to the 4 which are most interior. This is done so that greater resolution is present in the boundary region.

When the hierarchical iterative refinement procedure was applied to this decomposition, the same errors occurred as in Example 2.11 without improvement. The major problem lies in the few panels which lie mainly outside of the circle but which must be included in order to cover the whole original domain. This domain decomposition approach therefore offers a method of computing an approximation on a large percentage of the circle.
2.2.5 Conclusion

A global solution to an interpolation problem involving a large number of data points is too expensive to compute directly if inverse multiquadrics are to be used effectively. However, if the aim is to generate approximations to a function, then such a global solution is unnecessary and an alternative method has been presented.

The underlying idea is to transform given scattered data $f(x_i)$, to regular approximate data $\tilde{f}(y_i)$ which is easier to solve for. The aim is then no longer to interpolate the initial data, but to find a good approximation to it. Success is limited by minimising the local approximation errors $|f(y_i) - \tilde{f}(y_i)|$.

The presented algorithm is $\mathcal{O}(N)$ since the only work related to the number
of initial points is the search for $q$ neighbours to each $y_i$.

If $X$ is a polygonal domain then the domain decomposition is easier to arrange. Such a condition is not a necessary requirement. The inverses relating to gridded triangular domains may be beneficial in complicated decompositions. For further work concerning tiling a polygon with similar triangles see [35], unfortunately this is not very expansive. For nonpolygonal domains, special attention needs to be given to the coverage of the data points in order to reduce the production of spurious function values by the localisation technique. To improve the spacing of the regular points within the domains $Y$, a hexagonal lattice formation could be introduced.

The time required to solve each subdomain problem is reduced due to the use of the stored $(n \times n)$ key inverse matrix. Solving directly would be $O(n^3)$, but
the required matrix–vector product is $O(n^2)$.

Polynomial tensor product splines offer an efficient method for obtaining a single stage approximation, although further refinement is difficult. The hierarchical iterative refinement strategy provides a method of continual improvement, which can be terminated when the required accuracy has been achieved.
2.3 Multilevel polyharmonic splines

Polyharmonic splines are used to interpolate data in the stationary multilevel iterative refinement scheme as described in Section 2.2. It is assumed that the data is already on a regular grid for simplicity, therefore the local interpolation problems do not need to be solved. By using these basis functions the necessary tools are provided to obtain simple pointwise error bounds on the approximation. The key aspect is that the direct form of the seminorm associated with polyharmonic splines only contains derivatives. Therefore scaling the problem by a factor $\delta > 1$ reduces the size of the seminorm. This approach is not possible for positive definite functions $\Phi$ since the direct form of the norm contains $\|\Phi\|_2$ terms which do not scale advantageously.

Linear convergence between levels is shown for regular data on a scaled multiinteger grid, and a multilevel domain decomposition method.

2.3.1 Introduction

Consider data $f(x_i) \in \mathbb{R}$ located at distinct points $x_i \in \mathbb{R}^d$ for $i = 1, \ldots, N$. Let the interpolant be a linear combination of translated basis functions, appended by some polynomial thus

$$s(x) = \sum_{i=1}^{N} \lambda_i \Phi(x - x_i) + p_{m-1}(x)$$

where $p_{m-1} \in \mathbb{P}^d_{m-1}$, and $s(x_i) = f(x_i)$ for $i = 1, \ldots, N$, with extra conditions

$$\sum_{i=1}^{N} \lambda_i q(x_i) = 0, \text{ for all } q \in \mathbb{P}^d_{m-1}$$

as stated in Chapter 1. A similar hierarchical method is used as described in Section 2.2. The approximations that were generated on each panel via the inverse multiquadric are replaced by those obtained by polyharmonic splines.
The domain decomposition aspect of the algorithm is required to ensure that the point configurations for each subinterpolation problem are fixed. Such a condition is also satisfied by a scaled multiinteger grid.

2.3.2 Iterative refinement

Iterative refinement as described in Section 1.1 is employed, with the interpolant at each level being given by

\[ s_k(x) = \sum_{i=1}^{N_k} \lambda_i^k \Phi \left( \Delta^k(x - x_i^k) \right) + p_{m-1}^k(x) \]  

(2.13)

for some scale factor \( \Delta^k \). Then let \( X_k = \{x_1^k, \ldots, x_{N_k}^k\} \). Recall that the error at each level is

\[ e_k = f - \sum_{j=1}^{k} s_j. \]

Then \( s_k \) interpolates \( e_{k-1} \) on \( X_k \). Therefore the errors at different levels are related via

\[ e_1(x) = f(x) - s_1(x) \]

\[ e_k(x) = e_{k-1}(x) - s_k(x) \text{ for } k \geq 2. \]  

(2.14)

The philosophy of this approach is to use tighter basis functions as the levels increase, so \( \Delta^{k+1} > \Delta^k \). The main features of the data \( f \) are captured at the outset, and finer details are resolved as the levels progress.

The use of different basis functions at each level causes complications in the error analysis. By scaling the points in \( X_k \) it is possible to obtain an equivalent interpolation problem using a fixed basis function. The benefit of scaling the points and not the basis function is that there is no transition between different native spaces.

For ease consider the case when \( \Delta^k = \delta^{k-1} \) for \( \delta > 1 \), and define

\[ y_j^k = u + \delta^{k-1}x_j^k \text{ for } j = 1, 2, \ldots, N_k. \]
Further, let $y = u + x\delta^{k-1}$ and $Y_k = u + \delta^{k-1}X_k = \{y_1^k, y_2^k, \ldots, y_{N_k}^k\}$. A translation $u$ is included here in preparation for the application in Section 2.3.4. At the $k^{th}$ level, the previous error $e_{k-1}$ is scaled by a factor of $\delta^{k-1}$ and interpolated using $\Phi$ at the points in $Y_k$. The result is a scaled version of the original interpolant.

By introducing the affine scaling $y = u + x\delta^{k-1}$, the interpolant in (2.13) becomes

$$s_k(x) = \sum_{i=1}^{N_k} \lambda_i^k \Phi(\delta^{k-1}(x - x_1^k)) + p_{m-1}^k(x)$$

$$= \sum_{i=1}^{N_k} \lambda_i^k \Phi(y - y_i^k) + p_{m-1}^k((y - u)/\delta^{k-1})$$

$$= \sum_{i=1}^{N_k} \lambda_i^k \Phi(y - y_i^k) + q_{m-1}^k(y) = t_k(y),$$

where $q_{m-1}^k \in \mathcal{P}_{m-1}^d$. For clarity let the scaled error at each level be given by

$$g_k(y) := e_k(x) = e_k((y - u)/\delta^{k-1}). \quad (2.15)$$

From (2.14) the errors at different levels are related via

$$e_k(x) = e_{k-1}(x) - s_k(x).$$

With scaling this becomes

$$e_k((y - u)/\delta^{k-1}) = e_{k-1}((y - u)/\delta^{k-1}) - s_k(x)$$

and from the definition of $g_k$ and $t_k$ this yields

$$g_1 = f - t_1$$

$$g_k(y) = g_{k-1}(y/\delta) - t_k(y) \text{ for } k \geq 2. \quad (2.16)$$

So as $s_k$ interpolates $e_{k-1}$ on $X_k$, then $t_k$ interpolates the scaled error $g_{k-1}(\cdot/\delta)$ on $Y_k$, and is of the exact form of (1.6).

The key aspect is that the interpolant obtained on the scaled set of points $Y_k$ using $\Phi$ is simply a scaled version of the solution to the problem on $X_k$ which
uses tighter basis functions $\Phi(-\delta^{k-1})$. From (2.15) the maximum pointwise error of the original and scaled problem is the same, therefore proof of convergence for $\|g_k\|_\infty$ is also valid for $\|e_k\|_\infty$. It will be shown in Section 2.3.4 that the error at each level decreases linearly.

Simplification of the error analysis is possible when the points used in $Y_k$ are the same as those in $Y_{k-1}$. Examples of data configurations for which this is satisfied are given.

### 2.3.3 Data configuration

Let $X$ be a scaled multiinteger grid where $X = \mathbb{Z}^d$. Consider subsets of $X$ given by $X_k = \delta^{1-k} \mathbb{Z}^d$, then the scaling $y = \delta^{k-1} x$ provides $Y_k = \mathbb{Z}^d$. Therefore the interpolation problem on $Y_k$ uses the same points for all $k$.

A more practical application of this phenomenon occurs in the domain decomposition method presented in Section 2.2. Let $X$ be a finite regular grid, and let $Z = \{z_1, \ldots, z_n\}$ be a template of gridded points. The set $X$ can then be decomposed into subsets $X_i$ given by

$$X_i = \{x \in X : x = \sigma_i + z/\delta \text{ where } z \in Z\},$$

where $\sigma_i$ essentially translates the scaled template $Z$. Figure 2.24 shows such a decomposition of the unit grid, where $\{\sigma_1, \ldots, \sigma_4\}$ are the vertices of a square of length $\frac{1}{2}$.

At each level the smaller systems on the grid $Z/\delta$ are solved using basis functions $\Phi(\cdot, \delta)$ with support limited to the given subdomain. The affine transformation $y = \delta(x - \sigma_i)$ then generates $Y_i$ given by

$$Y_i = \{y \in \delta(X - \sigma_i) : y = z \text{ where } z \in Z\}.$$

Therefore each $Y_i$ consists of the fixed template $Z$, and so each subinterpolation problem is solved on the same set of points.
As previously discussed, the basis function used on the domain $Y_i$ is given by $\Phi(y)$, and thus independent of $i$. Therefore in the two given data configurations the scaled problems use the same points and basis function for each level. This is a fundamental point to the success of the error analysis.

### 2.3.4 Error analysis

This section concentrates on the error analysis for the multilevel domain decomposition algorithm previously described. It is straightforward to adapt this analysis to the infinite grid case.

In the setting of the multilevel scheme, it is necessary to bound the size of the error $(f - s_1 - s_2 \cdots - s_k)$ measured in some norm. From the work of Schaback presented in Section 1.3, the error bound (1.12) will be used extensively. It is given here again for convenience

$$|f(x) - s_1(x) - s_2(x) \cdots - s_k(x)| \leq P(x)\|f - s_1 - s_2 \cdots - s_{k-1}\|_\Phi \quad (2.17)$$

for every $f \in \mathcal{G}_\Phi$. The given multilevel method relies heavily on scaling, it is therefore important to show that the power function is invariant under affine
transformations. From (1.14)

\[ P(x) = \left\{ \Phi(0) - 2 \sum_{i=1}^{N} \Phi(x - x_i)u_i(x) + \sum_{i,j=1}^{N} \Phi(x_i - x_j)u_i(x)u_j(x) \right\}^{1/2} \]

where the functions \( u_i \) satisfy \( u_i(x_j) = \delta_i^j \). Let \( P_\gamma \) be the power function obtained using \( \Phi(\cdot/\gamma) \) as the basis function. If \( y_j = \bar{u} + \gamma x_j \) and \( y = \bar{u} + \gamma x \) for some \( \bar{u} \in \mathbb{R}^d \) and \( \gamma \in \mathbb{R} \), it must be shown that \( P_\gamma(y) = P(x) \). Now \( u_i^\gamma(y_j) = \delta_i^j \) for \( i, j = 1, \ldots, N \), and \( u_i^\gamma(y) = u_i(x) \). Then

\begin{align*}
P_\gamma(y) &= \left\{ \Phi(0) - 2 \sum_{j=1}^{N} u_j^\gamma(y)\Phi((y - y_j)/\gamma) \\
&\quad + \sum_{i,j=1}^{N} u_i^\gamma(y)u_j^\gamma(y)\Phi((y_i - y_j)/\gamma) \right\}^{1/2} \\
&= P(x).
\end{align*}

Therefore, unlike the case for the usual interpolation problem, the power function is not analysed to infer convergence of the multilevel algorithm. In order to prove pointwise convergence it will be necessary to show that the error in the \( \| \cdot \|_\Phi \) seminorm decreases with successive levels; see (2.17).

The errors on successive panels of different sizes are required throughout the refinement process. For simplicity let \( e_\delta \) now be the error restricted to one of the \( \delta^{dk} \) quadrants of the domain, and then \( g_\delta \) is just the scaled restricted error. At the \( k^{th} \) level \( t_k(y) \) interpolates, and is the best approximation to \( g_{k-1}(y/\delta) \) in \( \mathcal{G}_\Phi \), therefore the following holds

\[ \| g_{k-1}(\cdot/\delta) \|_\Phi^2 = \| g_{k-1}(\cdot/\delta) - t_k(\cdot) \|_\Phi^2 + \| t_k(\cdot) \|_\Phi^2 \quad (2.18) \]

from [38]. Using (2.16) this implies that

\[ \| g_{k-1}(\cdot/\delta) \|_\Phi^2 = \| g_k(\cdot) \|_\Phi^2 + \| t_k(\cdot) \|_\Phi^2. \]

Then

\[ \| g_k(\cdot) \|_\Phi \leq \| g_{k-1}(\cdot/\delta) \|_\Phi \quad (2.19) \]
is an inequality which will be required.

