Theory and Applications of the Multiwavelets for compression of Boundary Integral Operators.

Steven Paul Nixon B.Sc.

Institute for Materials Research

School of Computing, Science & Engineering,

University of Salford, Salford, UK.

Submitted in Partial Fulfillment of the Requirements of the Degree of

Doctor of Philosophy, June 2004

Contents

Acknowledgements

I would like to thank my supervisor Professor Sia Amini for his help, support and encouragement. I am also grateful to EPRSC for their financial support. My parents, as always have been very supportive and understanding. Finally, I would like to thank everyone in G11F, in particular, Ali, Rabea, Nina and Pushpa.

Abstract

In general the numerical solution of boundary integral equations leads to full coefficient matrices. The discrete system can be solved in $\mathcal{O}(N^2)$ operations by iterative solvers of the Conjugate Gradient type. Therefore, we are interested in fast methods such as fast multipole and wavelets, that reduce the computational cost to $\mathcal{O}(N \ln^p N)$.

In this thesis we are concerned with wavelet methods. They have proved to be very efficient and effective basis functions due to the fact that the coefficients of a wavelet expansion decay rapidly for a large class of functions. Due to the multiresolution property of wavelets they provide accurate local descriptions of functions efficiently. For example in the presence of corners and edges, the functions can still be approximated with a linear combination of just a few basis functions. Wavelets are attractive for the numerical solution of integral equations because their vanishing moments property leads to operator compression. However, to obtain wavelets with compact support and high order of vanishing moments, the length of the support increases as the order of the vanishing moments increases. This causes difficulties with the practical use of wavelets particularly at edges and corners. However, with multiwavelets, an increase in the order of vanishing moments is obtained not by increasing the support but by increasing the number of *mother* wavelets.

In chapter 2 we review the methods and techniques required for these reformulations, we also discuss how these boundary integral equations may be discretised by a boundary element method. In chapter 3, we discuss wavelet and multiwavelet bases. In chapter 4, we consider two boundary element methods, namely, the *standard* and *non-standard* Galerkin methods with multiwavelet basis functions. For both methods compression strategies are developed which only require the computation of the significant matrix elements. We show that they are $\mathcal{O}(N \log^p N)$ such significant elements. In chapters 5 and 6 we apply the standard and non-standard Galerkin methods to several test problems.

Chapter 1

Introduction

Over the last three-to-four decades it has become popular to reformulate linear second order partial differential equations as integral equations over the boundary of the region of interest. These boundary integral equations are then solved by finite element type discretisations; referred to as boundary element methods (BEM). Our research is concerned with methods for solving boundary integral equations with *almost optimal* efficiency.

There are several advantages to using BEM in place of finite element methods (FEM) applied to the original partial differential equation, see [1]:

- Exterior problems are treated more naturally, since BEM requires meshing over only a finite domain, whereas, FEM requires meshing over an infinite domain. Boundary conditions at infinity can be neatly incorporated into the boundary integral equation reformulation.

- Reformulating the problem on the boundary alone reduces the dimension of the problem by one, resulting in smaller matrices for the same mesh size h.

- BEM allows us to compute the solution only in a subdomain of special interest. When using FEM, the solution must be computed everywhere.

- The matrices formed by BEM are generally better conditioned than those formed by FEM.

There are also disadvantages to BEM:

- FEM can be applied to linear, nonlinear and time-dependent partial differential equations. Boundary element counterparts for more "complicated" partial differential equations have not yet fully developed, although research is underway, eg [2].

- The elements of matrices formed by FEM are easy to compute. By contrast, each element of a BEM matrix involves integration. For diagonal elements, these integrals may be singular.

- The matrices formed by FEM are sparse and can be solved quickly by fast solvers. However, boundary element matrices are full. Traditionally, they are solved by a direct method such as Gaussian elimination. However, we are interested in *fast methods* which reduce the computing time for large scale problems.

Briefly, boundary element methods partition the boundary into N elements. This results in an $N \times N$ system of linear equations. A direct solver such as Gaussian elimination solves

the system in $\mathcal{O}(N^3)$ arithmetic operations. In general, the use of an iterative solver, possibly with preconditioning, results in $\mathcal{O}(N^2)$ operations. However, these methods cannot improve upon an $\mathcal{O}(N^2)$ complexity estimate, since simply forming the coefficient matrix requires $\mathcal{O}(N^2)$ arithmetic operations.

The *fast methods* with which we are concerned aim to solve the boundary integral equation to within the discretisation error in $\mathcal{O}(N \log^p N)$ for some small integer value p; typically $p = 0, 1, 2$. This is the so-called *almost optimal* complexity one can achieve in finding N-unknowns.

A typical (Galerkin) boundary element matrix entry has the form

$$A_{ij} = \int_\Gamma \int_\Gamma K(\mathbf{p}, \mathbf{q}) \psi_i(\mathbf{p}) \psi_j(\mathbf{q}) \; d\Gamma_{\mathbf{q}} d\Gamma_{\mathbf{p}}. \tag{1.0.1}$$

Clearly, in a *fast method* we can not evaluate the whole coefficient matrix A. Currently, there are two distinct classed of fast methods for solving boundary integral equations. One is the so-called *fast multipole algorithm*, closely related to panel clustering [3]; see Profit, Amini & Profit [4, 5] for application to the Helmholtz equation. The basic idea here is that the kernel $K(\mathbf{p}, \mathbf{q})$ of the integral operator is approximated by a degenerate or "separable" kernel

$$K(\mathbf{p}, \mathbf{q}) \approx \sum_{l,m=-1}^{L} f_l(\mathbf{p}) b_{lm} g_m(\mathbf{q}) = f(\mathbf{p})^T B g(\mathbf{q}).$$

Substituting this into (1.0.1) we can see that

$$A \approx UBV = \tilde{A},$$

where U is an $N \times L$ matrix, B an $L \times L$ matrix, and V an $L \times N$ matrix with entries,

$$\begin{aligned} U_{il} &= \int_\Gamma f_l(\mathbf{p})\psi_i(\mathbf{p})\ d\Gamma_{\mathbf{p}} \\ B_{lm} &= b_{lm} \\ V_{mj} &= \int_\Gamma g_m(\mathbf{q})\psi_j(\mathbf{q})\ d\Gamma_{\mathbf{q}}. \end{aligned}$$

In place of A the elements of the sparse matrix decomposition UBV are computed. This requires $2NL + L^2$ elements, as opposed to N^2 for A. If $L = \mathcal{O}(\log N)$ we see that this requires only $\mathcal{O}(N \log N)$ elements are stored and a similar number of operations for forming $\tilde{A}\mathbf{x}$.

The second type of fast method, with which we are concerned with in this thesis, is the so-called *wavelet algorithm*, [6, 7, 8, 9]. Here, the basis functions ψ_i are the so-called *wavelet basis*. These are *refinable* bases obtained from translations and scalings of a single function ψ, the so-called *mother* wavelet. That is,

$$\psi_i = 2^{\frac{m}{2}}\psi(2^m \cdot -l) \quad \text{for } \ m \in \mathbb{Z},\ l \in \nabla_m.$$

They have the additional property of being orthogonal to low order polynomials; known as the property of vanishing moments.

We can show that using a wavelet basis, for a large class of kernels the elements of the Galerkin matrix A satisfy,

$$|A_{ij}| \leq c\frac{2^{-(m+m')(k+\frac{1}{2})-2k}}{(2k+1)\ dist(\Gamma_j, \Gamma_i)^{1+2k+\alpha}}.$$

We can prove that only $\mathcal{O}(N \log^p N)$ of these elements are sufficiently large enough to affect the accuracy of our solution. The rest of the elements need not be computed, resulting

in the desired efficiency.

In chapter 2 we review the methods and techniques required when partial differential equations are reformulated as boundary integral equations. We also discuss how these boundary integral equations may be discretised by a boundary element method. In chapter 3, we present the *multiresolution* framework for wavelets, along with our choice of basis functions for this thesis, namely, the *multiwavelets* of [10].

In chapter 4, we consider two boundary element methods, namely, the *standard* and *non-standard* Galerkin methods with multiwavelet basis functions. For both methods applied to operators of the standard analytical class, we find bounds for the size of the coefficient matrix elements. Using these bounds compression strategies are developed which only require the computation of the significant matrix elements. We show that there are $\mathcal{O}(N \log^p N)$ such significant elements, for some small integer value p.

In chapters 5 and 6 we apply the standard and non-standard Galerkin methods to several test problems. In chapter 5 we are concerned with the radiosity problem of image synthesis, whereas, in chapter 6 we are concerned with the boundary integral equation reformulation of Laplace's equation. However, when we consider Laplace's equation with Dirichlet boundary conditions the resulting coefficient matrix is ill-conditioned. Therefore, in order to use an iterative solver efficiently we must precondition the coefficient matrix. For a wavelet basis a diagonal scaling matrix is shown to be sufficient, see [11]. Here, we extend the preconditioner for use with multiwavelet basis functions. Finally we present a conclusion to our work and identify some future avenues of study.

Chapter 2

Boundary Integral Methods

In this chapter we introduce the methods and techniques required for solving boundary integral equations. In general boundary integral equations are derived as reformulations of partial differential equations over a domain Ω. We arrive at equations of the form

$$(\mathcal{A}u)(\mathbf{p}) = \left(\mathcal{B}\frac{\partial u}{\partial n}\right)(\mathbf{p}), \qquad \mathbf{p} \in \Gamma = \partial\Omega, \tag{2.0.1}$$

where $\mathcal{A}$ and $\mathcal{B}$ are pseudodifferential operators.

To discuss the existence and uniqueness of solutions to (2.0.1) and study the convergence analysis of boundary element methods, we need to introduce appropriate function spaces. Sobolev spaces are introduced in section 2.1. The operators $\mathcal{A}$ and $\mathcal{B}$ are pseudodifferential operators over Sobolev spaces. This allows us to study differential, integral and hypersingular operators within the same framework. Pseudodifferential operators are introduced in section 2.2. In section 2.3 we reformulate Laplace's equation as a boundary integral equation of the form (2.0.1), such equations are discretised using the projection methods introduced in section 2.4.

2.1 Sobolev Spaces

Sobolev spaces provide a natural setting in which to describe the smoothness of solutions in partial differential theory. In this section, we briefly introduce these spaces and their basic properties. For a more comprehensive study see [12].

Let Ω be a simply connected domain in $\mathbb{R}^n$. Initially the Sobolev spaces $\mathcal{W}_p^s(\Omega)$ are defined for non-negative integers s. For a multi-index of non-negative integers $\boldsymbol{l} = (l_1, \ldots, l_n)$, we define the partial derivative $\mathbf{D}^{\boldsymbol{l}}$ by

$$\mathbf{D}_x^{\boldsymbol{l}} = \mathbf{D}_1^{l_1}\mathbf{D}_2^{l_2}\ldots\mathbf{D}_n^{l_n} = \left(\frac{\partial^{l_1}}{\partial x_1^{l_1}}\right)\ldots\left(\frac{\partial^{l_n}}{\partial x_n^{l_n}}\right) = \frac{\partial^{|\boldsymbol{l}|}}{\partial x_1^{l_1}\ldots\partial x_n^{l_n}}, \tag{2.1.1}$$

where $|\boldsymbol{l}| = l_1 + \ldots + l_n$.

Definition 2.1.1. The space $\mathcal{W}_p^s(\Omega)$ is the space defined by

$$\mathcal{W}_p^s(\Omega) := \left\{u \in L_p(\Omega) |\ \mathbf{D}^{\boldsymbol{l}}u \in L_p(\Omega) \text{ for } |\boldsymbol{l}| \leq s\right\}, \tag{2.1.2}$$

and is equipped with the norm

$$\|u\|_{\mathcal{W}_p^s} = \left(\sum_{|\boldsymbol{l}|\leq s}\int_\Omega \left|\mathbf{D}^{\boldsymbol{l}}u(x)\right|^p dx\right)^{\frac{1}{p}}, \tag{2.1.3}$$

see [13].

Sobolev spaces with $p \neq 2$ are rarely used. Therefore, we concentrate on the case $p = 2$ and denote $\mathcal{W}_2^s(\Omega)$ by $H^s(\Omega)$. We note, that for $s = 0$, $H^0(\Omega) = L_2(\Omega)$. In order to introduce Sobolev spaces $H^s(\Omega)$ for real s, we consider the Fourier transform of a

function u,

$$\hat{u}(\xi) = \int_{\Omega} e^{-2\pi i x.\xi} u(x)\, dx. \tag{2.1.4}$$

Then, it can be shown, see [1], that for non-negative integers s,

$$c_1 \|u\|_{H^s}^2 \leq \int_{\Omega} \left(1 + |\xi|^2\right)^s |\hat{u}(\xi)|^2 \, d\xi \leq c_2 \|u\|_{H^s}^2 . \tag{2.1.5}$$

Therefore,

$$\left(\int_{\Omega} \left(1 + |\xi|^2\right)^s |\hat{u}(\xi)|^2 \, d\xi \right)^{\frac{1}{2}} \tag{2.1.6}$$

defines an equivalent norm in $H^s(\Omega)$. Furthermore, (2.1.6) has meaning for all real values of s. This allows us to define $H^s(\Omega)$ for any real s, possibly negative, by

$$H^s(\Omega) := \{u \in L_2(\Omega) |\ u \text{ is a generalized function such that (2.1.6) is finite}\} . \tag{2.1.7}$$

In fact, for $0 \leq s < \infty$ the space $H^{-s}(\Omega)$ is the dual of $H^s(\Omega)$, i.e. space of bounded linear functionals on $H^s(\Omega)$.