It will be necessary to show that

$$\|g_k\|_\Phi \leq C \cdot \|g_{k-1}\|_\Phi$$ (2.20)

for $C < 1$. This inequality can then be applied recursively to give

$$\|g_k\|_\Phi \leq C^{k-1} \|g_1\|_\Phi.$$ (2.21)

Since $t_1$ is the best approximation to $f$ in $G_\Phi$, a similar orthogonality result to (2.18) gives

$$\|f\|_\Phi^2 = \|f - t_1\|_\Phi^2 + \|t_1\|_\Phi^2
= \|g_1\|_\Phi^2 + \|t_1\|_\Phi^2.$$

Therefore

$$\|g_1\|_\Phi \leq \|f\|_\Phi$$

and (2.21) becomes

$$\|g_k\|_\Phi \leq C^{k-1} \cdot \|f\|_\Phi.$$ (2.22)

At the $k^{th}$ level $t_k(y)$ interpolates $g_{k-1}(y/\delta)$, so the error bound in (2.17) yields

$$|g_{k-1}(y/\delta) - t_k(y)| \leq P(y) \|g_{k-1}(\cdot/\delta)\|_\Phi$$

and from the definition of $g_k$ (2.16), this implies that

$$|g_k(y)| \leq P(y) \|g_{k-1}(\cdot/\delta)\|_\Phi.$$ (2.23)

It is also necessary to prove that

$$\|g_{k-1}(\cdot/\delta)\|_\Phi \leq C \cdot \|g_{k-1}\|_\Phi$$ (2.24)

so that (2.23) becomes

$$|g_k(y)| \leq P(y) \cdot C \cdot \|g_{k-1}\|_\Phi.$$
From the result in (2.22), a final bound on the error at each level is provided thus

$$|g_k(y)| \leq P(y) \cdot C^{k-1} \cdot \|f\|_\Phi.$$  \hspace{1cm} (2.25)

It has already been established that if the data configurations of Section 2.3.3 are employed, then $P$ is independent of the level $k$. Therefore the convergence of $|g_k(y)|$ comes solely from the powers of $C$.

It now remains to show that $C < 1$ in (2.20) and (2.24) for polyharmonic splines.

### 2.3.5 Polyharmonic splines

Seminorms generated by some basis functions can be written explicitly in terms of the function itself rather than the Fourier transform; see [36]. Let $\Phi$ be a polyharmonic spline which is conditionally positive definite of order $m$. In order that $G_\Phi \subseteq C^m(\mathbb{R}^d)$, so that interpolation makes sense, it is necessary for $m > d/2$; see [45]. Then $G_\Phi$ coincides with the Beppo–Levi space of order $m$ on $\mathbb{R}^d$. In these cases (see for instance [16] and [47])

$$\|f\|_\Phi = \left( \sum_{|\alpha| = m} c_\alpha \|D^\alpha f\|^2 \right)^{\frac{1}{2}}$$  \hspace{1cm} (2.26)

and the coefficients $c_\alpha$ are chosen such that

$$\sum_{|\alpha| = m} c_\alpha x^\alpha = |x|^{2m}.$$  

Now, if $g = f(\cdot/\delta)$ then

$$\|D^\alpha g\|_2^2 = \int_{\mathbb{R}^d} |D^\alpha g|^2$$

$$= \delta^{-2m} \int_{\mathbb{R}^d} |D^\alpha f(\cdot/\delta)|^2$$

$$= \delta^{d-2m} \int_{\mathbb{R}^d} |D^\alpha f|^2$$

$$= \delta^{d-2m} \|D^\alpha f\|_2^2.$$
Hence

\[ \| f ( \cdot / \delta ) \|_{\Phi} = \delta^{q - m} \| f \|_{\Phi}. \quad (2.27) \]

Since \( m > d/2 \), then by setting \( C = \delta^{q - m} \) this proves (2.24). It is interesting to note that the condition on \( m \) to obtain (2.26) is exactly that required to guarantee that \( C < 1 \). The proof of (2.20) comes from combining (2.19) and (2.27).

Note that the only condition has been that the point configurations at each level must be similar. The same result can be reproduced for the scaled multi-integer regular grid, where the restriction of analysing a particular quadrant is removed. Numerical results for such grids are however unavailable.

### 2.3.6 Numerical results

As a consequence of the error estimate in (2.25), one would expect to see the pointwise error decay exponentially at a rate of \( C = \delta^{q - m} \). A number of examples are given where a test point \( \tau \) has been chosen to examine the behaviour of \( |g(\tau)| \).

The multilevel domain decomposition scheme as described in Section 2.3.3 is implemented to interpolate a variety of data sets, where \( d = 1 \), given by the functions

\[
\begin{align*}
F_1(x) &= \sin(10x) + 3x^2 \\
F_2(x) &= \frac{\sin(16x)}{16(x + 1)} \\
F_3(x) &= x \exp(-x^2).
\end{align*}
\]

The template \( Z \) consists of 20 regularly spaced points on \([0, 1]\), and basis functions of the form \( \|x\|^\beta \log(\|x\|) \) for \( \beta = 2, 4 \) are used. The scale factor is chosen such that \( \delta = 2 \). Figures 2.25 and 2.26 show the error at the test point \( \tau = \frac{1}{2} \), for
initial data $F_3$ in both cases. Table 2.6 contains approximate convergence rates, and the appropriate values of $C = 2^{-\frac{1}{2}(\beta+1)}$ for comparison.

In $\mathbb{R}^2$ the template $Z$ consists of a $10 \times 10$ regular grid on $[0, 1]^2$. Again the scale factor is chosen such that $\delta = 2$, and the test point is located at $\tau = (\frac{1}{2}, \frac{1}{2})$. The initial data $F_4$ is given by Franke's function

$$F_4(x, y) = 0.75e^{-0.25(9x-2)^2 - 0.25(9y-2)^2} + 0.75e^{-\frac{(9x-2)^2}{49} - \frac{(9y-2)^2}{10}}$$
$$+ 0.5e^{-0.25(9x-7)^2 - 0.25(9y-7)^2} - 0.2e^{-\frac{(9x-4)^2}{49} - \frac{(9y-7)^2}{10}}.$$ 

Figures 2.27 and 2.28 show the pointwise error at each level.

In $\mathbb{R}^3$ then template $Z$ consists of a $6 \times 6 \times 6$ regular grid on $[0, 1]^3$. The test point is located at $\tau = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, and the data $F_5$ is given by

$$F_5(x, y, z) = x^2y^3\cos(10z).$$

Figures 2.29 and 2.30 show the pointwise error at each level. Tables 2.7 and 2.8 contain the approximate convergence rate, and the predicted values of $C$ for the 2D and 3D cases.

The main result (2.25) proves convergence of the error bound, and not the actual error itself. Therefore the similarity between the predicted and actual rates in Table 2.6 may be fortuitous. In the 2D and 3D cases however, it becomes clear that the actual pointwise error converges faster than the predicted error bound predicted. Such a deficiency in the theory may be caused by the use of the orthogonality result (2.18), which disregards possibly vital information to achieve a crude bound.
### CHAPTER 2. COMPUTATIONAL METHODS

<table>
<thead>
<tr>
<th>Data set</th>
<th>Basis function</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta = 2$</td>
<td>$\beta = 4$</td>
</tr>
<tr>
<td>$F_1$</td>
<td>0.353 0.177</td>
</tr>
<tr>
<td>$F_2$</td>
<td>0.353 0.175</td>
</tr>
<tr>
<td>$F_3$</td>
<td>0.353 0.174</td>
</tr>
<tr>
<td>$C$</td>
<td>0.354 0.177</td>
</tr>
</tbody>
</table>

Table 2.6: Convergence rates of the multilevel method in $\mathbb{R}$.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Basis function</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta = 2$</td>
<td>$\beta = 4$</td>
</tr>
<tr>
<td>$F_4$</td>
<td>0.344 0.15</td>
</tr>
<tr>
<td>$C$</td>
<td>0.5 0.25</td>
</tr>
</tbody>
</table>

Table 2.7: Convergence rates of the multilevel method in $\mathbb{R}^2$.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Basis function</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta = 2$</td>
<td>$\beta = 4$</td>
</tr>
<tr>
<td>$F_5$</td>
<td>0.3281 0.086</td>
</tr>
<tr>
<td>$C$</td>
<td>0.707 0.354</td>
</tr>
</tbody>
</table>

Table 2.8: Convergence rates of the multilevel method in $\mathbb{R}^3$. 
Figure 2.25: Convergence of the multilevel method for $r^2 \log(r)$ in $\mathcal{R}$.

Figure 2.26: Convergence of the multilevel method for $r^4 \log(r)$ in $\mathcal{R}$.
Figure 2.27: Convergence of the multilevel method for $r^2 \log(r)$ in $\mathbb{R}^2$.

Figure 2.28: Convergence of the multilevel method for $r^4 \log(r)$ in $\mathbb{R}^2$. 
Figure 2.29: Convergence of the multilevel method for $r^2 \log(r)$ in $\mathbb{R}^3$. 

Figure 2.30: Convergence of the multilevel method for $r^4 \log(r)$ in $\mathbb{R}^3$. 
2.3.7 Conclusion

Error estimates for the domain decomposition method in Section 2.2 using polyharmonic splines are provided. Whilst it would be preferable to obtain error bounds for use with the original inverse multiquadric, it is as yet unclear how to proceed.
Chapter 3

Compactly supported functions

3.1 Outline

The current methods for constructing PD compactly supported radial basis functions are discussed. The main result is a characterisation of compactly supported PD functions similar to that of Schoenberg, but in the Fourier domain. In the polynomial case, Wendland has shown a relation between the smoothness and the degree of positive definiteness, this result is here extended to all functions.

Hegland [30] highlights a curious aspect of high dimensional interpolation using compactly supported functions, which is mentioned for interest.

This chapter also provides the necessary construction details required in the sequel.

3.2 Introduction

Compactly supported PD radial basis functions have become popular computational tools because the associated interpolation matrices are sparse. There exist a number of different families of such functions, and these share common features; see [8], [56] and [60]. The purpose of this chapter is to discuss some of
these features and decide whether these common properties are true for all compactly supported basis functions, or whether they are artificial similarities created by the choice of construction.

To discuss these issues, some notation and terminology is required. Assume that all univariate functions in this chapter are defined on \([0, \infty)\). Let \(\text{CS}\) denote the set of functions with support on \([0,1]\). Let \(C^k\) denote the set of functions which have \(k\) continuous derivatives, and let \(C^0\) denote the set of continuous functions. Let \(\phi \in C^l(x_0)\) mean that \(\phi\) has \(l\) continuous derivatives in the neighbourhood of \(x_0\). By definition of compactly supported functions it is consistent to let \(\phi^{(j)}(1)\) denote the limit of \(\phi^{(j)}(t)\) as \(t\) tends to 1 from below.

Consider the interpolant given by (1.3), and the associated system of linear equations. For \(A = (\Phi(x_i - x_j))_{1 \leq i, j \leq N}\) to be \(\text{PD}\) is sufficient to ensure nonsingularity of the system.

**Definition 3.1** A function \(\phi\) is strictly positive definite in \(d\) dimensions, written \(\text{PD}_d\), if

\[
\mathcal{F}_d(\phi)(w) = \int_{\mathbb{R}^d} \phi(||x||) e^{-iw^T x} dx
\]

is nonnegative on \(w \in \mathbb{R}^d\), and positive at least on an open subset. The one dimensional Fourier transform \(\mathcal{F}_1(\phi)\) will be denoted by \(\hat{\phi}\).

If \(\phi \in \text{PD}_d\) and \(X \subseteq \mathbb{R}^d\), then \(A\) is strictly positive definite. There exist functions which are \(\text{PD}\) in all space dimensions for example

- (i) \(\phi(r) = \exp(-cr^2) \in \text{PD}_\infty\), gaussian,
- (ii) \(\phi(r) = (c^2 + r^2)^{-\frac{1}{2}} \in \text{PD}_\infty\), inverse multiquadric.

where \(c\) is a positive constant shape parameter.

By virtue of Bochner's theorem [11], any function \(\phi \in \text{PD}_\infty\) has the Fourier–Stieltjes integral representation

\[
\phi(r) = \int_{-\infty}^{+\infty} e^{iru} d\alpha(u)
\] (3.2)
where \( \alpha \) is a nondecreasing, bounded measure. It has been shown by Schoenberg [52] that the existence of such an integral is equivalent to certain conditions on the behaviour of the function \( f(r) = \phi(\sqrt{r}) \).

The Bernstein–Widder Theorem [59] shows that any function \( f \in \text{CM}[0, \infty) \) has the Laplace–Stieltjes integral representation

\[
f(r) = \int_0^\infty e^{-ru}d\beta(u)
\]

where \( \beta \) is a nondecreasing, bounded measure. In [52] the combination of (3.2) and (3.3) result in

Theorem 3.4 \( \phi \in \text{PD}_\infty \) if and only if \( f \in \text{CM}[0, \infty) \), where \( f(r) = \phi(\sqrt{r}) \).

This may be illustrated using the previous two examples

(i) \( \phi(\sqrt{r}) = \exp(-cr) \in \text{CM}[0, \infty) \), gaussian,

(ii) \( \phi(\sqrt{r}) = (c^2 + r)^{-\frac{1}{2}} \in \text{CM}[0, \infty) \), inverse multiquadric.

Since all interpolation problems must be set in finite dimensional space, \( \text{PD}_\infty \) functions are not necessary to ensure solvability. Moreover, due to the conditions on the measure \( \alpha \), compactly supported functions cannot be represented via (3.2), and therefore cannot be \( \text{PD}_\infty \). This motivates a search for generating \( \text{CS} \cap \text{PD}_d \) functions where \( d < \infty \).

There exist numerous methods for generating functions \( \phi \in \text{CS} \cap \text{PD}_d \); see Buhmann [8], Wendland [56] and Wu [60]. In order to examine the polynomials of Wendland and Wu, consider the following operators

\[
D(f)(r) = -\frac{1}{r} \frac{d}{dr} f(r)
\]

\[
I(f)(r) = \int_r^\infty rf(r)dr.
\]

Wendland summarises the connections between \( I, D \) and \( \mathcal{F}_d \) which are given here in
Proposition 3.5

(i) The operators $I$ and $D$ are inverse: If $\phi$ is continuous and satisfies $t\phi(t) \in L_1$, then $DI\phi = \phi$. On the other hand, if $\phi \in C^2$ is even and satisfies $\phi' \in L_1$, then $ID\phi = \phi$.

(ii) If $\phi(t)t^{d-1} \in L_1$ and $d \geq 3$, then $\mathcal{F}_d(\phi) = \mathcal{F}_{d-2}(I\phi)$.

(iii) If $\phi \in C^2$ and $\phi(t)t^{d-1} \in L_1$, then $\mathcal{F}_d(\phi) = \mathcal{F}_{d+2}(D\phi)$.

In the current setting where $\phi$ is finite and compactly supported, the above conditions such as $\phi(t)t^{d-1} \in L_1$ are not restrictive. To prove the main result, the following propositions are required.

Proposition 3.6 If $\phi \in C^0$ and $t^2\phi(t) \in L_1$ then $(I\phi)^\wedge = D\hat{\phi}$.

Proof: If $\phi \in C^0$ and $t^2\phi(t) \in L_1$ then

$$\int_0^\infty |I\phi| \, dr = \int_0^\infty \left| \int_r^\infty t\phi(t) \, dt \right| \, dr = \lim_{M \to \infty} \int_0^M \left| \int_r^M t\phi(t) \, dt \right| \, dr + \lim_{M \to \infty} \int_0^M \left| \int_M^\infty t\phi(t) \, dt \right| \, dr$$

$$\leq \lim_{M \to \infty} \int_0^M \left\{ \int_0^t \, dt \right\} |t\phi(t)| \, dt + \lim_{M \to \infty} \int_0^M \left| \int_M^\infty t|\phi(t)| \, dt \right| \, dr$$

$$= \lim_{M \to \infty} \int_0^M t^2|\phi(t)| \, dt + \lim_{M \to \infty} M\int_M^\infty t|\phi(t)| \, dt$$

$$\leq \lim_{M \to \infty} \left\{ \int_0^M t^2|\phi(t)| \, dt + \int_M^\infty t^2|\phi(t)| \, dt \right\}$$

$$= \int_0^\infty t^2|\phi(t)| \, dt < \infty$$

so that $I\phi \in L_1$. Taking the Fourier transform of $I\phi$ then

$$(I\phi)^\wedge(w) = \int_0^\infty \left\{ \int_r^\infty t\phi(t) \, dt \right\} \cos(wr) \, dr$$
CHAPTER 3. COMPACTLY SUPPORTED FUNCTIONS

\[ \int_0^\infty t \phi(t) \left\{ \int_0^t \cos(wr) \, dr \right\} \, dt \]
\[ = \frac{1}{w} \int_0^\infty t \phi(t) \sin(wt) \, dt \]
\[ = -\frac{1}{w} \frac{d}{dw} \int_0^\infty \phi(t) \cos(wt) \, dt \]
\[ = \hat{D}\hat{\phi}(w). \]

The change of order of integration is justified by using exactly the same style of argument as in proving that \( I\phi \in L_1 \). \( \square \)

A similar result will be required in Chapter 4 and is given here for completeness.

**Proposition 3.7** If \( \phi \in C^1 \), then \( (D\phi)^\wedge = I\hat{\phi} \).

**Proof:** Consider

\[ \frac{\partial}{\partial w} (D\phi)^\wedge = -\frac{\partial}{\partial w} \int_0^1 \frac{\cos(rw)}{r} \phi'(r) \, dr \]
\[ = \int_0^1 \sin(rw) \phi'(r) \, dr \]

and from integration by parts

\[ \hat{\phi}(w) = \int_0^1 \cos(rw) \phi(r) \, dr \]
\[ = -\int_0^1 \frac{\sin(rw)}{w} \phi'(r) \, dr. \]

Therefore

\[ \frac{\partial}{\partial w} (D\phi)^\wedge = -w\hat{\phi}(w). \]

Due to the continuity of \( \phi \), then \( \hat{\phi}(w) = O(w^{-3}) \) for large \( w \) from Morrison [41]. Therefore \( \int_w^\infty s\hat{\phi}(s) \, ds \) is finite. Hence

\[ (D\phi)^\wedge = \int_w^\infty s\hat{\phi}(s) \, ds \]

which completes the proof. \( \square \)
\[ \phi_{\text{We}}^{1,0} = (1 - r)_+^1 \in \text{PD}_1 \cap C^0 \]
\[ \phi_{\text{We}}^{2,1} = (1 - r)_+^3 (3r + 1) \in \text{PD}_1 \cap C^2 \]
\[ \phi_{\text{We}}^{3,2} = (1 - r)_+^5 (8r^2 + 5r + 1) \in \text{PD}_1 \cap C^4 \]
\[ \phi_{\text{We}}^{2,0} = (1 - r)_+^2 \in \text{PD}_3 \cap C^0 \]
\[ \phi_{\text{We}}^{3,1} = (1 - r)_+^4 (4r + 1) \in \text{PD}_3 \cap C^2 \]
\[ \phi_{\text{We}}^{4,2} = (1 - r)_+^6 (35r^2 + 18r + 3) \in \text{PD}_3 \cap C^4 \]
\[ \phi_{\text{We}}^{3,0} = (1 - r)_+^3 \in \text{PD}_5 \cap C^0 \]
\[ \phi_{\text{We}}^{4,1} = (1 - r)_+^5 (5r + 1) \in \text{PD}_5 \cap C^2 \]
\[ \phi_{\text{We}}^{5,2} = (1 - r)_+^7 (16r^2 + 7r + 1) \in \text{PD}_5 \cap C^4 \]

Table 3.1: Examples of Wendland's polynomials.

In [56] Wendland starts with \( C^0 \cap \text{PD}_{2l-1} \) polynomials created by the product of \( \text{PD}_1 \) polynomials thus

\[ \phi_{\text{We}}^{l} = (1 - r)_+^l \]

where

\[ (t)_+ = \begin{cases} 0 & \text{for } t < 0 \\ t & \text{for } t \geq 0. \end{cases} \]

Using Proposition 3.5 (ii), dimensions of positive definiteness are traded for increased smoothness by the \( I \) operator, then

\[ \phi_{\text{We}}^{l,k} = I^k (1 - r)_+^l. \quad (3.8) \]

Table 3.1 contains examples of Wendland's polynomials, where \( \doteq \) denotes equality up to a positive multiple.

In [60] Wu starts with smooth \( \text{PD}_1 \) polynomials created by the convolution of \( (1 - r^2)_+^l \) with itself

\[ \phi_{\text{Wu}}^{l} = (1 - r^2)_+^l \ast (1 - r^2)_+^l. \]

Continuous derivatives of \( \phi_{\text{Wu}}^{l} \) are exchanged for higher dimensions of positive
CHAPTER 3. COMPACTLY SUPPORTED FUNCTIONS

\[ \phi_{0,0}^{\text{Wu}} = (1 - r)_+ \in \mathbf{PD}_1 \cap C^0 \]
\[ \phi_{1,0}^{\text{Wu}} = (1 - r)^3 (r^2 + 3r + 1) \in \mathbf{PD}_1 \cap C^2 \]
\[ \phi_{2,0}^{\text{Wu}} = (1 - r)^5 (r^4 + 5r^3 + 9r^2 + 5r + 1) \in \mathbf{PD}_1 \cap C^4 \]
\[ \phi_{1,1}^{\text{Wu}} = (1 - r)^2 (r + 2) \in \mathbf{PD}_3 \cap C^0 \]
\[ \phi_{2,1}^{\text{Wu}} = (1 - r)^4 (3r^3 + 12r^2 + 16r + 4) \in \mathbf{PD}_3 \cap C^2 \]
\[ \phi_{3,1}^{\text{Wu}} = (1 - r)^6 (5r^5 + 30r^4 + 72r^3 + 82r^2 + 36r + 6) \in \mathbf{PD}_3 \cap C^4 \]
\[ \phi_{2,2}^{\text{Wu}} = (1 - r)^3 (3r^2 + 9r + 8) \in \mathbf{PD}_5 \cap C^0 \]
\[ \phi_{3,2}^{\text{Wu}} = (1 - r)^5 (5r^4 + 25r^3 + 48r^2 + 40r + 8) \in \mathbf{PD}_5 \cap C^2 \]

Table 3.2: Examples of Wu’s polynomials.

definiteness by using the \( D \) operator from Proposition 3.5 (iii). Then

\[ \phi_{i,k}^{\text{Wu}} = D^k \phi_i^{\text{Wu}}. \]

Table 3.2 contains examples of Wu’s polynomials.

In [8] Buhmann seeks to find functions of the form

\[ \phi^{\text{Bu}}(r) = p(r) + q(r^2) \log(r), \]

where \( p \) and \( q \) are polynomials. Such functions can be viewed as compactly supported polyharmonic splines in an even space dimension, appended by a cut–off polynomial. Positive definiteness is guaranteed by the Fourier transform \( \mathcal{F}_d(\phi^{\text{Bu}}) \) reducing to an integral of a Bessel function which is known to be positive. Basis functions of this type are generally represented by integrals

\[ \phi^{\text{Bu}}(r) = \int_0^\infty (1 - r^2 / \beta) d \beta, \]

where \( g \) is a compactly supported continuous function on \( \mathbb{R}_+ \). In [8] the function \( g \) is given as \( g(\beta) = (1 - \beta^\mu)^\epsilon \). Then

\[ \mathcal{F}_d(\phi^{\text{Bu}})(w) \equiv w^{-d - 2 - 2\mu_\epsilon} \int_0^\infty J_{\lambda + d/2}(t) t^{d/2 - \lambda + 1} (w^{2\mu} - t^{2\mu})^\epsilon dt \quad (3.9) \]
Variables p(r) q(r^2)
\(\lambda = \rho = 1, \mu = \frac{1}{2}\) \(\frac{1}{2} + r^2 - \frac{4}{3}r^3\) 2r^2
\(\lambda = 1, \rho = 4, \mu = \frac{1}{2}\) \(\frac{1}{15} + \frac{19}{6}r^2 - \frac{16}{3}r^3 + 3r^4 - \frac{16}{15}r^5 + \frac{6}{5}r^6\) 2r^2

Table 3.3: Examples of Buhmann's functions.

where \(J\) denotes the Bessel function of order according to the index; see [1]. The integral in (3.9) is strictly positive if \(0 < \mu \leq \frac{1}{2} \leq \rho\) and either

\[d = 1, \lambda \geq \frac{1}{2}\]

or

\[d \geq 2\) and \(\lambda \geq \frac{1}{2}(n - 1).\]

Table 3.3 contains examples of Buhmann's functions.

The sums and products of PD functions are also PD, which allows the construction of further families of functions. The first method is to construct a positive sum of different PD basis functions centred at the origin. Wendland generates these functions such that they consist of \(\phi_{i,k}^{\text{We}}\) polynomials of different support, therefore

\[
\psi^{\text{We}}(r) = \int_{t=r}^{1} \phi_{i,k}^{\text{We}}(r/t)\alpha(t)dt \tag{3.10}
\]

where \(\alpha\) is some nonnegative function. If \(\phi_{i,k}^{\text{We}}\) is replaced by the Hat function, it can be seen that all strictly monotone decreasing functions can be written in the form of (3.10) and are therefore PD.

A second method forms a linear combination of the same basis function centred at different locations. Consider the 1D case where the same PD function \(\phi_k\) (with arbitrary support) is centred at integer locations \(k\) with weights such that \(\alpha_i = \alpha_{-i}\), then the new function \(\psi\) is given by

\[
\psi(||x||) = \sum_{k=-n}^{n} \alpha_i \phi(||k - x||)
\]
where $\alpha \in \mathbb{R}^{2n+1}$. Since $\psi$ is even, then the one dimensional Fourier transform is given by

$$\hat{\psi}(w) = \sum_{k=-n}^{n} \alpha_k \int_{\mathbb{R}} \phi(|k-x|) e^{-iwx} dx$$

$$= \sum_{k=-n}^{n} \alpha_k \int_{\mathbb{R}} \phi(|x|) e^{iwx} e^{-iwx} dx$$

$$= \hat{\phi}(w) \left\{ \sum_{k=-n}^{n} \alpha_k e^{ikw} \right\}$$

$$= \hat{\phi}(w) \left\{ \alpha_0 + \sum_{k=1}^{n} \alpha_k (e^{ikw} + e^{-ikw}) \right\}$$

$$= \hat{\phi}(w) \left\{ \alpha_0 + 2 \sum_{k=1}^{n} \alpha_k \cos(kw) \right\}.$$ 

In order for $\psi \in \mathbf{PD}_1$ then the trigonometric sum must satisfy

$$\alpha_0 + 2 \sum_{k=1}^{n} \alpha_k \cos(kw) > 0 \text{ for all } w. \quad (3.11)$$

Various conditions are given by Askey [2] and Brown [7] for such trigometric sums to be positive. The following proposition comes from [2] and is typical of such results.

**Proposition 3.12** If $0 \leq a_0 \geq a_1 \geq \cdots \geq a_n > 0$ and $(2k)a_{2k} \leq (2k-1)a_{2k-1}$ for $k \geq 1$ then

$$\sum_{k=0}^{n} a_k \cos kx > 0 \text{ for } 0 < x < \pi.$$ 

These however are quite restrictive in terms of (3.11) due to the fact that the strictly positive Fourier transform $\hat{\phi}_0$ is only counted once, whereas the other contributions $\hat{\phi}_k$ have double weight.