Let Γ be the boundary of a simply connected domain $\Omega \subset \mathbb{R}^n$. Then, we can similarly define Sobolev spaces $H^s(\Gamma)$, see [1]. For the case $n = 2$, if Γ has a smooth parameterisation

$$\gamma : [0, 1) \to \Gamma,$$

then, we may define $H^s(\Gamma)$ by

$$H^s(\Gamma) := \{u |\ (u \circ \gamma) \in H^s[0, 1)\} , \tag{2.1.8}$$

where $(u \circ g)(x) = u(\gamma(x))$. This definition is invariant under changes of the parameterisation, see [14].

Suppose $s, t \in \mathbb{R}$ with $s > t$. Then, $H^s \subset H^t$ and for $u \in H^s$ we have $\|u\|_{H^t} \leq \|u\|_{H^s}$. In fact the imbedding (identity) operator $I : H^s \to H^t$ is compact, see [1, Theorem 2.1.5]. We now mention an important trace theorem, [1, Theorem 2.2.2].

Theorem 2.1.1. *Let Ω be a bounded open domain with smooth boundary Γ. If $s > \frac{1}{2}$, then the trace operator*

$$u \to u|_{\Gamma} \tag{2.1.9}$$

is a continuous mapping from $H^s(\Omega)$ to $H^{s-\frac{1}{2}}(\Gamma)$.

2.2 Pseudodifferential Operators

Pseudodifferential operators are a natural extension of linear integral and partial differential operators. The theory of pseudodifferential operator has developed alongside the study of singular integral operators, which occur in many areas of mathematical physics. A pseudodifferential operator is a linear operator $\mathcal{A} : H^s(\Omega) \to H^{s-\alpha}(\Omega)$ where α is called the *order* of the operator. We can write the pseudodifferential operator $\mathcal{A}$ as the integral operator

$$(\mathcal{A}u)(\mathbf{p}) = \int_{\Omega} a(\mathbf{p}, \mathbf{q}) u(\mathbf{q})\, d\Omega_{\mathbf{q}}, \tag{2.2.1}$$

where $a(\cdot, \cdot)$ is a kernel function or a distribution. If a is a weakly singular kernel this is a classical compact integral operator. However, this definition also covers the cases of differential and integro-differential operators. We follow the approach of [1] to introduce the pseudodifferential operator concept.

A general partial differential operator of order α is a polynomial expression of the form

$$P(x, \mathbf{D}) = \sum_{|\boldsymbol{l}| \leq \alpha} a_{\boldsymbol{l}}(x) \mathbf{D}_x^{\boldsymbol{l}}, \tag{2.2.2}$$

where $\boldsymbol{l} = (l_1, \ldots, l_n)$ is a multi-integer and the symbol of the operator P is defined by

$$\sigma(P) = p(x, \xi) = \sum_{|\boldsymbol{l}| \leq \alpha} a_{\boldsymbol{l}}(x) \, (i\xi)^{\boldsymbol{l}} \, . \tag{2.2.3}$$

Therefore, we wish to show that Pu can be written in the integral form (2.2.1). We consider the inverse Fourier transform

$$u(x) = \int_{\Omega} e^{2\pi i x.\xi} \hat{u}(\xi) \, d\xi. \tag{2.2.4}$$

It follows that the partial derivatives satisfy

$$\mathbf{D}_x^{\boldsymbol{l}} u(x) = \int_{\Omega} e^{2\pi i x.\xi} (2\pi i \xi)^{\boldsymbol{l}} \hat{u}(\xi) \, d\xi, \tag{2.2.5}$$

and hence,

$$P(x, \mathbf{D})u(x) = \int_{\Omega} e^{2\pi i x.\xi} p(x, 2\pi\xi) \hat{u}(\xi) \, d\xi. \tag{2.2.6}$$

Therefore, substituting the Fourier transform

$$\hat{u}(\xi) = \int_{\Omega} e^{-2\pi i y.\xi} u(y) \, dy, \tag{2.2.7}$$

into (2.2.6) we obtain

$$P(x, \mathbf{D})u(x) = \int_{\Omega} k(x, y) u(y) \, dy, \tag{2.2.8}$$

where

$$k(x, y) = \int_{\Omega} p(x, 2\pi\xi) e^{2\pi i (x-y).\xi} \, d\xi. \tag{2.2.9}$$

Definition 2.2.1. $p(x,\xi)$ is said to be a symbol of order $\alpha \in \mathbb{R}$, denoted by $p \in \mathcal{S}^{\alpha}$, of a pseudodifferential operator $P(x,\mathbf{D})$ defined by (2.2.6), if it satisfies the following:

1. $p(x,\xi)$ is C^{∞} in both variables;
2. $p(x,\xi)$ has compact x-support;
3. for all multi-indices $\boldsymbol{l}$,$\boldsymbol{m}$, there is a constant $c_{\boldsymbol{l},\boldsymbol{m}}$ such that

$$\left|\mathbf{D}_x^{\boldsymbol{l}}\mathbf{D}_{\xi}^{\boldsymbol{m}}p(x,\xi)\right| \leq c_{\boldsymbol{l},\boldsymbol{m}}\left(1+|\xi|\right)^{\alpha-|\boldsymbol{m}|}. \tag{2.2.10}$$

Definition 2.2.2. If $p \in \mathcal{S}^{\alpha}$ the pseudodifferential operator P, with symbol p, is a pseudodifferential operator of order α.

We now give the basic mapping property of a pseudodifferential operator [1, Theorem 4.1.1].

Theorem 2.2.1. *Let P be a pseudodifferential operator of order $\alpha \in \mathbb{R}$. Then,*

$$P : H^{s} \to H^{s-\alpha} \tag{2.2.11}$$

for all $s \in \mathbb{R}$ and the mapping is continuous.

Therefore, if $\alpha < 0$ the operator acts as a smoothing or classical integral operator. However, if $\alpha > 0$ the pseudodifferential operator is principally a differential operator.

2.2.1 Solvability of Pseudodifferential Operator Equations

Let $\mathcal{A} : \mathcal{X} \to \mathcal{Y}$ be an operator from a normed space $\mathcal{X}$ to a normed space $\mathcal{Y}$. The equation

$$\mathcal{A}u = f \tag{2.2.12}$$

is said to be well-posed if the mapping is bijective and the inverse operator $\mathcal{A}^{-1} : \mathcal{Y} \to \mathcal{X}$ is continuous. Otherwise, the equation is said to be ill-posed; [14, §15].

For pseudodifferential operators on Sobolev spaces we know that the mappings are continuous, Theorem 2.2.1. However, this does not guarantee the existence of a bounded inverse. The additional property we require is that the pseudodifferential operators are Strongly Elliptic; [1, §4.3].

Definition 2.2.3. Let $p(x, \xi) \in \mathcal{S}^{\alpha}$. Then,

1. p is said to be Elliptic of order α if there exists $R > 0$ and $c > 0$ such that

$$|p(x, \xi)| \geq c(1 + |\xi|)^{\alpha} \quad \forall \ |\xi| \geq R. \tag{2.2.13}$$

2. p is said to be Strongly Elliptic of order α if there exists $R > 0$ and $c > 0$ such that

$$\mathfrak{Re}\ p(x, \xi) \geq c(1 + |\xi|)^{\alpha} \quad \forall \ |\xi| \geq R. \tag{2.2.14}$$

The pseudodifferential operator P is said to be (strongly) elliptic if its symbol p is (strongly) elliptic.

We can now state the basic result which links all our boundary integral operators on $\Gamma \in C^{\infty}$; See [15].

Theorem 2.2.2. *The boundary integral operators associated with regular elliptic boundary value problems are strongly elliptic pseudodifferential operators of integer order.*

We next quote the important coerciveness result which is used to prove the solvability of the pseudodifferential operator equation.

Theorem 2.2.3. *(Gårding Inequality, [16, §0.7][17, Theorem 3.9]). If $\mathcal{A}$ is a strongly elliptic pseudodifferential operator of order α then there exists a positive constant γ and a compact operator $C : H^{\frac{\alpha}{2}}(\Gamma) \to H^{\frac{\alpha}{2}}(\Gamma)$ such that for all $g \in H^{\frac{\alpha}{2}}(\Gamma)$*

$$\mathfrak{Re}\ \langle (\mathcal{A}+\mathcal{C})g, g\rangle_{L_2(\Gamma)} \geq \gamma \left\|g\right\|^2_{H^{\frac{\alpha}{2}}(\Gamma)}. \tag{2.2.15}$$

Hence, if $\mathcal{D} = \mathcal{A} + \mathcal{C}$ then, the above result says that $\mathcal{D}$ is strictly coercive.

Theorem 2.2.4. *(Lax-Milgram,[14, Theorem 13.23]). In a Hilbert space $\mathcal{X}$, a strictly coercive operator $\mathcal{D} : \mathcal{X} \to \mathcal{Y}$ has a bounded (continuous) inverse.*

This says that for a strongly elliptic pseudodifferential operator $\mathcal{A}$ we can write $\mathcal{A} = \mathcal{D} - \mathcal{C}$, where $\mathcal{C}$ is compact and $\mathcal{D}$ has a bounded inverse. Thus, for strongly elliptic $\mathcal{A}$ we can write (2.2.12) in the equivalent form of a second kind equation

$$\left(\mathcal{I} - \mathcal{D}^{-1}\mathcal{C}\right) u = \mathcal{D}^{-1} f, \tag{2.2.16}$$

where $\mathcal{D}^{-1}\mathcal{C}$ is compact. This means that for strongly elliptic pseudodifferential operators, including first kind and hypersingular equations, the existence of unique solutions can be established from the Fredholm alternative, see [14, 18].

2.3 Boundary Integral Equations

Let Γ be a closed surface in $\mathbb{R}^3$ or a closed contour in $\mathbb{R}^2$ containing a number of subsurfaces of class C^2. We denote the interior and exterior of Γ by Ω_- and Ω_+, respectively. The equation

$$\nabla^2 u(\mathbf{p}) = 0, \qquad \mathbf{p} \in \Omega_\pm, \tag{2.3.1}$$

is called Laplace's equation. Here, we are interested in deriving the boundary integral equation solution of (2.3.1) with appropriate boundary conditions. We will use these boundary integral equations more fully in chapter 6, where we study their numerical solution by multiwavelets.

2.3.1 Free Space Green's Function

The function

$$G(\mathbf{p}, \mathbf{q}) = \begin{cases} -\frac{1}{2\pi} \ln r, & \text{in 2 dimensions,} \\ \frac{1}{4\pi r}, & \text{in 3 dimensions,} \end{cases} \tag{2.3.2}$$

where $r = |\mathbf{p} - \mathbf{q}|$, is called the free space Green's function or the fundamental solution for Laplace's equation, since G satisfies

$$\nabla^2 G(\mathbf{p}, \mathbf{q}) = -\delta(\mathbf{p} - \mathbf{q}), \tag{2.3.3}$$

both as a function of $\mathbf{p}$ and $\mathbf{q}$. The function δ is the Dirac delta function.

2.3.2 Boundary Integral Operators

We now define the boundary integral operators for Laplace's equation, namely the single- and double-layer potentials and their normal derivatives. We also study some of their pertinent smoothness properties.

Definition 2.3.1. Let the density function $\sigma \in C(\Gamma)$, we define the following operators: The single-layer potential,

$$(\mathcal{L}\sigma)(\mathbf{p}) = \int_\Gamma \sigma(\mathbf{q}) G(\mathbf{p},\mathbf{q}) d\Gamma_{\mathbf{q}}; \tag{2.3.4}$$

The double-layer potential,

$$(\mathcal{M}\sigma)(\mathbf{p}) = \int_\Gamma \sigma(\mathbf{p}) \frac{\partial G(\mathbf{p},\mathbf{q})}{\partial n_{\mathbf{q}}} d\Gamma_{\mathbf{q}}; \tag{2.3.5}$$

The normal derivative of the single-layer potential,

$$\left(\mathcal{M}^{\mathrm{T}}\sigma\right)(\mathbf{p}) = \frac{\partial}{\partial n_{\mathbf{p}}}(\mathcal{L}\sigma)(\mathbf{p}) = \frac{\partial}{\partial n_{\mathbf{p}}} \int_\Gamma \sigma(\mathbf{q}) G(\mathbf{p},\mathbf{q}) d\Gamma_{\mathbf{q}}; \tag{2.3.6}$$

The normal derivative of the double-layer potential (the hypersingular operator),

$$(\mathcal{N}\sigma)(\mathbf{p}) = \frac{\partial}{\partial n_{\mathbf{p}}}(\mathcal{M}\sigma)(\mathbf{p}) = \frac{\partial}{\partial n_{\mathbf{p}}} \int_\Gamma \sigma(\mathbf{q}) \frac{\partial G(\mathbf{p},\mathbf{q})}{\partial n_{\mathbf{q}}} d\Gamma_{\mathbf{q}}. \tag{2.3.7}$$

Where by $n_{\mathbf{p}}$ and $n_{\mathbf{q}}$ we denote the unit outward normal to Γ at $\mathbf{p}$ or at $\mathbf{q}$, respectively. We note that, the operator $\mathcal{M}^{\mathrm{T}}$ is the normal derivative of $\mathcal{L}$ and is the operator transpose of $\mathcal{M}$.