Examination of $\phi_{Bu}^\text{We}$ and $\phi_{Wu}^\text{We}$ reveals they all have the property that $\phi(\sqrt{\cdot}) \in \mathbf{CM}[0,1)$. An interesting problem is whether or not all $\mathbf{PD}_d$ functions have a monotonicity condition reminiscent of the Schoenberg result for $\mathbf{PD}_\infty$. 

This question is answered in the negative in the following section, showing that there exist nonpositive PD functions. However, Schoenberg’s characterisation of PD functions is recast in the Fourier domain, and it is shown that the dimension in which a function is PD may be characterised using its univariate Fourier transform. It is also noted that for the constructions employed by Buhmann, Wendland and Wu, if $\phi \in CS \cap C^{2k}(0) \cap C^l(1)$ then $\phi \in PD_{2(l-2k)+1}$. In [58] Wendland shows that this is the maximum dimension in which $\phi$ can be PD, if $\phi$ is a polynomial. In Section 3.4 this result is reproduced without the polynomial restriction, and using a counterexample, it is shown that there exist functions whose smoothness does not uniquely identify the dimension of positive definiteness.

The fact that compactly supported functions can only be positive definite in $d$ dimensions does have some curious consequences. Hegland [30] demonstrated that as the space dimension $n$ increases, so too does the weight of the diagonal in the associated interpolation matrix. This is interesting since the matrix can be singular for $n > d$.

### 3.3 Characterisation of positive definiteness

To demonstrate that there is no direct analogue to Schoenberg’s theorem for PD functions, it will be shown that there exist nonpositive PD functions in arbitrary space dimensions. To accomplish this, Green’s theorem will be required in the form of

**Proposition 3.13** Let $S$ be the boundary of the bounded set $B$. Let $f, g \in C^2(\mathbb{R}^d)$. Then

$$\int_B (f \Delta g - g \Delta f) dx = \int_S \left( f \frac{dg}{dn} - g \frac{df}{dn} \right) dS$$

where $\frac{d}{dn}$ denotes differentiation in the direction of the normal pointing out of the surface $S$. 
**Theorem 3.14** Let $f \in C^2(\mathbb{R}^d)$ be compactly supported and PD. Then $-\Delta f$ is also PD and compactly supported.

**Proof:** Assume without loss of generality that $f$ has support strictly contained inside the unit ball. Green's theorem will be used with $g(x) = e^{-ix^T y}$, $B$ is the unit ball in $\mathbb{R}^d$ and $S$ is the unit spherical shell in $\mathbb{R}^d$. Since $f$ is constantly zero outside of the unit ball, then $\frac{\partial f}{\partial n}$ and $f$ are zero on $S$. Therefore

\[
F_d(\Delta f)(y) = \int_{\mathbb{R}^d} \Delta f(x)e^{-ix^T y}dx
= \int_{B} \Delta f(x)e^{-ix^T y}dx
= \int_{B} f(x)\Delta x e^{-ix^T y}dx
= -\|y\|^2 F_d(f)(y) < 0.
\]

Therefore the proof is complete. \qed

**Example 3.15**

In 1 dimension it is now easy to construct nonpositive PD functions by differentiating twice. Because $\phi^{\text{We}}$ and $\phi^{\text{Wu}}$ have a zero derivative at the origin (unless they are at most $C^0(0)$) and a zero derivative at $r = 1$ it must be that these functions have a negative second derivative at the origin and a positive second derivative at $r = 1$. Thus, the second derivative of both of these families of functions is not one signed, but are strictly negative definite.

**Example 3.16**

In $d$ dimensions, for radial functions, the Laplacian has the form

\[
\Delta \phi = \frac{d^2 \phi}{dr^2} + \frac{d - 1}{r} \frac{d \phi}{dr}.
\]

In [56] it is given that $\phi(r) = (1 - r)^d(dr + 1) \in \text{PD}_{d-2}$ if $d \geq 3$ is odd. For this choice of $\phi$

\[
\Delta \phi = (1 - r)^{d-2}d(d + 1)(2dr - d - r)
\]
which is positive when \( \frac{d}{2d - 1} < r < 1 \) and negative when \( 0 \leq r < \frac{d}{2d - 1} \).

Thus demonstrating the existence of nonpositive \( \text{PD} \) functions in arbitrary space dimensions.

An interesting consequence of this search reveals that the most negative value of \( \phi \) determines a maximum bound on the dimension of positive definiteness.

**Theorem 3.17** Let \( \phi \) be a continuous radial function such that \( \phi(0) = 1 \), and \( \phi(r) \geq -b \) for all \( r \). If \( \phi \in \text{PD}_d \) then \( d < \frac{1}{b} \).

**Proof:** If \( \phi \in \text{PD}_d \) then the interpolation problem set in \( \mathbb{R}^d \) produces a nonsingular matrix. Consider the interpolation problem in \( \mathbb{R}^d \) with \( (d + 1) \) equidistant centres. In 2 dimensions this amounts to an equilateral triangle, in 3 dimensions a regular tetrahedron and so on. The associated interpolation matrix \( A \) is given by

\[
A = \begin{bmatrix}
1 & a & \cdots & a & a \\
a & 1 & \cdots & a & a \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
a & a & \cdots & 1 & a \\
a & a & \cdots & a & 1 \\
\end{bmatrix}
\]

for some value \( a < 1 \). It can be easily verified that

\[
\det(A) = (1 + ad)(1 - a)^d.
\]

Therefore if there exists an \( r^* \) such that \( 1 + \phi(r^*)d = 0 \), then \( A \) is singular and hence not \( \text{PD}_d \). Since \( \phi(r) \geq -b \) for all \( r \), then \( r^* \) can only exist when

\[
0 = 1 + \phi(r^*)d \geq 1 - bd
\]

which implies that \( d \geq \frac{1}{b} \). Since \( \phi \) is continuous, such an \( r^* \) will exist unless \( d < \frac{1}{b} \). □

Other carefully constructed data configurations may result in tighter bounds.

**Corollary 3.18** If \( \phi \in \text{PD}_\infty \) then \( \phi(r) > 0 \) for all \( r \).
These examples show that there is no analogue to Theorem 1.5 characterising the dimension of positive definiteness by the behaviour of the function and its derivatives. However, as will be seen in the rest of this section, the dimension in which a function is PD can be characterised by the behaviour of its univariate Fourier transform.

In [60] Wu notes that if $\psi(r^2) = \phi(r)$ then
\[
\frac{d\psi}{dr} = \frac{1}{2r} \frac{d\phi}{dr} = \frac{D\phi}{2}
\]
and therefore
\[
\frac{d^k\psi}{dr^k} = \left(\frac{-1}{2}\right)^k D^k\phi.
\]
Hence $\psi$ is completely monotone if and only if $\phi > 0$ and $D^k\phi > 0$ for all $k \in \mathbb{N}$.

Using this observation it is possible to characterise the dimension of positive definiteness in terms of the Fourier transform.

**Theorem 3.19** Fix $k \in \mathbb{N}$ and suppose that $t^l\phi(t) \in L_1$ for all $0 \leq l \leq k$. Then $\phi \in PD_{2k+1}$ if and only if $D^j\phi > 0$ for all $0 \leq j \leq k$. That is to say that $\phi(\sqrt{t})$ is monotone up to degree $k$.

**Proof:** Using Proposition 3.5 (ii) $\phi \in PD_{2k+1}$ if and only if $I^k\phi \in PD_1$. Now applying Proposition 3.6 $k$ times gives $\phi \in PD_{2k+1}$ if and only if $D^j\phi > 0$ for all $0 \leq j \leq k$. □

This theorem shows that if $\phi \in PD_{2k}$ then $\phi \in PD_{2k+1}$ for $k \in \mathbb{N}$. Therefore it only makes sense to talk about functions which are at most strictly positive definite in odd space dimensions.

It is now possible to recast Theorem 1.5 in terms of Fourier transforms.

**Corollary 3.20** Suppose $t^l\phi(t) \in L_1$ for all $l \in \mathbb{N} \cup \{0\}$. Then $\phi \in PD_\infty$ if and only if $D^j\phi > 0$ for all $j \in \mathbb{N} \cup \{0\}$. Alternatively, $\phi \in PD_\infty$ if and only if $\phi(\sqrt{u})$ is completely monotone.
Remark 3.21 The condition that \( t^l \phi(t) \in L_1 \) for all \( l \in \mathbb{N} \cup \{0\} \) means that \( \phi \) must have faster than algebraic decay. Therefore Corollary 3.20 excludes the generalised inverse multiquadric \((c^2 + r^2)^{\beta/2}\) when \( \beta < 0 \).

3.4 Smoothness of compactly supported basis functions

In this section the results of Wendland [58] are extended to include nonpolynomials. It will be shown that for some \( d \), if \( \phi \in \text{CS} \cap C^{2k}(0) \cap C^l(1) \cap \text{PD}_d \) then \( d \leq 2(l - 2k) + 1 \). The functions \( \phi^{Bu}, \phi^{We} \) and \( \phi^{Wu} \) all satisfy \( \phi \in \text{PD}_{2(l-2k)+1} \). It is demonstrated that the smoothness conditions at the origin and cut-off do not determine the dimension of positive definiteness of compactly supported functions, for there exist functions such that \( \phi \in \text{CS} \cap C^{2k}(0) \cap C^l(1) \cap \text{PD}_d \) and \( d < 2(l - 2k) + 1 \).

The following lemmas are required to construct a proof.

Lemma 3.22 If a CS function \( \phi \) has a simple jump-discontinuity at \( r = 1 \), then \( \phi \) is not PD in any dimension.

Proof: It is only necessary to show that \( \phi \notin \text{PD}_1 \), that is \( \hat{\phi}(w) < 0 \) for some \( w \). Assume \( \phi \in \text{PD}_1 \), and seek a contradiction. Due to the jump-discontinuity, \( \phi \) can be written as

\[ \phi = a + g \]

where \( a \) is a unit step function of value \( \phi(1) \), and \( g \) is a continuous CS function. Then

\[ \hat{\phi}(w) = \frac{\phi(1)}{w} \sin(w) + \hat{g}(w). \]

Now \( g \) is continuous, and \( \hat{g} \) exists because \( \phi \) is assumed to be in \( \text{PD}_1 \), so there exists a constant \( c \) such that \( |\hat{g}(w)| \leq c/w^2 \) for large \( w \); see Morrison [41]. There-
fore \( \hat{\phi} \) cannot be one signed, leading to a contradiction. Since \( \phi \notin \text{PD}_1 \), then \( \phi \) cannot be \( \text{PD} \) in any dimension. \( \square \)

It is necessary to see how the action of \( D \) on \( \phi \) affects the smoothness of the function around zero and the cut-off.

**Lemma 3.23** Assume \( \phi \in C^2 \) and \( t^2 \phi(t) \in L_1 \). If \( \phi \in \text{CS} \cap C^{2k}(0) \cap C^l(1) \cap \text{PD}_d \) then \( D\phi \in \text{CS} \cap C^{2k-2}(0) \cap C^{l-1}(1) \cap \text{PD}_{d+2} \).

**Proof:** The increase in dimension of positive definiteness comes directly from Proposition 3.5 (iii). Now let \( \psi = D\phi \) which implies that

\[
\frac{d}{dr} \phi(r) = r\psi(r).
\]

So using Leibnitz rule

\[
\phi^{(j)}(0) = (j-1)\psi^{(j-2)}(r) + r\psi^{(j-1)}(r) \quad \text{for} \quad j = 1, 2, 3, \ldots .
\]

Thus

\[
\phi^{(j)}(0) = 0 \iff \psi^{(j-2)}(0) = 0.
\]

So if \( \phi \in C^{2k}(0) \), then \( \psi = D\phi \in C^{2k-2}(0) \).

Now \( \phi \in C^l(1) \) so

\[
\phi^{(j)}(1) = (j-1)\psi^{(j-2)}(1) + \psi^{(j-1)}(1) = 0 \quad \text{for} \quad j = 1, 2, \ldots, l
\]

and \( \phi^{(l+1)}(1) = l\psi^{(l-1)}(1) + \psi^{(l)}(1) \neq 0 \).

Therefore \( \psi^{(j)}(1) = 0 \) for \( j = 1, 2, \ldots, l - 1 \), and \( \psi^{(l)}(1) \neq 0 \), which implies that

\[
\psi = D\phi \in C^{l-1}(1).
\]

\( \square \)

Lemma 3.23 can be easily rewritten for the \( I \) operator given the conditions in Proposition 3.5(i).

**Lemma 3.24** If \( \phi \in \text{CS} \cap C^{2k}(0) \cap C^l(1) \cap C^{l+1}(0,1) \) with \( 2k > l > 1 \), then \( \phi \) cannot be \( \text{PD} \) in any dimension. That is to say that if \( \phi \) is smoother around the origin than at the cut-off, then \( \phi \notin \text{PD}_1 \).
Proof: As in Lemma 3.22 it is only necessary to show that $\phi \notin \text{PD}_1$, that is $\hat{\phi}$ is not one signed. Since the lemma can equally be applied to $D\phi$, where $D\phi \in C^{l-1}(1)$, then it can be assumed without loss of generality that $l$ is odd. Since $\phi$ is a compactly supported radial function, then

$$\hat{\phi}(w) = \int_0^1 \cos(wr)\phi(r)dr.$$ 

Integration by parts twice gives

$$\hat{\phi}(w) = \frac{\sin(w)\phi(1)}{w} + \frac{\cos(w)\phi'(1)}{w^2} - \frac{\phi'(0)}{w^2} - \frac{[\phi'']^w(0)}{w^2}.$$ 

But $\phi$ is at least twice continuously differentiable, so $\phi(1) = 0$, $\phi'(1) = 0$, and $\phi'(0) = 0$. Therefore

$$\hat{\phi}(w) = -\frac{1}{w^2} [\phi'']^w$$

and $\phi''$ is continuous. Repeating this process yields

$$\hat{\phi}(w) = \frac{(-1)^j}{w^{2j}} [\phi^{(2j)}]^w$$

for $j = 1, 2, \ldots, \frac{l-1}{2}$, where $\phi^{(2j)}$ is a continuous function on $[0, 1]$. When $j = \frac{l+1}{2}$ for odd $l$, $\phi^{(2j)}(r)$ is discontinuous at $r = 1$, and so from Lemma 3.22, $\hat{\phi}$ cannot be one signed. Therefore $\phi$ is not PD in any dimension. □

Lemma 3.25 If $\psi \in C^{2k}(0) \cap C^{2k}(1)$ then $\psi$ can be at most in PD$_d$. Further, if $\varphi \in C^0(0) \cap C^k(1)$ then $\varphi$ can be at most in PD$_{2k+1}$.

Proof: Assume $\psi \in \text{CS} \cap C^{2k}(0) \cap C^{2k}(1) \cap \text{PD}_d$ for $d \geq 3$, and seek a contradiction. From Proposition 3.5 (ii), there exists a function $\phi \in \text{PD}_{d-2}$ such that $\psi = D\phi$. This means that from Lemma 3.23 $\phi \in C^{2k+2}(0) \cap C^{2k+1}(1)$ and must be at least PD$_1$. A contradiction arises from Lemma 3.24 which states that such a function which is smoother around the origin than at the cut-off cannot be PD$_1$. Proof of the second assertion follows by defining $\varphi = D^k\psi$ in conjunction with Lemma 3.23. □

The dimension of positive definiteness can now be bound by the smoothness of the basis function.
Theorem 3.26 If \( \psi \in \text{CS} \cap C^{2k}(0) \cap C^l(1) \cap \text{PD}_d \) then \( d \leq 2(l-2k) + 1 \).

Proof: Consider \( \psi \in C^{2k}(0) \cap C^l(1) \), then \( \varphi = D^k\psi \in C^0(0) \cap C^{l-k}(1) \), which Lemma 3.25 states is at most \( \text{PD}_{2(l-k)+1} \). From Proposition 3.5 (ii) \( \psi \) is therefore at most \( \text{PD}_{2(l-2k)+1} \). \( \square \)

To demonstrate that equality cannot be achieved in Theorem 3.26, nor can the smoothness of \( \phi \) determine the dimension of positive definiteness, consider the following example.

Example 3.27

Let \( \phi_\alpha(r) = (1-r)^2(1+\alpha r) \) for \( \alpha \in [0,2) \) then \( \phi_\alpha \in C^0(0) \cap C^1(1) \). If \( \phi_\alpha \in \text{PD}_d \), then \( d \) can be determined by the behaviour of \( \hat{\phi}_\alpha \) and \( D\hat{\phi}_\alpha \) via Theorem 3.19. Note that \( \phi_0 = \phi_{2,0}^w \) which is known to be \( \text{PD}_3 \). When \( \alpha = 1 \)

\[
\hat{\phi}_1(w) = \frac{6(1 - \cos(w)) + w^2 - 4w \sin(w)}{w^4}.
\]

Now, for all \( w > 0 \), \( 1 - \cos(w) > w^2/2 - w^4/24 \) and \( \sin(w) < w - w^3/6 + w^5/120 \). Thus

\[
6(1 - \cos(w)) + w^2 - 4w \sin(w) \geq 6 \left( \frac{w^2}{2} - \frac{w^4}{24} \right) + w^2 - 4w \left( w - \frac{w^3}{6} - \frac{w^5}{120} \right) = \frac{5w^4}{12} - \frac{w^6}{30} > 0,
\]

for \( w^2 < 25/2 \). This is satisfied with \( w \leq \pi \). For \( \pi < w < 2\pi \), then \( 6(1 - \cos(w)) + w^2 - 4w \sin(w) > 0 \) as \( \sin(w) \) is negative in this range. Since \( w^2 > 4w \sin(w) \) for \( w > 4 \) this gives \( 6(1 - \cos(w)) + w^2 - 4w \sin(w) > 0 \) for all \( w > 0 \). Thus, since \( \phi_1 \) is positive and \( \hat{\phi}_1 > 0 \) for all \( w \geq 0 \) then \( \phi_1 \in \text{PD}_1 \).

For \( \phi_1 \in \text{PD}_3 \) it is required by Proposition 3.6 that \( D\hat{\phi}_1 > 0 \). Now

\[
D\hat{\phi}_1(w) = \frac{1}{2} \frac{d}{dw} \frac{6(1 - \cos(w)) + w^2 - 4w \sin(w)}{w^4} = 2 \left( \frac{12(1 - \cos(w)) + w^2(1 + 2 \cos(w)) - 9w \sin(w)}{w^6} \right).
\]
CHAPTER 3. COMPACTLY SUPPORTED FUNCTIONS

Now when \( w = 9 \), \( D\dot{\phi}_1(w) \approx -0.0003 \) so that \( D\dot{\phi}_1 \neq 0 \) and therefore \( \phi \notin \text{PD}_3 \). Figures 3.1 and 3.2 show the behaviour of \( \phi_1 \) and \( D\dot{\phi}_1 \).

Table 3.4 shows the dimension of positive definiteness for various values of \( \alpha \). For values \( \alpha \geq 2 \) then \( \phi'_\alpha(0) \) becomes nonnegative, and therefore \( \phi_\alpha \) cannot be \( \text{PD} \).

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>( \phi \in \text{PD}_d )</th>
<th>( \alpha )</th>
<th>( \phi \in \text{PD}_d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3</td>
<td>0.51</td>
<td>1</td>
</tr>
<tr>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
</tr>
<tr>
<td>0.5</td>
<td>3</td>
<td>\geq 2</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 3.4: Dimension of positive definiteness.

Figure 3.1: Values of \( \dot{\phi}_1 \)
Figure 3.2: Values of $D\phi_1$
3.5 Polynomials of minimal degree

At the end of Wu’s paper [60], the problem is set of finding the cut-off polynomial of lowest degree in $\text{CS} \cap C^{2k} \cap \text{PD}_d$. Wendland polynomials (3.8) solve this problem. The main theoretical result of Wendland in [56] is given in the following proposition.

**Proposition 3.28** Up to a constant factor there exists exactly one $\text{CS} \cap \text{PD}_d$ polynomial $\phi$ of degree $\partial \phi = [d/2] + 3k + 1$ and satisfying $\phi \in C^{2k}$. Where $[d]$ means the integer $n$ satisfying $n \leq d < n + 1$.

Wendland provides a method for obtaining the coefficients of these polynomials of minimal degree. Some notation must first be introduced. Let

$$[\alpha]_{-1} = \frac{1}{\alpha + 1}, \quad [\alpha]_0 = 1, \quad [\alpha]_l = \alpha(\alpha - 1) \cdots (\alpha - l + 1), \quad \alpha \geq l - 1,$$

and

$$(\nu)_0 = 1, \quad (\nu)_l = \nu(\nu + 1) \cdots (\nu + l - 1).$$

Now

$$\phi_{l,k}^{\text{We}}(r) = \sum_{n=0}^{k} \beta_{n,k} r^n (1 - r)^{l + 2k - n}$$

where the coefficients satisfy the recursion

$$\beta_{0,0} = 1$$

$$\beta_{j,k+1} = \sum_{n=j-1}^{k} \beta_{n,k} \frac{[n + 1]_{n-j+1}}{(l + 2k - n + 1)_{n-j+2}}, \quad 0 \leq j \leq k + 1$$

if the term for $n = -1$ for $j = 0$ is ignored.

An alternative approach to finding these coefficients arises as a consequence of Theorem 3.26. Assume $\phi$ is a polynomial in $\text{CS} \cap C^{2k} \cap \text{PD}_d$, then from Lemma 3.24 the $C^{2k}$ condition must be applicable in the neighbourhood of the
origin. Then \( \phi \in \text{CS} \cap C^{2k}(0) \cap C^l(1) \cap \text{PD}_d \) for \( l \geq 2k \) with \( d \leq 2(l - 2k) + 1 \) from Theorem 3.26. Therefore

\[
l \geq \frac{d - 1}{2} + 2k = \lfloor d/2 \rfloor + 2k
\]

so \( \phi \) is at least \( C^{\lfloor d/2 \rfloor + 2k}(1) \), which produces \( \lfloor d/2 \rfloor + 2k + 1 \) constraints on the coefficients of \( \phi \). It is known that \( \phi \in C^{2k}(0) \) and \( \phi(0) > 0 \), which naturally enforces a further \( k + 1 \) constraints. This means that for \( \phi \in \text{CS} \cap C^{2k} \cap \text{PD}_d \) a total of

\[
\lfloor d/2 \rfloor + 3k + 2
\]

(3.29)

conditions are forced upon the choice of coefficients of \( \phi \). However, from Proposition 3.28 a unique polynomial exists of degree \( \lfloor d/2 \rfloor + 3k + 1 \). Such a polynomial can be determined by finding \( \phi \) which satisfies the conditions (3.29). The coefficients of \( \phi \) form the solution vector to the system of linear equations given in the following theorem.

**Theorem 3.30** Wendland polynomials can be recovered by solving the following system of linear equations.

\[
\phi_{l,k}^{\text{We}}(\tau) = \sum_{n=0}^{l+2k} \alpha_n \tau^{l+2k-n}
\]

where \( \alpha = [\alpha_1, \alpha_2, \ldots, \alpha_k]^T \) is the solution vector to \( A\alpha = g \) with

\[
A = \begin{pmatrix} B \\ C \\ g^T \end{pmatrix} \in \mathbb{R}^{l+2k+1} \times \mathbb{R}^{l+2k+1}
\]

where \( B \in \mathbb{R}^{l+k} \times \mathbb{R}^{l+2k+1} \) has elements

\[
B_{1 \leq i \leq l+k, 1 \leq j \leq l+2k+1} = \begin{cases} (l + 2k - i - j + 1)_{i+1} \\ (l + 2k - j + 1)! \\ (l + 2k - i - j + 2)! \end{cases}
\]
$C \in \mathbb{R}^k \times \mathbb{R}^{l+2k+1}$ has one nonzero entry in each row given by

$$C_{i,i+2(k-i+1)} = (2i-1)! \text{ for } 1 \leq i \leq k$$

and $g = [0,0,\ldots,0,1]^T$.

In Theorem 3.30, the matrix $B$ deals with the smoothness of $\phi$ around the cut-off, $C$ is concerned with smoothness at the origin, and the nonzero entry in $g$ determines the value $\phi(0)$.

Computational advantages are not important in the construction of these polynomials since the coefficients only need to be obtained once. So Theorem 3.30 is given for interest and not intended to compete with Wendland’s method.

### 3.6 Conclusion

It was hoped that compactly supported radial basis functions could be classified by their smoothness. However it is necessary to look at the univariate Fourier transform to determine the dimension of positive definiteness of a given function. Such information is crucial when designing compactly supported basis functions to a given specification. The smoothness properties can only provide a bound on this dimension.
Chapter 4

Creating a native space

Error analysis for translates of a smooth basis function $\Phi$ relies on the initial data itself being smooth. This results from the native space associated with $\Phi$ being restrictively small. The aim of this chapter is to create a new native space from an approximation to $\Phi$, which allows a wider class of functions to be admitted. The standard error bounds in Section 1.3 can then be applied. The approximating basis function $\Psi$ can be arbitrarily close to $\Phi$ on a fixed interval, therefore ensuring that the computed numerical interpolants arising from $\Phi$ and $\Psi$ are the same.

The work of Yoon [61] is described where the initial data $f$ is presumed to lie in a particular Sobolev space. Error bounds for a stationary method are then achieved by splitting the expression $(f - s_f)$ into components which are individually treated in the Sobolev space setting.

4.1 Introduction

For this brief summary of Yoon’s work [61], let $s_{f, X}$ denote the interpolant comprising of a linear combination of translates of $\phi_w = \phi(\cdot/w)$, to the given function $f$ at the points in the set $X$. The parameter $w = w(h)$ depends on $h$ such that
PROP. 4.1 For every function \( f \in W^k_\infty(\Omega) \) with \( k \) a positive integer, there exists an error bound of the form
\[
\| f - s_{f,X} \|_{L^\infty(\Omega)} = o(h^k)
\]
for \( h \rightarrow 0 \).

This bound is obtained by introducing a radially symmetric \( C^\infty \) cut-off function \( \sigma_\delta \), which has support on \([-\delta, \delta]^d\). A mollification of \( f \) is defined by
\[
f_H = \sigma_\delta(h \cdot)^\vee \ast f
\]
and \( (f - s_{f,X}) \) is split into components
\[
f - s_{f,X} = (f_H - s_{f_H,X}) + (f_T - s_{f_T,X}) \tag{4.2}
\]
where
\[
f_T = f - \sigma_\delta(h \cdot)^\vee \ast f.
\]
The function \( f_H \) has the useful property that \( \hat{f}_H = \sigma_\delta \hat{f} \), which is then compactly supported. It then remains to bound the two components of (4.2), both of which are shown to have asymptotic behaviour like \( o(h^k) \).

Since \( f_H \in \mathcal{G}_\phi \) then the usual error bound can be applied to \( \| f_H - s_{f_H,X} \|_{L^\infty(\Omega)} \).