The Laplace boundary integral operators are strongly elliptic pseudodifferential operators. The single-layer operator is of order -1. Therefore, it is a smoothing operator

from $H^s(\Gamma) \rightarrow H^{s+1}(\Gamma)$. The hypersingular operator $\mathcal{N}$ has order $+1$. Therefore, it acts like a differential operator, that is, $\mathcal{N} : H^s(\Gamma) \rightarrow H^{s-1}(\Gamma)$. The operators $\mathcal{M}$ and $\mathcal{M}^{\mathrm{T}}$ are infinitely smooth on C^∞ boundaries, that is, $\mathcal{M} : H^s(\Gamma) \rightarrow C^\infty(\Gamma)$ and $\mathcal{M}^{\mathrm{T}} : H^s \rightarrow C^\infty(\Gamma)$. However, this phenomenon is special to the 2 dimensional case. In the 3 dimensional case, $\mathcal{M}$ and $\mathcal{M}^{\mathrm{T}}$ have order -1 and hence, $\mathcal{M} : H^s(\Gamma) \rightarrow H^{s+1}(\Gamma)$ and $\mathcal{M}^{\mathrm{T}} : H^s(\Gamma) \rightarrow H^{s+1}(\Gamma)$.

Theorem 2.3.1. *Let $\Omega \subset \mathbb{R}^3$ (or $\Omega \subset \mathbb{R}^2$) be a bounded domain with a smooth boundary Γ. Also, we let $\sigma \in H^s(\Gamma)$, $s \geq 0$. We denote points in the domain Ω_- by $\boldsymbol{p}_-$, points in Ω_+ by $\boldsymbol{p}_+$ and points on the boundary Γ by $\boldsymbol{p}$. We define*

$$\left(\mathcal{L}^+\sigma\right)(\boldsymbol{p}) = \lim_{\boldsymbol{p}_+ \rightarrow \boldsymbol{p}} \left(\mathcal{L}\sigma\right)\left(\boldsymbol{p}_+\right), \tag{2.3.8}$$

$$\left(\mathcal{L}^-\sigma\right)(\boldsymbol{p}) = \lim_{\boldsymbol{p}_- \rightarrow \boldsymbol{p}} \left(\mathcal{L}\sigma\right)\left(\boldsymbol{p}_-\right) \tag{2.3.9}$$

and similarly define $\mathcal{M}^+$, $\mathcal{M}^-$, $\mathcal{M}^{T+}$, $\mathcal{M}^{T-}$, $\mathcal{N}^+$ and $\mathcal{N}^-$. Then, for $\boldsymbol{p} \in \Gamma$ we have;

$$\left(\mathcal{L}^+\sigma\right)(\boldsymbol{p}) = \left(\mathcal{L}^-\sigma\right)(\boldsymbol{p}) = \left(\mathcal{L}\sigma\right)(\boldsymbol{p}), \tag{2.3.10}$$

$$\left(\mathcal{M}^+\sigma\right)(\boldsymbol{p}) = \frac{1}{2}\sigma(\boldsymbol{p}) + \left(\mathcal{M}\sigma\right)(\boldsymbol{p}), \tag{2.3.11}$$

$$\left(\mathcal{M}^-\sigma\right)(\boldsymbol{p}) = -\frac{1}{2}\sigma(\boldsymbol{p}) + \left(\mathcal{M}\sigma\right)(\boldsymbol{p}), \tag{2.3.12}$$

$$\left(\mathcal{M}^{T+}\sigma\right)(\boldsymbol{p}) = -\frac{1}{2}\sigma(\boldsymbol{p}) + \left(\mathcal{M}^T\sigma\right)(\boldsymbol{p}), \tag{2.3.13}$$

$$\left(\mathcal{M}^{T-}\sigma\right)(\boldsymbol{p}) = \frac{1}{2}\sigma(\boldsymbol{p}) + \left(\mathcal{M}^T\sigma\right)(\boldsymbol{p}), \tag{2.3.14}$$

$$\left(\mathcal{N}^+\sigma\right)(\boldsymbol{p}) = \left(\mathcal{N}^-\sigma\right)(\boldsymbol{p}) = \left(\mathcal{N}\sigma\right)(\boldsymbol{p}). \tag{2.3.15}$$

Proof: See [1]. □

Therefore the operators $\mathcal{L}$ and $\mathcal{N}$ are continuous. However, the operators $\mathcal{M}$ and $\mathcal{M}^{\mathrm{T}}$ have a jump discontinuity at $\mathbf{p} \in \Gamma$.

2.3.3 Direct Formulation of the Boundary Integral Equation

The direct formulation makes use of Green's second Theorem.

Theorem 2.3.2. *(Second Green's Theorem). Let* $u, v \in C^2(\Omega)$*. Then,*

$$\int_\Omega \left(u\nabla^2 v - v\nabla^2 u\right) d\Omega = \int_\Gamma \left(u\frac{\partial v}{\partial n} - v\frac{\partial u}{\partial n}\right) d\Gamma. \tag{2.3.16}$$

Consider Laplace's equation in the exterior domain,

$$\begin{aligned} \nabla^2 u(\mathbf{p}) &= 0, \qquad \mathbf{p} \in \Omega_+ \\ \lim_{|\mathbf{p}|\to\infty} |u(\mathbf{p})| &= 0. \end{aligned} \tag{2.3.17}$$

In (2.3.16) if we take u to be the solution of Laplace's equation and v the free space Green's function satisfying (2.3.3), we obtain the Laplace integral equation representation,

$$u(\mathbf{p}) = \int_\Gamma u(\mathbf{p})\frac{\partial G(\mathbf{p},\mathbf{q})}{\partial n_\mathbf{q}} d\Gamma_\mathbf{q} - \int_\Gamma G(\mathbf{p},\mathbf{q})\frac{\partial u(\mathbf{q})}{\partial n_\mathbf{q}} d\Gamma_\mathbf{q}, \quad \mathbf{p} \in \Omega_+. \tag{2.3.18}$$

Then, by letting $\mathbf{p} \in \Omega_+ \to \mathbf{p} \in \Gamma$ and using the jump conditions of Theorem 2.3.1, we obtain

$$\frac{1}{2}u(\mathbf{p}) = \int_\Gamma u(\mathbf{q})\frac{\partial G(\mathbf{p},\mathbf{q})}{\partial n_\mathbf{q}} d\Gamma_\mathbf{q} - \int_\Gamma G(\mathbf{p},\mathbf{q})\frac{\partial u(\mathbf{q})}{\partial n_\mathbf{q}} d\Gamma_\mathbf{q}, \quad \mathbf{p} \in \Gamma. \tag{2.3.19}$$

Rewriting (2.3.19) in terms of the single- and double-layer operators, $\mathcal{L}$ and $\mathcal{M}$ respectively, we have

$$\left(-\frac{1}{2}\mathcal{I} + \mathcal{M}\right) u(\mathbf{p}) = \mathcal{L}\frac{\partial u}{\partial n}(\mathbf{p}), \qquad \mathbf{p} \in \Gamma. \tag{2.3.20}$$

Clearly if we have both u and $\frac{\partial u}{\partial n}$ on Γ (the so-called "Cauchy data"), (2.3.18) gives the

unique solution to Laplace's equation. In practice we have either u or $\frac{\partial u}{\partial n}$ on Γ (or part of Γ) and we solve (2.3.20) for the missing Cauchy data. Then, (2.3.18) is used to obtain $u(\mathbf{p})$ for $\mathbf{p} \in \Omega_+$.

Indeed it is the simple boundary integral equation (2.3.20) which we solve in chapter 6, both in the case of Dirichlet and Neumann boundary conditions, using multiwavelets.

2.4 Projection Methods

In this section we consider the numerical solution of pseudodifferential equations of the form

$$\mathcal{A}u = f, \tag{2.4.1}$$

where we assume $\mathcal{A} : H^s(\Gamma) \rightarrow H^{s-\alpha}(\Gamma)$ is any of the boundary integral operators introduced in section 2.3.2. The main idea of projection methods is to seek an approximate solution from some finite dimensional subspace of the space containing the exact solution. We then try to force the approximate solution to have small residual when the integral equation is projected onto this space. We consider the Galerkin method which is an orthogonal projection method and the collocation method in which the projection is interpolatory. For a more comprehensive study see [19].

First we define a projection operator and its corresponding projection method [14].

Definition 2.4.1. Let $\mathcal{X}$ be a Banach space and $\mathcal{Y}$ a non trivial subspace of $\mathcal{X}$. A bounded linear operator $P : \mathcal{X} \rightarrow \mathcal{Y}$ with the property that $Py = y$ for all $y \in \mathcal{Y}$, is called a *projection operator* from $\mathcal{X} \rightarrow \mathcal{Y}$.

Theorem 2.4.1. *A non trivial bounded linear operator is a projection operator if and only if it satisfies $P^2 = P$. Furthermore, $\|P\| \geq 1$.*

Definition 2.4.2. Let $\mathcal{A} : H^s(\Gamma) \to H^{s-\alpha}(\Gamma)$ be an injective bounded linear operator. Let $H_N \subset H^s(\Gamma)$ and $H'_N \subset H^{s-\alpha}(\Gamma)$ be two sequences of subspaces with $\dim H_N = \dim H'_N = N$ and let $P_N : H^{s-\alpha}(\Gamma) \to H'_N$ be projection operators. The projection method generated by H_N and P_N approximates equation (2.4.1) by the projection equation

$$P_N \mathcal{A} u_N = P_N f, \quad u_N \in H_N. \tag{2.4.2}$$

The projection method is said to be convergent if there exists some $C \in \mathbb{N}$ such that for each $f \in H^{s-\alpha}(\Gamma)$, the approximating equation $P_N \mathcal{A} u_N = P_N f$ has a unique solution $u_N \in H_N$ for all $N \geq C$ and $u_N \to u$ as $N \to \infty$.

We now discuss the collocation and Galerkin methods.

2.4.1 Collocation Method

We start by recalling a result regarding interpolation and interpolation operators [14, §13].

Theorem 2.4.2. *Let $H_N \subset H^s(\Gamma)$ be an N-dimensional subspace and $x_1, \ldots, x_N$ be N points in Γ such that H_N is unisolvent with respect to $x_1, \ldots, x_N$. That is, each function from H_N which vanishes at these points must be identically zero. Then, given values $f_1, \ldots, f_N$ there exists a unique function $v \in H_N$ such that*

$$v(x_i) = f_i, \quad i = 1, \ldots, N.$$

With the data given by the values $f_i = f(x_i)$, $i = 1, \ldots, N$, of a function $f \in H^s(\Gamma)$ the mapping $f \mapsto v$ defines a bounded linear projection operator $\mathcal{P}_N : H^s(\Gamma) \to H_N$ called the interpolation operator.

Given equation (2.4.1), $\mathcal{A}u = f$, where $\mathcal{A} : H^s(\Gamma) \to H^{s-\alpha}(\Gamma)$ is a strongly elliptic pseudodifferential operator of order α, the collocation method seeks an approximate solution, in the subspace $H_N \subset H^s(\Gamma)$, by requiring that the equation is satisfied at a finite number of collocation points. Choosing N points $\{x_i\}$, the collocation method approximates the solution of (2.4.1) by a function $u_N \in H_N$ such that

$$(\mathcal{A}u_N)(x_i) = f(x_i), \quad i = 1, \ldots, N. \tag{2.4.3}$$

Let us assume that H_N is the space of piecewise polynomials (splines) of degree d, with basis functions $\{\chi_i\}$. Then, the approximate solution has the form

$$u_N(x) = \sum_{j=1}^{N} \beta_j \chi_j(x). \tag{2.4.4}$$

Substituting (2.4.4) into (2.4.3) yields the system of linear equations

$$\sum_{j=1}^{N} \beta_j (\mathcal{A}\chi_j)(s_i) = f(x_i), \quad i = 1, \ldots, N \tag{2.4.5}$$

for the unknown coefficients $\{\beta_j\}$. This can be interpreted as a projection method with $\mathcal{P}_N$ being the interpolation operator in Theorem 2.4.2.

The following convergence result holds for the collocation method, see [15, Theorem 3.6],[20].

Theorem 2.4.3. *Let $\alpha < d$ if d is odd and $\alpha < d + \frac{1}{2}$ if d is even. Let $\alpha \le t \le s \le d + 1$, $t < d + \frac{1}{2}$ and $\alpha + \frac{1}{2} < s$. Then, there exist constants c and c' such that*

$$\|u - u_N\|_{H^t} \le c \|u - \mathcal{P}_N u\|_{H^t} \tag{2.4.6}$$

$$\le c' h^{s-t} \|u\|_{H^s}. \tag{2.4.7}$$

Remark: The first inequality shows that the error in collocation is of the same order as the error in approximation by the interpolation projector. The second part simply states the error for projection into spline spaces.