PROP. 4.3 Let \( s_{f_H,X} \) interpolant \( f_H \) at the points in \( X \) using the basis function \( \phi_w \). Let \( w \) be a parameter depending on \( h \), so \( w = w(h) \), and \( \phi_w = \phi(\cdot/w) \). Then, for every \( f \in L^2(\mathbb{R}^d) \) there exists an estimate of the form
\[
|f_H(x) - s_{f_H,X}(x)| \leq P_{\phi,X/w}(x/w) M_{\phi,w}(\delta/h) \| f \|_{L^2(\mathbb{R}^d)}
\]
for all \( x \in \Omega \) as \( h \rightarrow 0 \) where \( P_{\phi,X/w} \) is the power function, and \( M_{\phi,w} \) is defined by
\[
M_{\phi,w}(r) = \sup_{\theta \in B_r} |\hat{\phi}_w(\theta)|^{-1/2}
\]
where \( B_r \) is the Euclidean ball of radius \( r \).
It is noted that the value $\delta$ can be chosen arbitrarily, therefore it is assumed that there exists a $\delta_0$ such that

$$P_{\phi,x/w}(x/w)M_{\phi,w}(\delta_0/h) \leq o(h^k)$$

for every $f \in W^k_\infty(\Omega)$. Therefore the $(f_h - s_{f_\delta,x})$ component is bound by $o(h^k)$.

The construction of $f_T$ does not ensure that it belongs to the native space $G_\phi$. A new basis is introduced and the interpolant $s_{f_T}$ is reformulated to allow the application of similar bounds used in Proposition 4.3.

**Proposition 4.4** For every $f \in W^k_\infty(\Omega)$ with positive integer $k$ there exists an error bound of the form

$$|f_T(x) - s_{f_T,x}(x)| \leq o\left(h^k(1 + P_{\phi,x/w}(x/w)M_{\phi,w}(\delta/h))\right)$$

for every $x \in \Omega$ as $h \rightarrow 0$.

The combination of these two bounds proves Proposition 4.1. Examples are then given of the appropriate weights $w$ which apply to multiquadrics, polyharmonic splines, and inverse multiquadric basis functions. The gaussian case is not covered by this method.

It should be noted that the given error bounds only apply to stationary methods of interpolation, where the scale of the basis function is adjusted as a function of $h$. A technique is presented in the following section whereby error bounds are obtained in the nonstationary setting using PD basis functions.

### 4.2 Designing a basis function

The objective is to design a new basis function $\Psi$ which returns the same numerical interpolant as $\Phi$, but which has a significantly different Fourier transform to $F_d(\Phi)$. The following proposition will be proved in the remaining sections.
Proposition 4.5 Let $\Phi$ be a gaussian or generalised inverse multiquadric with $\beta + d < 0$. Then for a domain $\Omega \subseteq \mathbb{R}^d$ there exists a function $\Psi$ such that

(i) $|\Psi(x - y) - \Phi(x - y)| < \delta$ for $y, x \in \Omega$ and arbitrarily small $\delta$

(ii) $\Psi \in \text{CS} \cap \text{PD}_{d+2}$

(iii) there exists $c_1, c_2, m, \rho \in \mathbb{R}$ such that $c_1 \leq \mathcal{F}_d(\Psi)(w) \cdot \|w\|^{d+\rho} \leq c_2$

for all $\|w\| > m$.

Let $s_\Phi$ and $s_\Psi$ interpolate the same data using translates of their respective basis functions. Since the difference between $\Phi$ and $\Psi$ is arbitrarily small, this can be specified such that they are identical up to working precision $\varepsilon$. Then the associated interpolation matrices, and therefore resultant coefficients, will be identical when computing $s_\Phi$ and $s_\Psi$. For example, if the working precision is set to $10^{-16}$, then there exists a $\delta \ll 10^{-16}$, such that the two interpolation matrices cannot differ within the first 16 decimal places. Therefore there exists a $\delta < \varepsilon$ such that

$$|s_\Psi(x) - s_\Phi(x)| < \varepsilon \text{ for all } x \in \Omega$$

and

$$\|f - s_\Psi\|_{L_\infty(\Omega)} - \|f - s_\Phi\|_{L_\infty(\Omega)} < \varepsilon.$$  \hspace{1cm} (4.6)

With $\varepsilon$ set to working precision, backward error analysis can provided the necessary value for $\delta$; see Higham [54]. The notion of $\varepsilon$ set to working precision does pollute the purity of the theory, nevertheless the aim is to explain observational data.

The norms $\|\cdot\|_\Psi$ and $\|\cdot\|_\Phi$ will be significantly different. This is not important since it is the convergence rate which is required, and not the magnitude of the error.

As a result of Proposition 4.5 (ii), the error bounds in Section 1.3 can be applied to functions in the native space $\mathcal{G}_\Phi$. Note that $\Psi$ is PD in 2 space dimensions higher than which the interpolation problem is set. This is specified such that $\mathcal{F}_d(\Psi)$ can be easily bounded below.
CHAPTER 4. CREATING A NATIVE SPACE

Due to the freedom to manipulate functions outside of the domain $\Omega \subseteq \mathbb{R}^d$, let $f^0$ be a compactly supported function such that $f^0(x) = f(x)$ on $x \in \Omega$, then

$$f(x) - s_\Phi(x) = f^0(x) - s_\Phi(x) \quad \text{for} \quad x \in \Omega.$$  \hspace{1cm} (4.7)

Further assume that if $f \in C^k(\Omega)$ then $f^0 \in C^k(\mathbb{R}^d)$. Note that the Fourier transforms $\mathcal{F}_d(f)$ and $\mathcal{F}_d(f^0)$ will be different. It is now possible to view $\mathcal{G}_\Psi$ as containing functions $f^0$ whose support encloses $\Omega$. Hence the native space $\mathcal{G}_\Psi$ contains compactly supported versions of functions $f \in \mathcal{G}_\Psi$. Furthermore, genuine functions of limited smoothness are allowed in $\mathcal{G}_\Psi$ which would have been excluded from $\mathcal{G}_\Phi$. The aim is to find an error bound to the problem of interpolating $f \in C^k(\Omega)$ by translates of $\Phi$. This is rewritten in terms of $f^0$ and $\Psi$, where $\Psi$ has to be found such that $f^0 \in \mathcal{G}_\Psi$.

The native space $\mathcal{G}_\Psi$ can be realised from (1.20), given here for convenience

$$\mathcal{G}_\Psi = \left\{ f^0 : \int_{\mathbb{R}^d} \left( \mathcal{F}_d(f^0)(w) \right)^2 / \mathcal{F}_d(\Psi)(w) \, dw < \infty \right\}. \hspace{1cm} (4.8)$$

Before $\mathcal{G}_\Psi$ can be determined explicitly, a number of results are required.

**Lemma 4.9** For every even, compactly supported function $g \in C^b(\mathbb{R})$ there exists a constant $c$ such that $|\hat{g}(w)| \leq c/w^{b+2}$ for large $w$; see Folland [21] and Morrison [41].

This result can then be extended to higher dimensions.

**Lemma 4.10** For every radial, compactly supported function $g \in C^b(\mathbb{R}^d)$ there exists a positive constant $c$ such that $|\mathcal{F}_d(g)(w)| \leq c/||w||^{b+d+1}$ for large $w$ and odd $d$.

**Proof:** Since $g \in \text{CS}$ then Proposition 3.5 (ii) can be employed to give

$$\mathcal{F}_d(g) = \mathcal{F}_1 \left( t^{d+1} g \right).$$
The application of $I$ causes $(I^{d+1}_{-\frac{1}{2}} g) \in C^{0} L^{1} (\mathbb{R})$ from Lemma 3.23. Now Lemma 4.9 yields

$$|\mathcal{F}_d(g)| = |\mathcal{F}_1 (I^{d+1}_{-\frac{1}{2}} g)| \leq c / ||w||^{b+d+1}.$$ 

**Lemma 4.11** The integral $\int_{\mathbb{R}^d \setminus B_R} ||w||^\alpha dw$ is finite when $(\alpha + d) < 0$, where $B_R \subset \mathbb{R}^d$ is the Euclidean ball of radius $R$.

**Proof:** Let $S_{d-1}(r)$ be a hyperspherical shell in $\mathbb{R}^d$ with radius $r$. Then the surface area of $S_{d-1}(r)$ is proportional to $r^{d-1}$. Therefore

$$\int_{\mathbb{R}^d \setminus B_R} ||w||^\alpha dw = \int_{r}^{\infty} \int_{S_{d-1}(r)} ||w||^\alpha dw dr$$

$$= 2\pi^{d/2} / \Gamma(d/2) \cdot \int_{r}^{\infty} r^{\alpha + d - 1} dr$$

$$= 2\pi^{d/2} / \Gamma(d/2) \cdot \lim_{M \to \infty} \left[ \frac{r^{\alpha + d}}{\alpha + d} \right]_r^M$$

which is finite when $\alpha + d < 0$. \qed

It was hoped that the asymptotic decay of $\mathcal{F}_d(f^0)$ could be determined exactly by the smoothness of $f^0$. However, only such a relationship has been discovered for radial functions as in Lemma 4.10. Therefore assume that $\mathcal{F}_d(f^0)(w) = \mathcal{O}(||w||^{-d-j})$ for large $||w||$. Since both $f^0$ and $\Psi$ are compactly supported then $\mathcal{F}_d(f^0)(0)$ and $\mathcal{F}_d(\Psi)(0)$ are finite. Therefore the integral in (4.8) need only be considered on the region $\mathbb{R}^d \setminus B_R$, for some radius $R > 0$. Using Proposition 4.5 (iii), the native space $\mathcal{G}_\Psi$ is given by

$$\mathcal{G}_\Psi = \left\{ f^0 : \int_{\mathbb{R}^d} (\mathcal{F}_d(f^0)(w))^2 / \mathcal{F}_d(\psi)(w) dw < \infty \right\}$$

$$= \left\{ f^0 : \int_{\mathbb{R}^d \setminus B_R} (c/||w||^{d+j})^2 / \mathcal{F}_d(\psi)(w) dw < \infty \right\}$$

$$= \left\{ f^0 : \int_{\mathbb{R}^d \setminus B_R} (c^2/||w||^{2d+2j}) \cdot ||w||^{d+p} / c_1 dw < \infty \right\}$$

$$= \left\{ f^0 : \int_{\mathbb{R}^d \setminus B_R} ||w||^{p-d-2j} dw < \infty \right\}.$$
From Lemma 4.11 this final integral is finite when \((\rho - 2j) < 0\). Therefore \(\Psi\) must be found with \(\rho = 2j - 1\) so that \(f^0 \in \mathcal{G}_\Psi\). These results combined with the asymptotic error bound in (1.19) provide the following.

**Lemma 4.12** If \(\mathcal{F}_d(f^0)(w) = \mathcal{O}(\|w\|^{-d-j})\) for large \(\|w\|\), and \(\Psi\) exists from Proposition 4.5 with \(\rho = 2j - 1\), then

\[
\|f^0 - s\Psi\|_{L\infty(\Omega)} \leq C \cdot h^{j - \frac{1}{2}} \cdot \|f^0\|_\Psi.
\]  

(4.13)

Lemma 4.10 shows the connection between the smoothness of \(\Psi\) and the asymptotic behaviour of \(\mathcal{F}_d(\Psi)\). Therefore in order for \(\rho = 2j - 1\), this is satisfied when \(\Psi \in C^{2(j-1)}(\mathbb{R}^d)\). Hence the approximating basis function \(\Psi\) must be of specific smoothness, and satisfy the conditions in Proposition 4.5.

### 4.3 Constructing a basis function

The following two sections describe the construction process of finding an approximation \(\Psi\) to a given smooth basis function \(\Phi\), in accordance with Proposition 4.5.

An obvious solution would be to truncate \(\Phi\), and attach an extension of specific smoothness which preserves positive definiteness. However, such an extension does not exist since the PD\(_\infty\) basis functions \(\Phi\) are infinitely smooth around the origin; see Lemma 3.24. For this reason it is impossible to find a \(\Psi\) which matches \(\Phi\) exactly on some interval, and which remains PD. Generating a sequence of functions \(\Psi_k\) which converge to the conditions in Proposition 4.5 has been investigated. It has not been possible however to control both the shape and support of the approximating functions. In order for a fixed function \(\Phi\) to be approximated, the support of \(\Psi_k\) has to grow, which undesirably causes \(\mathcal{F}_d(\Psi_k)\) to tend to \(\mathcal{F}_d(\Phi)\). Therefore the argument of matching \(\Phi\) up to machine precision is required. It is useful to consider the univariate functions \(\phi(||x||) = \Phi(x)\) and
\( \psi(||x||) = \Psi(x) \). Let \( q \) be the maximum separation distance of \( \Omega \), then condition (i) in Proposition 4.5 can be viewed as \( |\psi(r) - \phi(r)| < \varepsilon \) for \( r \in [0, 2q] \).

It will be shown that functions of the form

\[
\psi = D^k(v * v)
\]

for some generating function \( v \) will satisfy the conditions in Proposition 4.5. The construction is similar to that of Wu [60] as described in Chapter 3. Section 4.3.1 shows the approximating properties of \( \psi \), and Section 4.3.2 deals with the behaviour of \( \mathcal{F}_d(\psi) \).

### 4.3.1 Approximating \( \phi \)

The Central Limit Theorem offers a simple formulation for approximating the gaussian, whereas other basis functions such as the generalised inverse multi-quadric require more detailed methods.

It is interesting to note that Wu's polynomials are themselves good approximations to a gaussian. Figure 4.1 shows a selection of scaled Wu polynomials which closely match \( e^{-r^2} \). Unfortunately each unscaled \( \phi_{i,k}^{\text{Wu}} \) approximates a differently shaped gaussian, and so a sequence of the form \( \{\phi_{1,k}^{\text{Wu}}, \phi_{2,k}^{\text{Wu}}, \ldots\} \) would not converge to \( e^{-\beta r^2} \) for a fixed \( \beta \).

The simplest way to obtain smooth \( \mathbf{PD}_1 \) functions is by the convolution of a function with itself, thus

\[
\chi(r) = f * f(r) = \int_{-\infty}^{\infty} f(r - t)f(t)dt
\]

then

\[
\hat{\chi} = (\hat{f})^2.
\]

**Lemma 4.15** *If the function \( f \in C^b(\mathbb{R}) \) and \( f^{(b+1)} \) is piecewise continuous, then* \( f * f \in C^{2b+2}(\mathbb{R}) \).
Figure 4.1: Scaled Wu polynomials compared to $e^{-r^2}$.

**Proof:** From the definition of convolution

$$\frac{d}{dx}(f * g) = \frac{df}{dx} * g = f * \frac{dg}{dx}.$$  

Therefore

$$\frac{d^{2b+2}}{dx^{2b+2}}(f * f) = f^{(b+1)} * f^{(b+1)}$$

$$= u * u$$

$$\in C^0(\mathbb{R})$$

where $u$ is a discontinuous function. Therefore $(f * f)$ has $(2b + 2)$ continuous derivatives. □

**Theorem 4.16 Central Limit Theorem**

The mean of random variables $X_1, \ldots, X_n$ tends to a normal distribution as $n \to \infty$; see Weisstein [55].
A consequence of the above theorem is that the repeated convolution of a function with itself converges to a gaussian distribution. Furthermore, the Fourier transform of a gaussian is also a gaussian. This explains why the polynomials of Wendland and Wu look like scaled gaussians. The products \((1 - r)^l\) or \((1 - r^2)^l\) in these constructions cause repeated convolutions of \(F_l(1 - r)\) or \(F_l(1 - r^2)\) in the Fourier domain. Therefore the Fourier transform converges to a gaussian distribution.

Let \(T_m(g)\) denote the function \(g\) convolved with itself \(m\) times, that is

\[
T_m(g) = g * g * \cdots * g
\]

Let \(g\) be a discontinuous cut-off generating function and define

\[
\varphi = T_{2m-1}(g)
\]

for \(m \geq 1\). From repeated application of Lemma 4.15, if \(g\) is discontinuous then \(\varphi \in C^{2m-2}(\mathbb{R})\).

From the Central Limit Theorem, any discontinuous function \(g\) will suffice to obtain a gaussian distribution. It is a disadvantage however to rely on repeated convolutions to produce an approximation to \(\phi\), since this operation heavily affects the smoothness. To accelerate convergence let \(g\) be the cut-off gaussian \(e_t^{-r^2}\) which has support on \(r \in [0,t]\) and where \(\sigma = 2^{2m-2}\). Note that \(e_t^{-r^2} * e_t^{-r^2} = e_{2t}^{-\frac{1}{2}r^2}\), therefore \(\varphi\) has support on \([0,2mt]\) and approximates \(e^{-\frac{1}{2}r^2}\) around the origin. Consider the case when \(m = 1\), and define \(\varphi = \frac{\sqrt{\pi}}{2} \cdot g * g\). To find \(\varphi\) explicitly the following result is required.

**Lemma 4.17** From [7.4.32] in [1]

\[
\int e^{-(ax^2+2bx+c)}dx = \frac{1}{2\sqrt{a}} e^{\frac{b^2-ac}{a}} \text{erf} \left( \sqrt{ax + \frac{b}{\sqrt{a}}} \right) + \text{const}
\]

for \(a \neq 0\), where

\[
\text{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-s^2} ds.
\]
Then

\[ \varphi(r) = \frac{\sqrt{\pi}}{2} \cdot e^{-r^2} \ast e^{-r^2} \]
\[ = \frac{\sqrt{\pi}}{2} \int_{-t+r}^{t} e^{-(r-\tau)^2} \cdot e^{-\tau^2} d\tau \]
\[ = \frac{\sqrt{\pi}}{2} \cdot e^{-r^2} \int_{-t+r}^{t} e^{-2\tau^2+2r\tau} d\tau \]
\[ = \frac{1}{2} e^{-\frac{1}{2}r^2} \left[ \text{erf} \left( \sqrt{2}(\tau - r/2) \right) \right]_{-t+r}^{t} \]
\[ = \frac{1}{2} e^{-\frac{1}{2}r^2} \left[ \text{erf} \left( \sqrt{2}(t - r/2) \right) - \text{erf} \left( \sqrt{2}(-t + r/2) \right) \right] \]
\[ = e^{-\frac{1}{2}r^2} \cdot \text{erf} \left( \sqrt{2}(t - r/2) \right) . \]

The error between the target function and \( \varphi \) is given by

\[ E_t(r) = e^{-\frac{1}{2}r^2} - \varphi(r) \]
\[ = e^{-\frac{1}{2}r^2} \cdot \text{erfc} \left( \sqrt{2}(t - r/2) \right) \]

where \( \text{erfc}(z) = 1 - \text{erf}(z) \). From [7.1.16] in [1] \( \text{erfc}(z) \to 0 \) as \( z \to \infty \), so for a fixed interval \( r \in [0, 2q] \) the error \( E_t(r) \to 0 \) as \( t \to \infty \). Then there exists a \( t_* \) such that \( \|E_{t_*}\|_{\infty} < \varepsilon \) as required. The convergence of \( E_t \) is rapid due to the \( \text{erfc}(z) \) term; see Example 4.18 and Figure 4.2.

**Example 4.18**

This example is given to demonstrate the speed at which \( \varphi \) converges to \( e^{-\frac{1}{2}r^2} \). Let \( \varphi(r) = \frac{\sqrt{\pi}}{2} \cdot e^{-r^2} \ast e^{-r^2} \). Table 4.1 shows the maximum error \( \|\varphi(r) - e^{-\frac{1}{2}r^2}\|_{L_{\infty}([0,1])} \) for different values of \( t \).

Application of the \( D \) operator is required to further manipulate the Fourier transform in accordance with Theorem 3.19. For \( m \geq 2 \) the convolution of \( \varphi = T_{2m-1}(g) \) ensures that there are at least \( (2^m - 2) \) continuous derivatives.
around the origin and in the neighbourhood of the cut-off. This is sufficient
smoothness for $\psi = D^k(T_{2m-1}(g))$ to be a valid operation from Lemma 3.23 for
$k \leq (2^{m-1} - 1)$. As will be shown in Section 4.3.2, $\varphi$ can be at most $PD_1$, therefore
from Proposition 3.5 (iii) $\psi$ can be at most $PD_{2k+1}$. In the case of the gaussian
there exists the convenient relation that $D(e^{-\beta r^2}) = 2\beta \cdot e^{-\beta r^2}$. Furthermore the error

$$D(E_t)(r) = -\frac{1}{r} \frac{d}{dr} \left\{ e^{-\frac{1}{2}\tau^2} \cdot \text{erfc}(\sqrt{2}(t - r/2)) \right\}$$

$$= e^{-\frac{1}{2}\tau^2} \cdot \text{erfc}(\sqrt{2}(t - r/2)) - \frac{e^{-\frac{1}{2}\tau^2}}{r} \frac{d}{dr} \left\{ 1 - \frac{2}{\sqrt{\pi}} \int_0^{\sqrt{2}(t-r/2)} e^{-s^2} ds \right\}$$

$$= E_t(r) - \frac{\sqrt{2}}{\sqrt{\pi} r} e^{-\frac{1}{2}\tau^2} \cdot e^{-2(t-r/2)^2}$$
Table 4.1: Maximum error in the construction of $\varphi$.

<table>
<thead>
<tr>
<th>$t$</th>
<th>Maximum error on $[0, 1]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>$1.92 \times 10^{-1}$</td>
</tr>
<tr>
<td>1.5</td>
<td>$2.76 \times 10^{-2}$</td>
</tr>
<tr>
<td>2.0</td>
<td>$1.64 \times 10^{-3}$</td>
</tr>
<tr>
<td>2.5</td>
<td>$3.84 \times 10^{-5}$</td>
</tr>
<tr>
<td>3.0</td>
<td>$3.48 \times 10^{-7}$</td>
</tr>
<tr>
<td>5.0</td>
<td>$1.37 \times 10^{-19}$</td>
</tr>
</tbody>
</table>

vanishes as $t \to \infty$ for a fixed $r \in [0, 2q]$. Therefore $D^k(E_t)(r) \to 0$ as $t \to \infty$ for fixed $r \in [0, 2q]$.

So repeated convolutions are used to obtain the required smoothness, the $D$ operator then creates the correct dimension of positive definiteness without affecting the shape of $\varphi$, and the support of the generating function $g$ ensures the necessary approximation to a gaussian. This is summarised in the following Theorem.

**Theorem 4.23** Let $g$ be the cut-off gaussian $e^{-(r^2)/\sigma^2}$ where $\sigma = 2^{2m-2}$ with support on $r \in [0,t_j]$, and define $\psi = D^k(T_{m-1}(g))$. Then $\psi \in C^{2m-2(k+1)}(\mathbb{R})$ and for arbitrarily small $\epsilon$ there exists a $t_\ast$ such that $|\psi(r) - e^{-\frac{1}{4}r^2}| < \epsilon$ and $r \in [0, 2q]$. Furthermore $\psi \in CS \cap PD_{2k+1}$.

In the unified setting (4.14) where $\psi = D^k(u * v)$, this requires $v = T_{m-1}(g)$.

The Central Limit Theorem allowed the use of convolution to approximate a gaussian. In the case of other basis functions, the convolution process ought only to be used to obtain positive definiteness, otherwise convergence to a gaussian
distribution is inevitable. In particular, consider the generalised inverse multi-quadric
\[ \phi(r) = (\sigma^2 + r^2)^{\beta/2} \] for \( \beta < 0 \).

The function \( \psi \) will be constructed via
\[ \psi = D^k(v * v) \] (4.24)

for \( k \geq 1 \), where \( v \) has support on \([0, t + 1]\) and \((m - 1)\) continuous derivatives so that from Lemma 4.15 \( \psi \in C^{2(m-k)}(IR) \). For clarity let \( \chi = v * v \), so \( \psi = D^k\chi \).

In order to determine \( v \), the following result is required.

**Lemma 4.25** If \( \phi \in C^0(IR) \cap PD \cap L_1 \) is an even function, then there exists a generating function \( \theta \) such that \( \phi = \theta * \theta \). Moreover, if \( \phi \in C^\infty(IR) \) then \( \theta \in C^\infty(IR) \) and \( \theta(r) \longrightarrow 0 \) as \( r \longrightarrow \infty \).

**Remark 4.26** However, it may be impossible to obtain an explicit expression for \( \theta \).

**Proof:** Assume \( \phi = \theta * \theta \), then by the action of convolution in the Fourier domain
\[ \hat{\theta} = \sqrt{\hat{\phi}}. \]

Now in 1 dimension, an even function \( \phi \) has the property that \( \hat{\phi} \hat{\psi} = \hat{\phi} \) where \( \hat{\psi} \) denotes the inverse Fourier transform. Therefore
\[ \theta = \left[ \sqrt{\hat{\phi}} \right]^\wedge. \]

A sufficient condition for \( \theta \) to exist is that
\[ \sqrt{\hat{\phi}} \in L_2 \]

which implies that
\[ \left( \int_R |\hat{\phi}(w)| \, dw \right)^{\frac{1}{2}} < \infty. \]
So it is required that $\hat{\phi}(0)$ is finite and $\hat{\phi}(w)$ decays at least like $1/w^2$ for large $w$.

The first condition is satisfied since $\phi \in L_1$, and the later is true if $\phi \in C^0(\mathbb{R})$; see Lemma 4.9.

In order to produce an infinitely differentiable $\phi$ then clearly $\theta \in C^\infty(\mathbb{R})$ from the definition of convolution. Further, since $\phi(r) \to 0$ as $r \to \infty$ then $\theta(r)$ must also tend to zero as $r$ increases. □

From the construction (4.24) it can be seen that a suitable $v$ must be found such that $\chi \approx I^k \phi$ on a fixed interval. In the case of the generalised inverse multiquadric, this restricts which functions can be approximated. To ensure that $v$ exists from Lemma 4.25, $I^k \phi$ must be both PD and in $L_1$. Now

$$I^k((c^2 + r^2)^{\beta/2}) = \frac{(-1)^k}{\prod_{i=1}^{k}(\beta + 2i)}(c^2 + r^2)^{\beta/2+k}$$

which requires $\beta + 2k + 1 < 0$ to ensure that $I^k \phi \in \text{PD} \cap L_1$. The familiar case when $\beta = -1$ is therefore redundant in this setting.

Assume that the function $\theta$ exists from Lemma 4.25. The generating function $v$ is then a truncated version of $\theta$ with support on $[0, t]$, which is extended to $[0, t+1]$ by the addition of an arbitrarily smooth pulse $p$ with specific smoothness $p \in C^{m-1}(t+1)$. The extension $p$ is determined such that $\|p\|_{\infty} = \theta(t)$ and $p^{(j)}(t+1) = -\delta_{m,j}$, for $j = 0, 1, \ldots, m$; see Figure 4.3. Let $P$ denote the contribution from $p$ in the convolution $v \ast v$, then

$$P(r) = \int_{-(t+1)+r}^{-t+r} p(s-r)v(s) \, ds + \int_{t+r}^{t+1} p(s-r)v(s) \, ds.$$ 

This can be bounded thus

$$|P(r)| \leq \left| \int_{-(t+1)+r}^{-t+r} v(t)v(s) \, ds \right| + \left| \int_{t+r}^{t+1} v(t)v(s) \, ds \right|$$

$$\leq 2 v_{\max} v(t),$$

where

$$v_{\max} = \max_{r \in \mathbb{R}} |v(r)|.$$
Figure 4.3: The function $v$ for $t = 1, m = 1$.