2.4.2 Galerkin Method

We consider the Galerkin solution of (2.4.1), $\mathcal{A}u = f$, where $\mathcal{A} : H^s(\Gamma) \to H^{s-\alpha}(\Gamma)$ is a strongly elliptic pseudodifferential operator of order α. The Galerkin method, as in the collocation method, seeks an approximate solution $u_N \in H_N \subset H^s(\Gamma)$, where $\dim H_N = N$. The Galerkin method approximates the solution of (2.4.1) by a function $u_N \in H_N$ such that

$$\langle \mathcal{A}u_N, v_N \rangle_{L_2} = \langle f, v_N \rangle_{L_2}, \tag{2.4.8}$$

holds for all $v_N \in H_N$. Or equivalently $\langle \mathcal{A}\left(u - u_N\right), v_N \rangle_{L_2} = 0$, showing that this is an orthogonal projection method. Let us assume that H_N is the space of piecewise polynomials (splines) of degree d, with basis functions $\{\chi_i\}$. Then, we can express u_N in the form

$$u_N(x) = \sum_{i=1}^{N} \beta_i \chi_i(x). \tag{2.4.9}$$

Then, the Galerkin equation (2.4.8) is equivalent to the linear system

$$\sum_{j=1}^{N} \beta_j \langle \mathcal{A}\chi_j, \chi_i \rangle_{L_2} = \langle f, \chi_i \rangle_{L_2} \quad i = 1, \ldots, N, \tag{2.4.10}$$

for the unknown coefficients $\{\beta_j\}$.

The following convergence result holds for the Galerkin method, see [15, Theorem 2.10], [1, Cor. 10.1.2].

Theorem 2.4.4. *Let $\alpha < 2d+1$. Let $\alpha - d - 1 \leq t \leq s \leq d+1$ and $t < d + \frac{1}{2}$. Then, there exist constants c and c' such that*

$$\|u - u_N\|_{H^t} \leq c \|u - \mathcal{P}_N u\|_{H^t} \tag{2.4.11}$$

$$\leq c' h^{s-t} \|u\|_{H^s} . \tag{2.4.12}$$

Remark: The first inequality shows that the error in the Galerkin method is of the same order as the error in approximation by the orthogonal projector. The second part simply states the error for projection into spline spaces. We note that for the Galerkin method, the range of t is different, allowing more accuracy if negative norms are used.

Chapter Review

In this chapter we have introduced the methods and techniques required for solving boundary integral equations. In order to be able to discuss the existence and uniqueness of solutions to $(\mathcal{A}u)(\mathbf{p}) = \left(\mathcal{B}\frac{\partial u}{\partial n}\right)(\mathbf{p})$, the Sobolev spaces were introduced in section 2.1. We briefly discussed the theory of pseudodifferential operators in section 2.2. Within the framework of pseudodifferential operators, we can then study differential, integral and hypersingular operators.

In section 2.3 we introduced the single- and double-layer potentials $\mathcal{L}$ and $\mathcal{M}$, respectively. Then, using Green's second Theorem we reformulated Laplace's equation as a boundary integral equation. Such boundary integral equations are discretised using the projection methods introduced in section 2.4. In particular, we discussed the collocation

and Galerkin methods, and their respective convergence properties. It is the Galerkin method which we employ mainly, in this thesis.

Chapter 3

Wavelet Analysis

Wavelets were developed independently by mathematicians, physicists and engineers who came together in the 1980's to develop the subject of wavelets. In simplest terms wavelets are just a basis for a Hilbert space H, that have several interesting and important features that make them different to other basis functions. This has lead to wavelets being widely used in applications ranging from data compression to data denoising in multimedia to the fast solution of problems in numerical analysis, see [11, 21, 22, 23, 24, 25].

The wide applicability of wavelets is due to the fact that wavelets can very efficiently and effectively approximate a large class of functions. They provide efficient approximations to functions at edges and corners, due to their *multiresolution* property. The multiresolution property acts like a 'mathematical microscope' letting us zoom in on the finer detail of functions and then zoom out again to see the coarser detail. They also have the property of *vanishing moments*, this leads to the wavelet coefficients being small when the function is smooth over the support of the wavelet and consequently leads to the compression of data.

Before continuing let us introduce a compact notation for general bases and their transforms. Let $\mathbf{\Phi}$ denote a countable set or collection of functions in the Hilbert space H. Here, we write a linear combination of elements of $\mathbf{\Phi}$ in the form

$$\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{\Phi} = \sum_{\phi\in\mathbf{\Phi}} \alpha_\phi \phi, \tag{3.0.1}$$

where α_ϕ are some real or complex valued coefficients. Furthermore, for any $f \in H$, the quantities $\langle f, \mathbf{\Phi}\rangle$ and $\langle \mathbf{\Phi}, f\rangle$ mean the row and column vectors, respectively, of the coefficients $\langle f, \phi\rangle$ and $\langle \phi, f\rangle$, $\phi \in \mathbf{\Phi}$. Now, we consider two countable collections of functions $\mathbf{\Phi}$ and $\mathbf{\Upsilon}$. Then, the possibly infinite dimensional matrix $(\langle \phi, \upsilon\rangle)_{\phi\in\mathbf{\Phi},\upsilon\in\mathbf{\Upsilon}}$ can be represented in shorthand by $\langle \mathbf{\Phi}, \mathbf{\Upsilon}\rangle$.

3.1 Multiresolution

The multiresolution property plays an important role in the context of wavelets on $\mathbb{R}$. Let H be a Hilbert space with inner product $\langle\cdot,\cdot\rangle$ and norm $\|\cdot\| = \|\cdot\|_H = \langle\cdot,\cdot\rangle^{\frac{1}{2}}$. Let us consider a *refinable* countable set $\mathbf{\Phi}_m \subset H$, for $m \in \mathbb{Z}$. That is, $\mathbf{\Phi}_m$ is obtained by translations and scalings of a single function ϕ. For any countable set $\mathbf{\Phi}_m \subset H$, let

$$V_m := \overline{\operatorname{span} \mathbf{\Phi}_m}, \quad \mathbf{\Phi}_m = \{\phi_\lambda := 2^{\frac{m}{2}}\phi(2^m\cdot - l)|\ \ \lambda = \{m, l\},\ l \in \Delta_m\}, \tag{3.1.1}$$

for some, possibly infinite, countable index set Δ_m. We will give an example later in this section. Note that here, the parameter λ is a couplet $\lambda = \{m, l\}$ identifying the level, i.e. $m \in \mathbb{Z}$ and the location, $l \in \Delta_m$.

Definition 3.1.1. Any countable set $\mathbf{\Phi}_m \subset H$ is called a Riesz basis of H, if there exist positive constants a and b, with $0 < a \leq b < \infty$, such that

$$a \left\|\boldsymbol{\alpha}\right\|_{\ell_2(\Delta_m)} \leq \left\|\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{\Phi}_m\right\|_H \leq b \left\|\boldsymbol{\alpha}\right\|_{\ell_2(\Delta_m)}. \tag{3.1.2}$$

In shorthand we denote this as

$$\left\|\boldsymbol{\alpha}\right\|_{\ell_2} \sim \left\|\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{\Phi}_m\right\|_H. \tag{3.1.3}$$

In the above relations,

$$\left\|\boldsymbol{\alpha}\right\|_{\ell_2} = \sqrt{\sum_{l\in\Delta_m} |\alpha_l|^2}.$$

Then, a sequence $\mathcal{V} = \{V_m\}_{m\in\mathbb{Z}}$ of closed subspaces $V_m \subset H$ is said to form a multiresolution analysis of H, if it satisfies the following conditions, [26]:

1. $\ldots \subset V_{-1} \subset V_0 \subset V_1 \subset \ldots \subset H$;

2. $\overline{\left(\bigcup_{m\in\mathbb{N}} V_m\right)} = H$;

3. $\bigcap_{m\in\mathbb{N}} V_m = \{0\}$;

4. $f(x) \in V_m \iff f(2^{-m}x) \in V_0$;

5. The basis $\mathbf{\Phi}_m$ is a Riesz basis.

Definition 3.1.2. If $\mathbf{\Phi}_m$ is a Riesz basis for the space V_m, and the spaces V_m satisfy the multiresolution conditions 1, 2 and 4, then, the function ϕ is called the *scaling* function.

3.1.1 Multiscale Basis

The sequence of nested subspaces $\mathcal{V}$ is dense in H. Therefore, we can assemble a basis for H from the functions that span the differences between trial spaces. Define W_m to be the complement of the trial space V_m in V_{m+1}, that is

$$V_{m+1} = V_m \dot{+} \ W_m. \tag{3.1.4}$$

Hence, we look for countable sets

$$\mathbf{\Psi}_m = \{\psi_\lambda := 2^{\frac{m}{2}}\psi(2^m \cdot -l)| \ \ \lambda = \{m, l\},\ l \in \nabla_m\} \subset V_{m+1}, \tag{3.1.5}$$

such that

$$W_m = \overline{\text{span}\ \mathbf{\Psi}_m}. \tag{3.1.6}$$

The set $\{\mathbf{\Phi}_m \cup \mathbf{\Psi}_m\}$ satisfies (3.1.3) and therefore is a Riesz basis for V_{m+1}. If there also exists a space

$$\widetilde{W}_m := \overline{\text{span}\ \widetilde{\mathbf{\Psi}}_m}, \tag{3.1.7}$$

such that

$$\langle \mathbf{\Psi}_m, \widetilde{\mathbf{\Psi}}_{m'} \rangle = \text{I}, \tag{3.1.8}$$

then, $\mathbf{\Psi}_m$ is a wavelet series and the function ψ in (3.1.5) is called the *mother* wavelet. Furthermore, $\widetilde{\mathbf{\Psi}}_m = \{\tilde{\psi}_\lambda := 2^{\frac{m}{2}}\tilde{\psi}(2^m \cdot -l)|\ \lambda = \{m, l\},\ l \in \nabla_m\}$ is also a wavelet series and the function $\tilde{\psi}$ is called the *dual* wavelet. Similarly, there is a dual basis $\tilde{\mathbf{\Phi}}_m = \{\tilde{\phi}_\lambda := 2^{\frac{m}{2}}\tilde{\phi}(2^m \cdot -l)|\ \lambda = \{m, l\},\ l \in \Delta_m\}$ that generates a sequence $\widetilde{\mathcal{V}} = \left\{\widetilde{V}_m\right\}_{m\in\mathbb{Z}}$ of closed subspaces, which form a different multiresolution of H. Note that we use the countable index set ∇_m for the location of wavelet functions, whereas, we use the countable index set Δ_m for the location of scaling functions. We see later that for the orthogonal bases

on which we concentrate, $\nabla_m = \Delta_m$. The wavelet series $\mathbf{\Psi}_m$ and $\widetilde{\mathbf{\Psi}}_m$ are the so-called *biorthogonal* wavelet bases; see [27, 28]. In this thesis we concentrate on orthogonal wavelet bases. If $\mathbf{\Psi}_m$ is an orthogonal wavelet basis, then, the wavelet ψ is said to be *self-dual*. That is,

$$\psi = \tilde{\psi}.$$

In this case we have $V_{m+1} = V_m \oplus W_m$.

Throughout this thesis we denote the highest level of discretisation by M and N_M will denote the dimension of the space V_M. Then, through recursive use of decomposition (3.1.4), we can write the trial space V_M as the sum of complement spaces

$$V_M = V_{m_0} \bigoplus_{m=m_0}^{M-1} W_m, \tag{3.1.9}$$

where m_0 is some fixed coarsest level. This relationship is shown in Figure 3.1. Thus, any $f_M \in V_M$ can be expressed in single scale form, that is, with respect to the basis $\mathbf{\Phi}_M$, as

$$f_M = \mathbf{\Phi}_M^{\mathrm{T}} \boldsymbol{\alpha}_M, \tag{3.1.10}$$

where $\boldsymbol{\alpha}_M = \langle \mathbf{\Phi}_M, f \rangle$. We can also express the function in multiscale form, that is, with respect to the basis

$$\mathbf{\Psi}^M := \mathbf{\Phi}_{m_0} \bigcup_{m=m_0}^{M-1} \mathbf{\Psi}_m, \tag{3.1.11}$$

as

$$f_M = \mathbf{\Phi}_{m_0}^{\mathrm{T}} \boldsymbol{\alpha}_{m_0} + \mathbf{\Psi}_{m_0}^{\mathrm{T}} \boldsymbol{\beta}_{m_0} + \ldots + \mathbf{\Psi}_{M-1}^{\mathrm{T}} \boldsymbol{\beta}_{M-1}, \tag{3.1.12}$$

where $\boldsymbol{\beta}_m = \langle \mathbf{\Psi}_m, f \rangle$ for $m = m_0, \ldots, M-1$. Since the sequence $\mathcal{V}$ is dense in H, the union

$$\mathbf{\Psi} := \mathbf{\Phi}_{m_0} \bigcup_{m=m_0}^{\infty} \mathbf{\Psi}_m \tag{3.1.13}$$