Since $v(t) \to 0$ as $t \to \infty$ from Lemma 4.25, then $|P(r)|$ vanishes for fixed $r$ and large $t$. Then $\chi(r) \to I^k \phi(r)$ on $r \in [0, 2q]$ as $t \to \infty$ as required.

For completeness, a method for finding a suitable extension $p$ is provided. Let $q$ be a polynomial which satisfies the following, $q(t) = \theta(t)$, $q^{(j)}(t+1) = -\delta^j_m$ and $q^{(j)}(t^-) = \theta^{(j)}(t^-)$ for $j = 1, \ldots, m$. Let $Q_0$ be the sum of two gaussians centred at $r = t$ and $r = t + 1$ with shape parameter $\varrho$

$$Q_0(r) = \frac{e^{-\varrho(t-r)^2} + e^{-\varrho(t+1-r)^2}}{1 + e^{-\varrho}}$$

scaled such that $Q_0(t) = Q_0(t + 1) = 1$. Figure 4.4 shows $Q_0$ for different values of $\varrho$ when $t = 1$. Now $p = qQ_0$ preserves the continuity conditions of $q$, whilst $\varrho$ can be adjusted to enforce $\|p\|_{\infty} = \theta(t)$. Note that using this method produces a function $p$ such that $p^{(j)}(t + 1) = \alpha \delta^m_j$ for some $\alpha \in \mathbb{R}_+$, the value of $\alpha$ is not important. The contribution from $p$ in the Fourier domain is significant due to the specified derivative $p^{(m)}(t + 1) = -1$. Since $\theta$ is infinitely smooth around the
origin from Lemma 4.25, the first nonzero term in the repeated integration by parts of \( \dot{v} \) come from the evaluation of \( p^{(m)} \) at \( (t + 1) \). Therefore \( v^{(2j-1)}(0) = 0 \) for \( j \in \mathbb{N} \), and \( v^{(j)}(t + 1) = -\delta_j^m \) for \( j = 1, \ldots, m \). The 1D Fourier transform of \( v \) is determined by a process of integration by parts thus
\[
\dot{v} = \int_0^{t+1} \cos(wr)v(r)dr \\
= \left[ \frac{\sin(wr)}{w}v'(r) \right]_0^{t+1} - \int_0^{t+1} \frac{\sin(wr)}{w}v'(r)dr \\
= 0 - \left[ \frac{\cos(wr)}{w^2}v'(r) \right]_0^{t+1} + \int_0^{t+1} \frac{\cos(wr)}{w^2}v''(r)dr \\
= 0 + \left[ \frac{\sin(wr)}{w^3}v''(r) \right]_0^{t+1} - \int_0^{t+1} \frac{\sin(wr)}{w^3}v'''(r)dr
\]

which terminates when the value \(v^{(m)}(t+1) = -1\) is invoked. Hence

\[
\dot{v}(w) = \frac{(-1)^{[m/2]+1}}{w^{m+1}} \left( \text{trig}(w(t+1)) + \int_0^{t+1} \text{trig}(wr)v^{(m+1)}(r)dr \right)
\]

where

\[
\text{trig}(z) = \begin{cases} 
\sin(z) & \text{for even } m, \\
\cos(z) & \text{for odd } m.
\end{cases}
\]

Recall that \(\chi = v \ast v\), therefore the dominant term in the asymptotic decay of \(\dot{\chi}\) is given by

\[
\frac{\text{trig}(w(t+1))}{w^{2m+2}}.
\]

Thus the inclusion of the pulse \(p\) forces \(\dot{\chi}\) to decay asymptotically at a prescribed rate.

These results are summarised in the following Theorem.

**Theorem 4.28** Let \(v\) be the truncated function \(\theta\) plus the smooth extension \(p\), with support on \([0, t+1]\). Let \(\phi = \theta \ast \theta\) be the generalised inverse multiquadric \((c^2 + r^2)^{\beta/2}\). Then define \(\psi = D^k(v \ast v)\) such that \(\beta + 2k + 1 < 0\) to have support on \([0, t+1]\). Then for arbitrarily small \(\varepsilon\) there exists a \(t_\ast\) such that

\[
|\psi(r) - (c^2 + r^2)^{\beta/2}| < \varepsilon \text{ and } r \in [0, 2q].
\]

Furthermore \(\psi \in \text{CS} \cap \text{PD}_{2k+1}\).  

4.3.2 Bounding $\mathcal{F}_d(\psi)$

The bounds on $\mathcal{F}_d(\Psi)$ in Proposition 4.5 (iii) are here proved. The constructions of the previous section can be unified by selecting the appropriate generating function $v$ such that

$$\psi = D^k \chi$$

where $\chi = v \ast v$. The function $v$ and value of $k$ depend on which basis function $\phi$ is to be approximated. Assume that $\chi \in C^{2l}$ and has support on $[0, T]$. Since the generating functions are infinitely smooth around the origin, then $v \not\in \text{PD}$. Hence there exist distinct values $w^*$ such that $(\hat{v}(w^*))^2 = \hat{\chi}(w^*) = 0$, implying that $\frac{\partial}{\partial w} \hat{\chi}$ is not one signed. From Theorem 3.19 where $\text{PD}$ functions are characterised by the univariate Fourier transforms, $\chi$ can be at most $\text{PD}_1$. Having such zeros in the Fourier transform obviously causes problems when trying to bound $\hat{\chi}$ below in Proposition 4.5 (iii). From Theorem 3.19 in Chapter 3 it can be seen that a function $\psi = D^k \chi$ which is $\text{PD}_{2k+1}$ will overcome this difficulty. This introduces the idea of having a function which is $\text{PD}$ in a higher dimension than which the problem is set. From repeated integration by parts, the dominant term in the 1D Fourier transform of $\chi$ is given by

$$\frac{(\alpha \cdot \cos(wT))^2}{w^{2l+2}}$$

in a similar way to (4.27) for some $\alpha \in \mathbb{R}$ and large $w$. In order to fix the native space $\mathcal{G}_\Psi$ it must be shown that $\mathcal{F}_d(\psi)$ can be bound below by $c_1/\|w\|^{d+\rho}$. The convergence rate of the error estimates is determined by the upper bound $c_2/\|w\|^{d+\rho}$. In the 1 dimensional case when $\psi = D\chi \in C^{2l-2} \cap \text{PD}_3$, then

$$\hat{\psi} = (D\chi)^\wedge = I\hat{\chi}$$

which has the dominant term

$$\int_{w}^{\infty} \frac{(\alpha \cdot \cos(sT))^2}{s^{2l+1}} ds$$
for large $w$. Therefore,
\[
\frac{d}{dw}(D\chi)(w) = -\frac{(\alpha \cdot \cos(wT))^2}{w^{2l+1}}
\]
which leads to
\[
\frac{-\alpha^2}{w^{2l+1}} \leq \frac{d}{dw}(D\chi)(w) \leq 0
\]
for large $w$. Furthermore,
\[
\frac{d}{dw}\left(\frac{1}{w^{2l}}\right) = \frac{-2l}{w^{2l+1}}
\]
so $(\alpha^2/2l)w^{2l}$ is steeper than $(D\chi)^\wedge$ for large $w$, hence there exists a positive constant $c_1$ such that
\[
\frac{c_1}{w^{2l}} \leq (D\chi)^\wedge(w) = \hat{\psi}
\] (4.29)
for large $w$. Therefore $\psi = D\chi$ produces a $\mathbf{PD}_3$ function which approximates $\phi$ and has a positive Fourier transform which can be bounded in $\mathbb{R}^1$. It is then straightforward to create functions which satisfy similar conditions in higher dimensions. Now consider $\psi = D^2\chi \in C^{2l-4} \cap \mathbf{PD}_5$, then from Lemma 3.6
\[
\mathcal{F}_3(\psi)(w) = \mathcal{F}_3(D^2\chi)(w)
\]
\[
= \mathcal{F}_1(ID^2\chi)(w)
\]
\[
= \mathcal{F}_1(D\chi)(w)
\]
\[
\geq c_1/w^{2l}
\]
as in (4.29). Therefore even in higher dimensions the bound is the same. This makes sense in that more applications of $D$ are required to produce $\psi = D^k\chi \in \mathbf{PD}_{2k+1}$, and then an equal number of $I$ operations are employed in the evaluation of the Fourier transform. The above result can be used recursively to prove the following theorem.

**Theorem 4.30** Let $\chi \in C^{2l}$, then $\psi = D^k\chi \in C^{2l-k}$ and there exists a constant $c_1$ such that
\[
\mathcal{F}_{2k-1}(\psi)(w) \geq c_1/\|w\|^{2l}.
\]
for large $w$. Furthermore $\psi \in \mathbf{PD}_{2k+1}$. 
Now convert these values for $k$ into the more familiar setting where the interpolation problem takes place on $\Omega \subseteq \mathbb{R}^d$, then $k = \frac{d+1}{2}$. Hence Theorem 4.30 importantly states that
\[
\psi = D^{\frac{d+1}{2}} \chi \in C^{2l-d-1} \cap \mathbf{PD}_{d+2}
\] (4.31)
and
\[
\mathcal{F}_d(\psi)(w) \geq c_1/\|w\|^{2l}
\]
for large $w$. So the function $\psi$ is required to be PD in 2 dimensions higher than which the interpolation problem is set.

From Lemma 4.10 the upper bound
\[
\mathcal{F}_d(\psi)(w) \leq c_2/\|w\|^{2l}
\]
follows from the smoothness of $\psi$ given in (4.31). Then for Proposition 4.5 (iii) to be fulfilled, set $2l = d + \rho$.

Finally, given initial data $f^0$ whose Fourier transform decays like $\mathcal{O}(\|w\|^{-d-j})$, a basis function $\psi$ can be found for which $c_1 \leq \mathcal{F}_d(\psi)(w) \cdot \|w\|^{d+\rho} \leq c_2$ where $\rho = 2j - 1$. Then $f^0 \in \mathcal{G}_\psi$ and the error bound (4.13) is valid.

### 4.4 Numerical results

These numerical results are given to demonstrate how the pointwise error converges when interpolating nonsmooth data using the gaussian basis function. From Lemma 4.12 one would expect to see convergence of $\mathcal{O}(j - \frac{1}{2})$. Interpolants of the form (1.3) using a gaussian with shape parameter $c = 3000$ are formulated using $n$ regularly spaced points on $\Omega = [0, 1]$. The initial data $f^0$ is given by the following Wendland polynomials scaled and shifted to fit onto $\Omega$.

\[
\phi^\text{We}_{1,0} = (1 - r)_+
\]
\[
\phi^\text{We}_{2,1} = (1 - r)^3(3r + 1)
\]
\[
\varphi_{3,2}^{\text{We}} = (1 - r)^5 (8r^2 + 5r + 1)
\]

From [57], the asymptotic convergence rate of \( \mathcal{F}_d(\varphi_{i,k}^{\text{We}})(w) \) is \( O(\|w\|^{-d-2k-1}) \). Therefore \( j = 2k + 1 \), and the expected convergence of the error bound is \( O(2k + \frac{1}{2}) \).

The approximation error is measured on a selection of points, including the location of the limited smoothness at \( r = \frac{1}{2} \). Figures 4.5, 4.6 and 4.7 show the error converging, and Table 4.2 contains the approximate rates, together with the expected rates for comparison. From the figures it is clear that the error bound in Lemma 4.12 only holds for sufficiently small \( h \).

The same experiment was then conducted on \([0,1]^2\). It is difficult to obtain numerical results due to size and conditioning of linear systems. Maple required 250 decimal place arithmetic to produce the \( IR^2 \) numerics found in Table 4.3.

The evident loss of \( \frac{1}{2} \) in the convergence rate may be due to the fact that \( \psi \) is PD in 2 dimensions higher than which the problem is set. This forces \( \chi \) to be smoother in order to accommodate the extra \( D \) operation, and so adds a further restriction to the behaviour of \( \psi \).
Table 4.3: Convergence rates in $\mathbb{R}^2$.

<table>
<thead>
<tr>
<th>$f^0$</th>
<th>Observed rate</th>
<th>Expected rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_{1,0}^{\text{We}}$</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>$\phi_{2,1}^{\text{We}}$</td>
<td>3</td>
<td>2.5</td>
</tr>
<tr>
<td>$\phi_{3,2}^{\text{We}}$</td>
<td>5</td>
<td>4.5</td>
</tr>
</tbody>
</table>

Figure 4.5: Initial data $\phi_{1,0}^{\text{We}}$ in $\mathbb{R}$. 

Figure 4.6: Initial data $\phi_{2,1}^{\text{We}}$ in $\mathcal{R}$.

Figure 4.7: Initial data $\phi_{3,2}^{\text{We}}$ in $\mathcal{R}$. 
4.5 Conclusion

It is possible to create approximations to smooth PD basis functions which effectively allows standard error analysis to be applied in a less restrictive native space. It is straightforward to see how a similar approach might be used for the multiquadric and other CPD functions. Although it is not clear how such functions might be approximated. The pole at the origin of the Fourier transform of these CPD functions implies that the approximation must be globally supported, which removes some of the useful tools employed throughout Chapters 3 and 4.
Bibliography