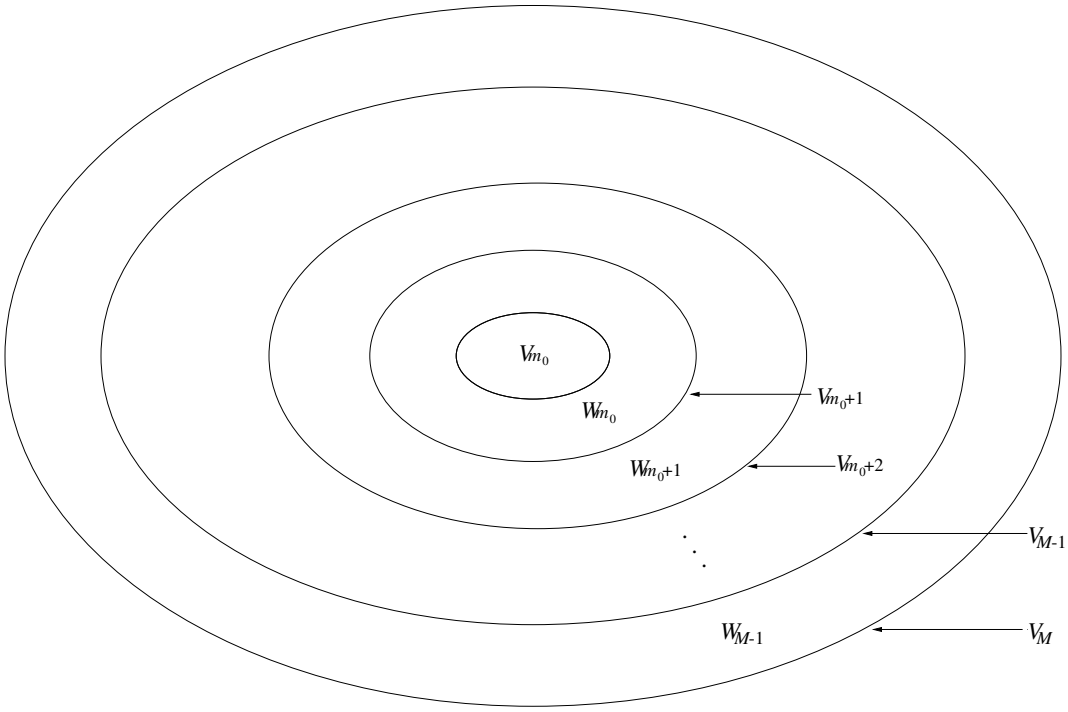


Figure 3.1: Decomposition of V_M into the complement spaces W_m

is a candidate for a basis for the whole space H.

3.1.2 Vanishing Moments

An important property of wavelets is that of vanishing moments. A wavelet ψ has vanishing moments of order d if

$$\int_{\mathbb{R}} x^i \psi(x)\, dx = 0 \quad \text{for } i = 0, \ldots, d-1. \tag{3.1.14}$$

It is the order of vanishing moments that governs the compression capacity of a wavelet. Thus, for numerical applications we wish to have a high order of vanishing moments to maximize operator compression.

Example 3.1.1. (Haar Basis).

The simplest example of an orthogonal wavelet, is the Haar basis or Haar wavelets. Here, $H = L_2[0,1]$ *and* V_m *is the space of piecewise constant functions with nodes at* $x_i = \frac{i}{2^m}$*, for* $i = 0, 1, \ldots, 2^m - 1$*. Our scaling function is*

$$\phi(x) = \begin{cases} 1 & \text{for } 0 \leq x < 1; \\ 0 & \text{elsewhere}. \end{cases} \tag{3.1.15}$$

Therefore, the countable set $\mathbf{\Phi}_m = \{\phi_\lambda := 2^{\frac{m}{2}}\phi(2^m \cdot -l) | \ \ \lambda = \{m, l\},\ l \in \Delta_m\}$ *where* $\Delta_m = \{0, 1, \ldots, 2^m - 1\}$ *is a basis for the space* V_m*.*

For the coarsest level $m = m_0 = 0$*,* $\mathbf{\Phi}_0 = \{\phi\}$ *is a basis for the space* V_0*. The space* V_1 *is the space of piecewise constant functions with nodes at* 0 *and* $\frac{1}{2}$*. Therefore,* $\mathbf{\Phi}_1 = \{\phi_{1,0}, \phi_{1,1}\}$ *is a basis for the space* V_1*, where*

$$\phi_{1,0}(x) = \begin{cases} \sqrt{2}\phi_{0,0}(2x) & \text{for } 0 \leq x < \frac{1}{2}; \\ 0 & \text{otherwise}, \end{cases} \tag{3.1.16}$$

$$\phi_{1,1}(x) = \begin{cases} \sqrt{2}\phi_{0,0}(2x-1) & \text{for } \frac{1}{2} \leq x < 1; \\ 0 & \text{otherwise.} \end{cases} \tag{3.1.17}$$

The Haar function or Haar wavelet is

$$\psi(x) = \begin{cases} 1 & \text{for } 0 \leq x < \frac{1}{2}; \\ -1 & \text{for } \frac{1}{2} \leq x < 1; \\ 0 & \text{elsewhere.} \end{cases} \tag{3.1.18}$$

Clearly, the Haar wavelet has vanishing moments of order 1. The countable set $\mathbf{\Psi}_m =$

$\{\psi_\lambda := 2^{\frac{m}{2}}\psi(2^m \cdot -l)|\ \ \lambda = \{m, l\},\ l \in \nabla_m\}$*, where* $\nabla_m = \{0, 1, \ldots, 2^m - 1\}$*, is a basis for the space* W_m*.*

Since, $V_1 = W_0 \oplus V_0$*, we have two distinct bases for the space* V_1*, namely,* $\mathbf{\Phi}_1$ *and* $\{\mathbf{\Psi}_0 \cup \mathbf{\Phi}_0\}$*. Therefore, a function in the space* V_1 *can be represented as a linear combination of either the basis* $\mathbf{\Phi}_1$ *or the basis* $\{\mathbf{\Phi}_0 \cup \mathbf{\Psi}_0\}$*. For the Haar basis it is easy to verify this,*

$$\phi_{1,0}(x) = \frac{1}{\sqrt{2}}\phi_{0,0}(x) + \frac{1}{\sqrt{2}}\psi_{0,0}(x), \tag{3.1.19}$$

$$\phi_{1,1}(x) = \frac{1}{\sqrt{2}}\phi_{0,0}(x) - \frac{1}{\sqrt{2}}\psi_{0,0}(x). \tag{3.1.20}$$

This is shown graphically in Figure 3.2, where $c_{0,0} = c_{1,0} = d_{0,0} = \frac{1}{\sqrt{2}}$ *and* $d_{1,0} = -\frac{1}{\sqrt{2}}$*.*

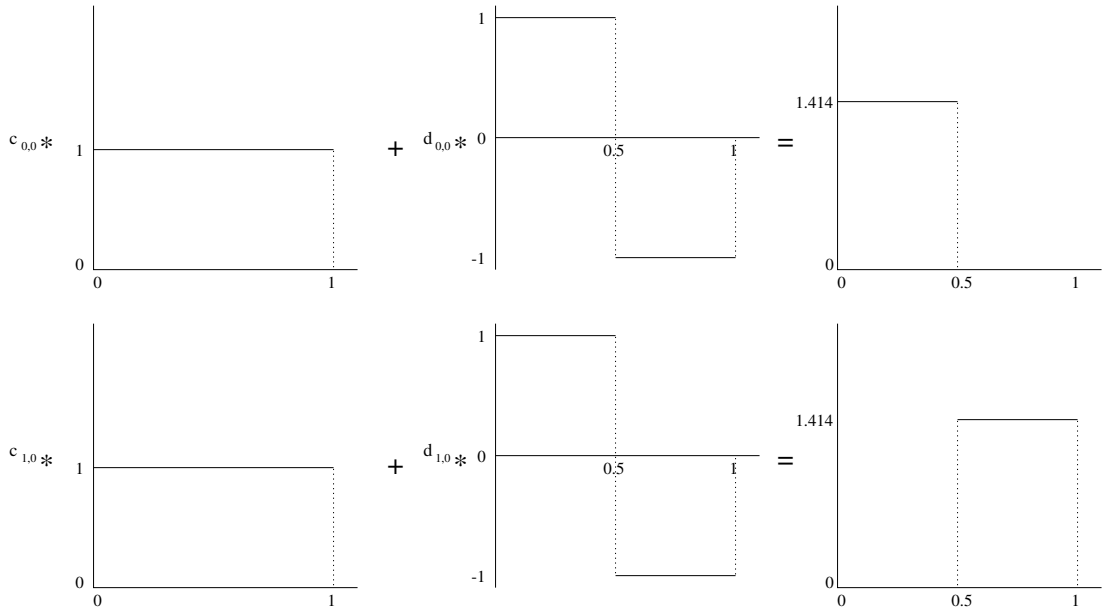


Figure 3.2: $V_1 = W_0 \oplus V_0$

3.1.3 Multiscale Transformations

The coefficient vectors $\boldsymbol{\alpha}_M$ and $\boldsymbol{\beta}_m$, for $m = 0, \ldots, M-1$, in (3.1.10) and (3.1.12), respectively, convey different information. The coefficients $\boldsymbol{\alpha}_M$ in (3.1.10) indicate the geometric location of the function f_M. However, the coefficients $\boldsymbol{\beta}_m$ represent the *difference* between the function representation at the current level and that of the previous level. That is, the wavelets encode the detail information, or the correction that must be added to the higher-level representation of a function. Therefore, we usually need all the entries of $\boldsymbol{\alpha}_M$ to obtain an accurate representation of f_M. However, many of the entries of $\boldsymbol{\beta}_m$ may be small and replacing such entries by zero may still permit a sufficiently accurate approximation to f_M. On the other hand, the pointwise evaluation of f_M is simpler in the single scale form. Therefore to exploit the benefits of both representations, one needs a method to convert one into the other.

Due to the nestedness of the spaces V_m and the stability condition (3.1.3), every $\phi_{m,l} \in V_m$ can be represented as an expansion of the functions $\phi_{m+1,i} \in V_{m+1}$. That is,

$$\phi_{m,l} = \sum_{i \in \Delta_{m+1}} c_{i,l} \phi_{m+1,i}, \tag{3.1.21}$$

with a mask $\mathbf{c}_l^m = \{c_{i,l}\}_{i \in \Delta_{m+1}} \in \ell_2(\Delta_{m+1})$. Let $\mathrm{C}_{m,0}$ be the so-called *refinement matrix* containing the $\mathbf{c}_l^m$ as columns, then (3.1.21) can be rewritten as

$$\boldsymbol{\Phi}_m^{\mathbf{T}} = \boldsymbol{\Phi}_{m+1}^{\mathbf{T}} \mathrm{C}_{m,0}. \tag{3.1.22}$$

Similarly, due to (3.1.4) every $\psi_{m,l} \in W_m$ can be represented as an expansion of the

functions $\phi_{m+1,i} \in V_{m+1}$. That is,

$$\psi_{m,l} = \sum_{i \in \Delta_{m+1}} d_{i,l} \phi_{m+1,i}, \tag{3.1.23}$$

with a mask $\mathbf{d}_l^m = \{d_{i,l}\}_{i \in \Delta_{m+1}} \in \ell_2(\Delta_{m+1})$. Let $\mathrm{C}_{m,1}$ be the matrix containing the $\mathbf{d}_l^m$ as columns, then (3.1.23) can be rewritten as

$$\mathbf{\Psi}_m^{\mathrm{T}} = \mathbf{\Phi}_{m+1}^{\mathrm{T}} \mathrm{C}_{m,1}. \tag{3.1.24}$$

Collectively (3.1.22) and (3.1.24) are known as the *two scale relationships*.

The decomposition $V_{m+1} = V_m \oplus W_m$ is equivalent to the fact that the operator $\mathrm{C}_m : \ell_2(\Delta_m) \times \ell_2(\nabla_m) \to \ell_2(\Delta_{m+1})$ is invertible, where

$$\mathrm{C}_m := (\mathrm{C}_{m,1}, \mathrm{C}_{m,0}), \tag{3.1.25}$$

and

$$\mathrm{C}_m \begin{pmatrix} \boldsymbol{\beta}_m \\ \boldsymbol{\alpha}_m \end{pmatrix} := \mathrm{C}_{m,1} \boldsymbol{\beta}_m + \mathrm{C}_{m,0} \boldsymbol{\alpha}_m, \tag{3.1.26}$$

for $\boldsymbol{\alpha}_m \in \ell_2(\Delta_m)$, $\boldsymbol{\beta}_m \in \ell_2(\nabla_m)$. Additionally the basis $\{\mathbf{\Phi}_m \cup \mathbf{\Psi}_m\}$, of the space V_{m+1}, is uniformly stable if and only if

$$\|\mathrm{C}_m\| = \mathcal{O}(1), \quad \left\|\mathrm{C}_m^{-1}\right\| = \mathcal{O}(1),\ m \to \infty, \tag{3.1.27}$$

see [29].

For convenience, let $\mathrm{G}_m := \mathrm{C}_m^{-1}$, where

$$\mathrm{G}_m = \begin{pmatrix} \mathrm{G}_{m,1} \\ \mathrm{G}_{m,0} \end{pmatrix}, \tag{3.1.28}$$

and

$$\mathrm{C}_m\mathrm{G}_m = \mathrm{C}_{m,1}\mathrm{G}_{m,1} + \mathrm{C}_{m,0}\mathrm{G}_{m,0} = \mathrm{I}. \tag{3.1.29}$$

Therefore, the matrix C_m describes a change of basis for the space V_{m+1}, from the basis $\{\boldsymbol{\Phi}_m \cup \boldsymbol{\Psi}_m\}$ to the basis $\boldsymbol{\Phi}_{m+1}$. The matrix G_m describes the reverse change. From relationship (3.1.29) and the two scale relationships (3.1.22) and (3.1.24), we obtain the reconstruction relationship

$$\boldsymbol{\Phi}_{m+1}^{\mathrm{T}} = \boldsymbol{\Phi}_m^{\mathrm{T}}\mathrm{G}_{m,0} + \boldsymbol{\Psi}_m^{\mathrm{T}}\mathrm{G}_{m,1}. \tag{3.1.30}$$

Relationships (3.1.22), (3.1.24) and (3.1.30) are transformations that can be used to convert the coefficients of (3.1.10) into the coefficients of (3.1.12) and vice versa. Let us now derive these explicitly here. A function $f_M \in V_M$ can be expanded in single scale as in (3.1.10), as well as in double scale form as

$$f_M = \boldsymbol{\Phi}_{M-1}^{\mathrm{T}}\boldsymbol{\alpha}_{M-1} + \boldsymbol{\Psi}_{M-1}^{\mathrm{T}}\boldsymbol{\beta}_{M-1}, \tag{3.1.31}$$

with respect to the basis $\{\boldsymbol{\Phi}_{M-1} \cup \boldsymbol{\Psi}_{M-1}\}$. Therefore, using (3.1.22) and (3.1.24), yields

$$f_M = \boldsymbol{\Phi}_{M-1}^{\mathrm{T}}\boldsymbol{\alpha}_{M-1} + \boldsymbol{\Psi}_{M-1}^{\mathrm{T}}\boldsymbol{\beta}_{M-1} = \boldsymbol{\Phi}_M^{\mathrm{T}}\left(\mathrm{C}_{M-1,0}\boldsymbol{\alpha}_{M-1} + \mathrm{C}_{M-1,1}\boldsymbol{\beta}_{M-1}\right). \tag{3.1.32}$$

Comparing the r.h.s. of (3.1.32) with (3.1.10), we obtain

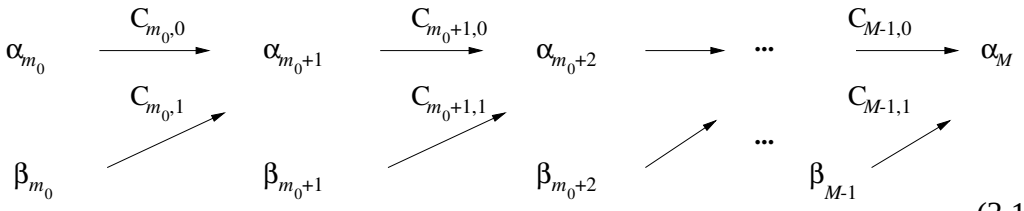
$$\mathrm{C}_{M-1,0}\boldsymbol{\alpha}_{M-1} + \mathrm{C}_{M-1,1}\boldsymbol{\beta}_{M-1} = \boldsymbol{\alpha}_M. \tag{3.1.33}$$

That is, the operator C_m applied to the coefficients $\binom{\boldsymbol{\beta}_m}{\boldsymbol{\alpha}_m}$ produces the coefficients $\boldsymbol{\alpha_{m+1}}$. Thus, repeated application of the operator C_m converts the coefficients of the multiscale form of f_M, (3.1.12), into the coefficients of the single scale form (3.1.10), giving the

transformation

$$\mathrm{R}_M : \boldsymbol{\beta}_{m_0,M-1} \rightarrow \boldsymbol{\alpha}_M, \tag{3.1.34}$$

where $\boldsymbol{\beta}_{m_0,M-1} = \begin{pmatrix} \boldsymbol{\beta}_{M-1} \\ \vdots \\ \boldsymbol{\beta}_{m_0} \\ \boldsymbol{\alpha}_{m_0} \end{pmatrix}$. The transformation (3.1.34) is called the *reconstruction* transformation, or reconstruction algorithm and is schematically given by



(3.1.35)

To express the transformation R_M in matrix form, for $m < M$, we define the $N_M \times N_M$ matrix

$$\mathrm{R}_{M,m} := \begin{pmatrix} \mathrm{I} & 0 \\ 0 & \mathrm{C}_m \end{pmatrix}, \tag{3.1.36}$$

where I is the identity matrix of size $N_M - N_{m+1}$. Then, the reconstruction transformation R_M in (3.1.34) can be written as

$$\mathrm{R}_M = \mathrm{R}_{M,M-1} \cdots \mathrm{R}_{M,m_0}. \tag{3.1.37}$$

The inverse transformation, transforms the single scale coefficients $\boldsymbol{\alpha}_M$ of f_M (see (3.1.10)) to the multiscale coefficients of f_M as in (3.1.12). Applying (3.1.30) to f_M in single scale form we obtain,

$$f_M = \boldsymbol{\Phi}_M^{\mathrm{T}} \boldsymbol{\alpha}_M$$

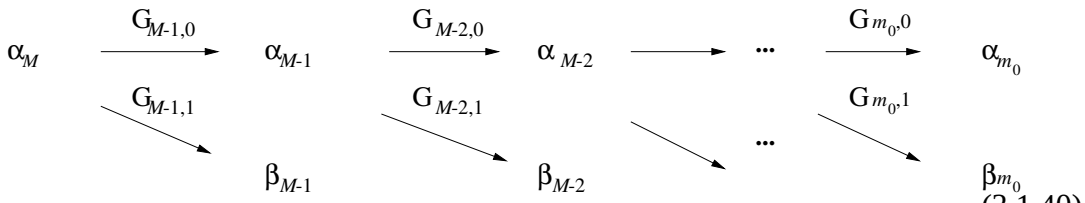
$$= \mathbf{\Phi}_{M-1}^{\mathrm{T}} \left(\mathrm{G}_{M-1,0} \boldsymbol{\alpha}_M \right) + \mathbf{\Psi}_{M-1}^{\mathrm{T}} \left(\mathrm{G}_{M-1,1} \boldsymbol{\alpha}_M \right)$$

$$= \mathbf{\Phi}_{M-1}^{\mathrm{T}} \boldsymbol{\alpha}_{M-1} + \mathbf{\Psi}_{M-1}^{\mathrm{T}} \boldsymbol{\beta}_{M-1}. \tag{3.1.38}$$

That is, the operator G_m applied to the coefficients $\boldsymbol{\alpha}_{m+1}$ results in the coefficients $\binom{\boldsymbol{\beta}_m}{\boldsymbol{\alpha}_m}$. Thus, repeated application of the operator G_m converts the coefficients of the single scale form of f_M, (3.1.10), into the coefficients of the multiscale form (3.1.12), giving the transformation

$$\mathrm{T}_M : \boldsymbol{\alpha}_M \rightarrow \boldsymbol{\beta}_{m_0,M-1}. \tag{3.1.39}$$

The inverse transformation (3.1.39) is called the *decomposition* transformation, or decomposition algorithm and is schematically given by



(3.1.40)

For $m < M$, we define the $N_M \times N_M$ matrix

$$\mathrm{T}_{M,m} := \begin{pmatrix} \mathrm{I} & 0 \\ 0 & \mathrm{G}_m \end{pmatrix}, \tag{3.1.41}$$

where I is the identity matrix of size $N_M - N_{m+1}$. Then, the decomposition transformation T_M in (3.1.39) can be written as

$$\mathrm{T}_M = \mathrm{T}_{M,m_0} \cdots \mathrm{T}_{M,M-1}. \tag{3.1.42}$$

There are infinitely many possible complement basis $\mathbf{\Psi}_m$ that yield the decomposition

(3.1.4). A constraint on the choice of $\mathbf{\Psi}_m$ is the stability of the multiscale transformations.

Theorem 3.1.1. ([30, Theorem 3.3]). *The transformations R_M and T_M are well conditioned in the sense that*

$$\|R_M\|, \|T_M\| = \mathcal{O}(1), \quad M \to \infty, \tag{3.1.43}$$

if and only if the collection $\mathbf{\Psi}$ of (3.1.13) is a Riesz basis of H.

Example 3.1.2. *Using the Haar basis of Example 3.1.1, we obtain the decomposition and reconstruction algorithms for the Haar basis.*

When we consider the Haar basis, the two scale relationships (3.1.22) and (3.1.24) become

$$\phi_\lambda = c_{0,0}\phi_{m+1,2l}(x) + c_{1,0}\phi_{m+1,2l+1}(x), \tag{3.1.44}$$

$$\psi_\lambda = d_{0,0}\phi_{m+1,2l}(x) + d_{1,0}\phi_{m+1,2l+1}(x), \tag{3.1.45}$$

for $\lambda = \{m, l\}$, $m \in \mathbb{Z}$, $l = 0, \ldots, 2^m - 1$. To find the coefficients $c_{0,0}$ and $c_{1,0}$, the equidistant values $x_1 = \frac{1}{3}$ and $x_2 = \frac{2}{3}$ are substituted into relationship (3.1.44). Hence, for $m = 0$, we obtain the linear system

$$\begin{aligned} \phi_{0,0}\left(\frac{1}{3}\right) &= c_{0,0}\phi_{1,0}\left(\frac{1}{3}\right) + c_{1,0}\phi_{1,1}\left(\frac{1}{3}\right), \\ \phi_{0,0}\left(\frac{2}{3}\right) &= c_{0,0}\phi_{1,0}\left(\frac{2}{3}\right) + c_{1,0}\phi_{1,1}\left(\frac{2}{3}\right). \end{aligned} \tag{3.1.46}$$

Solving (3.1.46), we find $c_{0,0} = c_{1,0} = \frac{1}{\sqrt{2}}$. Similarly, using relationship (3.1.45), we find $d_{0,0} = \frac{1}{\sqrt{2}}$ and $d_{1,0} = -\frac{1}{\sqrt{2}}$.

We now derive the decomposition algorithm for the Haar basis. Given a function f, we

consider the projection coefficients of f onto the space V_m,

$$\begin{aligned}\alpha_\lambda &= \int_{I_\lambda} f(x)\phi_\lambda(x)\ dx \\ &= c_{0,0}\int_{I_{m+1,2l}} f(x)\phi_{m+1,2l}(x)\ dx + c_{1,0}\int_{I_{m+1,2l+1}} f(x)\phi_{m+1,2l+1}\ dx \\ &= c_{0,0}\alpha_{m+1,2l} + c_{1,0}\alpha_{m+1,2l+1}, \end{aligned} \tag{3.1.47}$$

for $\lambda = \{m, l\}$, $m \in \mathbb{Z}$, $l = 0, \ldots, 2^m - 1$. Similarly, the projection coefficients of f onto the space W_m are

$$\begin{aligned}\beta_\lambda &= \int_{I_\lambda} f(x)\psi_\lambda(x)\ dx \\ &= d_{0,0}\int_{I_{m+1,2l}} f(x)\phi_{m+1,2l}(x)\ dx + d_{1,0}\int_{I_{m+1,2l+1}} f(x)\phi_{m+1,2l+1}(x)\ dx \\ &= d_{0,0}\alpha_{m+1,2l} + d_{1,0}\alpha_{m+1,2l+1}, \end{aligned} \tag{3.1.48}$$

for $\lambda = \{m, l\}$, $m \in \mathbb{Z}$, $l = 0, \ldots, 2^m - 1$. Relationships (3.1.47) and (3.1.48) together are the decomposition algorithm in filter form for the Haar basis. In matrix form, for $M = 3$, the decomposition algorithm is $T_3 = T_{3,0}T_{3,1}T_{3,2}$, where

$$T_{3,0} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$T_{3,1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$T_{3,2} = \begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

We now derive the reconstruction algorithm for the Haar basis. Due to the decomposition (3.1.4), a function $f \in V_{m+1}$ has two distinct representation, namely,

$$f(x) = \sum_{l'=0}^{2^{m'}-1} \alpha_{\lambda'} \phi_{\lambda'}(x), \tag{3.1.49}$$

where $\lambda' = \{m', l'\}$ with $l' = 0, \ldots, 2^{m'} - 1$, here $m' = m + 1$; and

$$f(x) = \sum_{l=0}^{2^{m}-1} \left(\alpha_{\lambda} \phi_{\lambda}(x) + \beta_{\lambda} \psi_{\lambda}(x) \right), \tag{3.1.50}$$

where $\lambda = \{m, l\}$ with $l = 0, \ldots, 2^m - 1$. Therefore, applying the two scale relationship, (3.1.44) and (3.1.45), to equation (3.1.50), we obtain

$$f(x) = \sum_{l=0}^{2^m-1} \left(c_{0,0}\alpha_\lambda + d_{0,0}\beta_\lambda\right) \phi_{m+1,2l}(x) + \left(c_{1,0}\alpha_\lambda + d_{1,0}\beta_\lambda\right) \phi_{m+1,2l+1}(x). \quad (3.1.51)$$

Then, comparing equations (3.1.49) and (3.1.51) we obtain the following reconstruction algorithm, in filter form,

$$\begin{aligned} \alpha_{m+1,2l} &= c_{0,0}\alpha_\lambda + d_{0,0}\beta_\lambda \\ \alpha_{m+1,2l+1} &= c_{1,0}\alpha_\lambda + d_{1,0}\beta_\lambda. \end{aligned} \quad (3.1.52)$$

In matrix form, for $M = 3$, the reconstruction algorithm is $R_3 = R_{3,2}R_{3,1}R_{3,0}$, where

$$R_{3,2} = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 \\ -\frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 0 & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$R_{3,1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$R_{3,0} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

To illustrate a use of the decomposition and reconstruction algorithms, we consider the function $f(x) = 2\cos 2\pi x + \sin 2\pi x$*. We approximate the function* $f(x)$ *in the space* V_3*,*

in multiscale form, that is, we use the basis $\mathbf{\Psi}^3$. The resulting multiscale coefficients are

$$\begin{pmatrix} \beta_{2,0} \\ \beta_{2,1} \\ \beta_{2,2} \\ \beta_{2,3} \\ \beta_{1,0} \\ \beta_{1,1} \\ \beta_{0,0} \\ \alpha_{0,0} \end{pmatrix} = \begin{pmatrix} 0.131842719 \\ 0.3955448157 \\ -0.1318482721 \\ -0.3955448156 \\ 0.9003163161 \\ -0.9003163163 \\ 0.6366197718 \\ 0 \end{pmatrix}.$$

Therefore, applying the reconstruction algorithm to the multiscale coefficients, we obtain the single scale coefficients,

$$R_3 \begin{pmatrix} 0.131842719 \\ 0.3955448157 \\ -0.1318482721 \\ -0.3955448156 \\ 0.9003163161 \\ -0.9003163163 \\ 0.6366197718 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.7384680443 \\ 0.5820064299 \\ 0.0546133422 \\ -0.5047715007 \\ -0.7684680444 \\ -0.5820064297 \\ -0.0546133419 \\ 0.5047715010 \end{pmatrix}.$$

Applying the decomposition algorithm to the single scale coefficients, we re-obtain the multiscale coefficients.

The transformations R_M and T_M in the present form are for theoretical analysis. In practice the matrices R_M and T_M are not computed, instead local filters are applied, see section 3.2.3

3.1.4 Wavelets on $[0, 1]$

The wavelets we have discussed so far are defined on $\mathbb{R}$. However, we are concerned with integral equations defined over a subset of $\mathbb{R}$. Therefore, we require wavelets defined on a closed interval.

There are several methods which adapt wavelets defined on $\mathbb{R}$ to wavelets defined on an interval; see [31, 32, 33, 34]. Here, following Daubechies [26], we briefly introduce two methods for defining wavelets on the interval $[0, 1]$. The first and most basic method is to use wavelets defined over $\mathbb{R}$, with the function f set to zero outside of $[0, 1]$. However, this method introduces a discontinuity in the function at the interval's boundary. This leads to large wavelet coefficients for wavelets whose support overlaps the interval's boundary. Furthermore, this method is not computationally efficient.

The second method is the so-called *periodized wavelets*. We start with the scaling function ϕ and the wavelet ψ defined over $\mathbb{R}$. For $m = 0, 1, \ldots$, we define the periodized scaling function and wavelet as

$$\phi_\lambda^{\text{per}}(x) = \sum_{i \in \Delta_m} \phi_\lambda(x + i) = 2^{\frac{m}{2}} \sum_{i \in \Delta_m} \phi(2^m x - 2^m i - l), \tag{3.1.53}$$

for $\lambda = \{m, l\}$, $l \in \Delta_m$ and

$$\psi_\lambda^{\text{per}}(x) = \sum_{i \in \nabla_m} \psi_\lambda(x + i) = 2^{\frac{m}{2}} \sum_{i \in \nabla_m} \psi(2^m x - 2^m i - l), \tag{3.1.54}$$

for $\lambda = \{m, l\}$, $l \in \nabla_m$. Here, $\Delta_m = \nabla_m = \mathbb{Z}$. Clearly, both $\phi_\lambda^{\text{per}}$ and $\psi_\lambda^{\text{per}}$ are periodic of period 1. The spaces V_m^{per} and W_m^{per} are defined as

$$V_m^{\text{per}} := \overline{\text{span}\{\phi_\lambda^{\text{per}} |\ \lambda = \{m, l\},\ l \in \mathbb{Z}\}}, \tag{3.1.55}$$

and

$$W_m^{\text{per}} := \overline{\text{span}\{\psi_\lambda^{\text{per}} |\ \lambda = \{m, l\},\ l \in \mathbb{Z}\}}, \tag{3.1.56}$$

respectively.

Since the scaling function and wavelet are periodic with period 1, $\phi_{m,l+2^m}^{\text{per}} = \phi_{m,l}^{\text{per}}$ and $\psi_{m,l+2^m}^{\text{per}} = \psi_{m,l}^{\text{per}}$, the spaces V_m^{per} and W_m^{per} are 2^m-dimensional spaces, [26, 23]. These spaces inherit the multiresolution properties of the non-periodized spaces V_m and W_m. That is,

$$V_0^{\text{per}} \subset V_1^{\text{per}} \subset \ldots \subset L_2[0,1], \tag{3.1.57}$$

$$\overline{\bigcup_{m=0}^{\infty} V_m^{\text{per}}} = L_2[0,1] \tag{3.1.58}$$

and

$$V_m^{\text{per}} = V_{m-1}^{\text{per}} \oplus W_{m-1}^{\text{per}}. \tag{3.1.59}$$

In the remaining section of this chapter we discuss the bases we use in this thesis, namely, the multiwavelets. These are a wavelet basis developed for the interval $[0, 1]$.

3.2 Multiwavelets on $[0, 1]$

Wavelets are attractive for the numerical solution of integral equations, because their vanishing moments property leads to operator compression [21, 35]. However, to obtain wavelets with compact support and high order of vanishing moments, the length of the support increases as the order of the vanishing moments increases, [26, 36]. This causes difficulties with the practical use of wavelets particularly at edges and corners. To avoid

such problems, we consider the orthonormal multiwavelets of [10]. With such basis functions, the order of vanishing moments d is related to the number of *mother* wavelets rather than the size of the compact support.

Suppose k is a positive integer and m a non-negative integer, we define the space V_m^k of piecewise polynomial functions

$$V_m^k := \left\{ \begin{array}{c} g : g|_{[2^{-m}l, 2^{-m}(l+1)]} \text{ is a polynomial of degree less than } k \\ \forall\ l = 0, 1, \ldots, 2^m - 1 \text{ and vanishes elsewhere} \end{array} \right\}.$$

It is clear that the spaces V_m^k have dimension $2^m k$ and are nested subspaces such that,

$$V_0^k \subset V_1^k \subset \ldots \subset V_m^k \subset \ldots \subset L_2[0,1]. \tag{3.2.1}$$

For $m = 0, 1, 2, \ldots$, we define the $2^m k$ dimensional space W_m^k to be the orthogonal complement of V_m^k in V_{m+1}^k; that is

$$V_{m+1}^k = V_m^k \oplus W_m^k. \tag{3.2.2}$$

Then, we have the decomposition

$$V_m^k = V_0^k \oplus W_0^k \oplus W_1^k \oplus \ldots \oplus W_{m-1}^k. \tag{3.2.3}$$

The space V_0^k is the space of polynomials of degree less than k on the interval [0,1] and we assume $\{\phi_1, \phi_2, \ldots, \phi_k\}$ to be a basis for it. Suppose $\{\psi_1, \psi_2, \ldots, \psi_k\}$ is a basis of W_0^k. Therefore, for the orthogonality condition $V_0^k \perp W_0^k$ to be satisfied we require the

first k moments of $\{\psi_1, \ldots, \psi_k\}$ to vanish. That is

$$\int_0^1 \psi_j(x) x^i dx = 0 \qquad \text{for } j = 1, 2, \ldots, k;\ i = 0, 1, \ldots, k-1. \tag{3.2.4}$$

The $2k$-dimensional space W_1^k is spanned by the functions $\{\psi_1(2x), \ldots, \psi_k(2x), \psi_1(2x-1), \ldots, \psi_k(2x-1)\}$. In general, if we define

$$\psi_\lambda := 2^{\frac{m}{2}} \psi_j(2^m x - l), \qquad \text{where } \lambda := \{j, m, l\}, \tag{3.2.5}$$

the space W_m^k is spanned by the set

$$\mathbf{\Psi}_m := \{\psi_\lambda |\ l = 0, \ldots, 2^m - 1,\ j = 1, \ldots, k\}. \tag{3.2.6}$$

Therefore, the wavelet spaces $\{W_m^k\}$ are generated from the k mother wavelets $\{\psi_1, \psi_2, \ldots, \psi_k\}$. Similarly the spaces $\{V_m^k\}$ can be generated from the scaling functions $\{\phi_1, \phi_2, \ldots, \phi_k\}$, as the span of the set

$$\mathbf{\Phi}_m := \left\{\phi_\lambda = 2^{\frac{m}{2}} \phi_j(2^m x - l) |\ l = 0, \ldots, 2^m - 1,\ j = 1, \ldots, k\right\}. \tag{3.2.7}$$

Note that when dealing with multiwavelets the parameter λ is a triplet $\lambda = \{j, m, l\}$ identifying the *mother* wavelet, i.e. $j = 1, \ldots, k$, the level, i.e. $m = 0, \ldots, M$ and the location, $l = 0, \ldots, 2^m - 1$ indicating the span of the wavelet is over $[2^{-m} l, 2^{-m}(l+1)]$.

Let us now define the set of basis functions $\mathbf{\Psi}^M$ as follows:

$$\mathbf{\Psi}^M := \mathbf{\Phi}_0 \bigcup_{m=0}^{M-1} \mathbf{\Psi}_m. \tag{3.2.8}$$

It is easy to see that both $\mathbf{\Phi}_M$ and $\mathbf{\Psi}^M$ are bases for the $2^M k$-dimensional space V_M^k.

3.2.1 Multiwavelet Construction

We now show one possible way to generate $\psi_1, \psi_2, \ldots, \psi_k$ as proposed in [10]. First, we construct k functions $g_1, g_2, \ldots, g_k$ defined from $\mathbb{R}$ to $\mathbb{R}$, with compact support on $[-1, 1]$, satisfying the following conditions:

1. The restriction of g_j to $(0, 1)$ is a polynomial of degree $k - 1$.

2. The function g_j is extended to the interval $(-1, 0)$ as an even or odd function according to whether $i + k - 1$ is even or odd, respectively.

3. The functions $g_1, g_2, \ldots, g_k$ satisfy the following orthonormality conditions:

$$\int_{-1}^{1} g_i(x) g_j(x)\, dx = \delta_{ij}, \quad i, j = 1, \ldots, k.$$

4. The function g_j has the following vanishing moment properties:

$$\int_{-1}^{1} g_j(x) x^i\, dx = 0, \quad i = 0, 1, \ldots, j + k - 2;\ j = 1, \ldots, k.$$

We determine g_j constructively. Suppose we have $2k$ functions, $1, x, \ldots, x^{k-1}, g_1^1, \ldots, g_k^1$, which span the space of polynomials of degree less than k on the intervals $(0, 1)$ and $(-1, 0)$. Then, we first orthogonalize k of them to the functions $1, x, \ldots, x^{k-1}$ then to the functions $x^k, x^{k+1}, \ldots, x^{2k+1}$ and finally to themselves. We define g_j^1 as follows:

$$g_j^1(x) = \begin{cases} x^{j-1}, & x \in (0, 1), \\ -x^{j-1}, & x \in (-1, 0), \\ 0, & \text{otherwise}, \end{cases} \quad \text{for } j = 1, \ldots, k. \tag{3.2.9}$$

Note that the $2k$ functions $1, x, \ldots, x^{k-1}, g_1^1, \ldots, g_k^1$ are linearly independent. Therefore, they span the space of functions which are polynomials of degree less than k on $(0,1)$ and on $(-1,0)$. Then;

1. By the Gram-Schmidt process we orthogonalize g_j^1 with respect to $1, x, \ldots, x^{k-1}$ over $(-1,1)$, obtaining g_j^2 for $j = 1, \ldots, k$.

2. Using the following sequence of steps we obtain $k-1$ functions orthogonal to x^k, of which $k-2$ functions are orthogonal to x^{k+1}, and so forth, down to one function which is orthogonal to x^{2k-2}. We proceed in the following manner: If at least one of the functions g_j^2 is not orthogonal to x^k, we reorder the functions so that it appears first. We define $g_j^3 = g_j^2 - a_j \cdot g_1^2$, where a_j is chosen such that $\langle g_j^3, x^k \rangle = 0$ for $j = 2, \ldots, k$. Therefore, obtaining the desired orthogonality to x^k. In the same way, we orthogonalize to $x^{k+1}, \ldots, x^{2k-1}$ to obtain $g_1^2, g_2^3, g_3^4, \ldots, g_k^{k+1}$, such that $\langle g_j^{j+1}, x^i \rangle = 0$ for $i \leq j + k - 2$.

3. In the final step we apply the Gram-Schmidt orthogonalization algorithm to g_k^{k+1}, $g_{k-1}^k, \ldots, g_1^2$. We then normalize these functions to obtain $g_k, g_{k-1}, \ldots, g_1$.

If we now define,

$$\psi_j(x) = \sqrt{2} g_j(2x-1), \quad j = 1, ..., k, \quad x \in [0,1], \tag{3.2.10}$$

we have obtained a basis for W_0^k. Here, we present g_j for $k = 1, \ldots, 4$.

$$g_1(x) = \begin{cases} \sqrt{\frac{1}{2}} \text{ for } x \in (0,1), \\ -\sqrt{\frac{1}{2}} \text{ for } x \in (-1,0). \end{cases} \tag{3.2.11}$$

$$k = 1$$

$$g_1 = \begin{cases} \sqrt{\frac{3}{2}}(-1+2x) \ \text{ for } x \in (0,1), \\ -\sqrt{\frac{3}{2}}(1+2x) \ \text{ for } x \in (-1,0), \end{cases}$$
$$g_2 = \begin{cases} \sqrt{\frac{1}{2}}(-2+3x) \ \text{ for } x \in (0,1), \\ \sqrt{\frac{1}{2}}(2+3x) \ \text{ for } x \in (-1,0). \end{cases} \tag{3.2.12}$$

$$k = 2$$

$$g_1 = \begin{cases} \frac{1}{3}\sqrt{\frac{1}{2}}(1-24x+30x^2) \ \text{ for } x \in (0,1), \\ -\frac{1}{3}\sqrt{\frac{1}{2}}(1+24x+30x^2) \ \text{ for } x \in (-1,0), \end{cases}$$
$$g_2 = \begin{cases} \frac{1}{2}\sqrt{\frac{3}{2}}(3-16x+15x^2) \ \text{ for } x \in (0,1), \\ \frac{1}{2}\sqrt{\frac{3}{2}}(3+16x+15x^2) \ \text{ for } x \in (-1,0). \end{cases} \tag{3.2.13}$$
$$g_3 = \begin{cases} \frac{1}{3}\sqrt{\frac{5}{2}}(4-15x+12x^2) \ \text{ for } x \in (0,1), \\ -\frac{1}{3}\sqrt{\frac{5}{2}}(4+15x+12x^2) \ \text{ for } x \in (-1,0). \end{cases}$$

$$k = 3$$

$$
\begin{aligned}
g_1(x) &= \begin{cases} \sqrt{\frac{15}{34}}(28x^3 - 30x^2 + 4x + 1) \ \text{ for } x \in (0,1), \\ \sqrt{\frac{15}{34}}(-28x^3 - 30x^2 - 4x + 1) \ \text{ for } x \in (-1,0), \end{cases} \\
g_2(x) &= \begin{cases} \sqrt{\frac{1}{42}}(210x^3 - 300x^2 + 105x - 4) \ \text{ for } x \in (0,1), \\ \sqrt{\frac{1}{42}}(210x^3 + 300x^2 + 105x + 4) \ \text{ for } x \in (-1,0), \end{cases} \\
g_3(x) &= \begin{cases} \frac{1}{2}\sqrt{\frac{35}{34}}(64x^3 - 105x^2 + 48x - 5) \ \text{ for } x \in (0,1), \\ \frac{1}{2}\sqrt{\frac{35}{34}}(-64x^3 - 105x^2 - 48x - 5) \ \text{ for } x \in (-1,0), \end{cases} \\
g_4(x) &= \begin{cases} \frac{1}{2}\sqrt{\frac{5}{42}}(105x^3 - 192x^2 + 105x - 16) \ \text{ for } x \in (0,1), \\ \frac{1}{2}\sqrt{\frac{5}{42}}(105x^3 + 192x^2 + 105x + 16) \ \text{ for } x \in (-1,0). \end{cases}
\end{aligned}
\tag{3.2.14}
$$

$$k = 4$$

3.2.2 Multiwavelet Approximation

Given a function $f \in L_2[0,1]$, the projection $P_m^k f$ of f onto V_m^k is

$$\left(P_m^k f\right)(x) = \sum_{l,j} \langle f, \phi_\lambda \rangle \phi_\lambda(x). \tag{3.2.15}$$

Then the following result can be proved.

Lemma 3.2.1. *Suppose that the function $f : [0,1] \to \mathbb{R}$ is k times continuously differentiable. Then, $P_m^k f$ approximates f with the following error bound:*

$$\left\| P_m^k f - f \right\|_{L_2} \leq 2^{-mk} \frac{2}{4^k k!} \sup_{x \in [0,1]} \left| f^{(k)}(x) \right|. \tag{3.2.16}$$

Proof: The interval $[0,1]$ is divided into subintervals $I_{m,l}$, such that $P_m^k f$ is a polynomial

of degree less than k that approximates f with minimum mean error. We then use the maximum error estimate for the polynomial of degree k which agrees with f at Chebyshev nodes of order k on $I_{m,l}$. We define $I_{m,l} = [2^{-m}l, 2^{-m}(l+1))$ for $l = 0, \ldots, 2^m - 1$. Then, we obtain

$$\begin{aligned}
\|P_m f - f\|_{L_2}^2 &= \int_0^1 [P_m f(x) - f(x)]^2 dx \\
&= \sum_l \int_{I_{m,l}} [P_m f(x) - f(x)]^2 dx \\
&\leq \sum_l \int_{I_{m,l}} [C_{m,l}^k f(x) - f(x)]^2 dx \\
&\leq \sum_l \int_{I_{m,l}} \left(\frac{2^{1-mk}}{4^k k!} \sup_{x \in I_{m,l}} \left| f^{(k)}(x) \right| \right)^2 dx \\
&\leq \left(\frac{2^{1-mk}}{4^k k!} \sup_{x \in [0,1]} \left| f^{(k)}(x) \right| \right)^2 .
\end{aligned}$$

Then, by taking square roots we obtain bound (3.2.16). Here, $C_{m,l}^k f$ denotes the polynomial of degree k which agrees with f at the Chebyshev nodes of order k on $I_{m,l}$. □

Therefore, when using the multiwavelet basis to approximate f the error decays like 2^{-mk}.

3.2.3 Multiwavelet Transformations

As seen in section 3.1.3 for a given wavelet basis we can obtain multiscale transformations R_m and T_m, that allow us to move between single and multiscale representations of a function. However, in practice we do not form the matrices R_m and T_m, instead we apply local filters. In this section we follow [37] in developing such filters for multiwavelet bases.

We start by rewriting the two scale relationships (3.1.22) and (3.1.24) in terms of the

individual basis functions. That is,

$$\phi_\lambda(x) = \sum_{\substack{j'=1,\dots,k \\ l'=2l,2l+1}} c_{j,j'+kl'-2kl}\phi_{\{j',m+1,l'\}}(x) \tag{3.2.17}$$

and

$$\psi_\lambda(x) = \sum_{\substack{j'=1,\dots,k \\ l'=2l,2l+1}} d_{j,j'+kl'-2kl}\phi_{\{j',m+1,l'\}}(x) \tag{3.2.18}$$

where $\lambda = \{j, m, l\}$, $j = 1, \dots, k,\ l = 0, \dots, 2^m - 1$. In (3.2.17) to find the $2k$ unknowns $\{c_{j,1}, c_{j,2}, \dots, c_{j,2k}\}$, for each $j = 1, \dots, k$, we solve $2k \times 2k$ linear systems

$$\phi_j(x_i) = \sqrt{2} \sum_{\substack{j'=1,\dots,k \\ l'=0,1}} c_{j,j'+kl'}\phi_{j'}\left(2x_i - 1\right), \quad i = 1, \dots, 2k. \tag{3.2.19}$$

The points x_i are equidistant in $[0, 1]$ and given by $x_i = \frac{i}{2k+1}$. In the same way we can find $\{d_{j,1}, d_{j,2}, \dots, d_{j,2k}\}$.

Next, we consider the reconstruction transform R_m. Due to decomposition (3.1.4) there are two distinct basis for the space V_{m+1}^k, namely $\mathbf{\Phi}_{m+1}$ and $\{\mathbf{\Phi}_m \cup \mathbf{\Psi}_m\}$. Therefore, every $f \in V_{m+1}^k$ has two distinct representations,

$$f(x) = \sum_{\lambda'} \alpha_{\lambda'}\phi_{\lambda'}(x), \tag{3.2.20}$$

and

$$f(x) = \sum_{\lambda} \left(\alpha_\lambda\phi_\lambda(x) + \beta_\lambda\psi_\lambda(x)\right), \tag{3.2.21}$$

where $\lambda' = \{j', m+1, l'\}$, $j' = 1, \dots, k,\ l' = 0, \dots, 2^{m+1} - 1$ and $\lambda = \{j, m, l\}$

$j = 1, \ldots, k,\ l = 0, \ldots, 2^m - 1$. Applying (3.2.17) and (3.2.18) to (3.2.21), we obtain

$$\begin{aligned} f(x) &= \sum_{\lambda} \sum_{\substack{j'=1,\ldots,k \\ l'=2l,2l+1}} \left(c_{j,j'+kl'-2kl}\alpha_\lambda + d_{j,j'+kl'-2kl}\beta_\lambda\right) \phi_{\{j',m+1,l'\}}(x) \\ &= \sum_{\lambda} \Big\{ \left(c_{j,1}\alpha_\lambda + d_{j,1}\beta_\lambda\right) \phi_{\{1,m+1,2l\}}(x) + \ldots + \left(c_{j,k}\alpha_\lambda + d_{j,k}\beta_\lambda\right) \phi_{\{k,m+1,2l\}}(x) \\ &\qquad + \left(c_{j,k+1}\alpha_\lambda + d_{j,k+1}\beta_\lambda\right) \phi_{\{1,m+1,2l+1\}}(x) + \ldots \\ &\qquad + \left(c_{j,2k}\alpha_\lambda + d_{j,2k}\beta_\lambda\right) \phi_{\{k,m+1,2l+1\}}(x) \Big\}. \end{aligned} \tag{3.2.22}$$

Then, comparing (3.2.22) and (3.2.20) we obtain the following reconstruction relationship. For $j' = 1, \ldots, k,\ l = 0, \ldots, 2^m - 1$,

$$\begin{aligned} \alpha_{\{j',m+1,2l\}} &= \sum_{j=1,\ldots,k} \left(c_{j,j'}\alpha_\lambda + d_{j,j'}\beta_\lambda\right), \\ \alpha_{\{j',m+1,2l+1\}} &= \sum_{j=1,\ldots,k} \left(c_{j,j'+k}\alpha_\lambda + d_{j,j'+k}\beta_\lambda\right), \end{aligned} \tag{3.2.23}$$

for $\lambda = \{j, m, l\}$. Hence, (3.2.23) is the filter representation of (3.1.25) for multiwavelet bases. Therefore, repeated application of the local filter (3.2.23) converts the coefficients of the multiscale representation of f_M into the coefficients of the single scale representation of f_M.

Finally, we consider the decomposition transformation T_m. The projection coefficients of f onto the space V_m^k are

$$\alpha_\lambda = \int_{I_\lambda} f(x)\phi_\lambda(x)dx, \tag{3.2.24}$$

and the projection coefficients onto the space W_m^k are

$$\beta_\lambda = \int_{I_\lambda} f(x)\psi_\lambda(x)dx. \tag{3.2.25}$$

Then, applying the two scale relationship (3.2.17) to (3.2.24), we obtain

$$\begin{aligned}\alpha_\lambda &= \sum_{\substack{j'=1,\dots,k\\ l'=2l,2l+1}} c_{j,j'+kl'-2kl} \int_{I_\lambda} f(x)\phi_{\{j',m+1,l'\}}(x)dx \\ &= \sum_{\substack{j'=1,\dots,k\\ l'=2l,2l+1}} c_{j,j'+kl'-2kl}\alpha_{\{j',m+1,l'\}}. \end{aligned} \tag{3.2.26}$$

Similarly, applying the two scale relationship (3.2.18) to (3.2.25), we obtain

$$\begin{aligned}\beta_\lambda &= \sum_{\substack{j'=1,\dots,k\\ l'=2l,2l+1}} d_{j,j'+kl'-2kl} \int_{I_\lambda} f(x)\phi_{\{j',m+1,l'\}}(x)dx \\ &= \sum_{\substack{j'=1,\dots,k\\ l'=2l,2l+1}} d_{j,j'+kl'-2kl}\alpha_{\{j',m+1,l'\}}. \end{aligned} \tag{3.2.27}$$

Hence, (3.2.26) and (3.2.27) are the filter representation of (3.1.28) for multiwavelet bases. Therefore, repeated application of the local filter, (3.2.26) and (3.2.27), converts the coefficients of the single scale representation of f_M into the coefficients of the multi-scale representation of f_M.

Example 3.2.1. (Multiwavelet Basis).

To show the compression power of using a multiwavelet basis we consider the multiwavelet and scaling function representations of a function,

$$f(x) = \begin{cases} \frac{3}{2}\sin 4\pi x & \textit{for } 0 \le x \le \frac{1}{2}, \\ \frac{31}{5}\sin 80\pi x & \textit{for } \frac{1}{2} < x \le \frac{21}{40}, \\ \frac{3}{2}\sin 4\pi(x-\frac{1}{40}) & \textit{for } \frac{21}{40} < x \le 1, \\ 0 & \textit{otherwise}, \end{cases} \tag{3.2.28}$$

as shown in Figure 3.3.

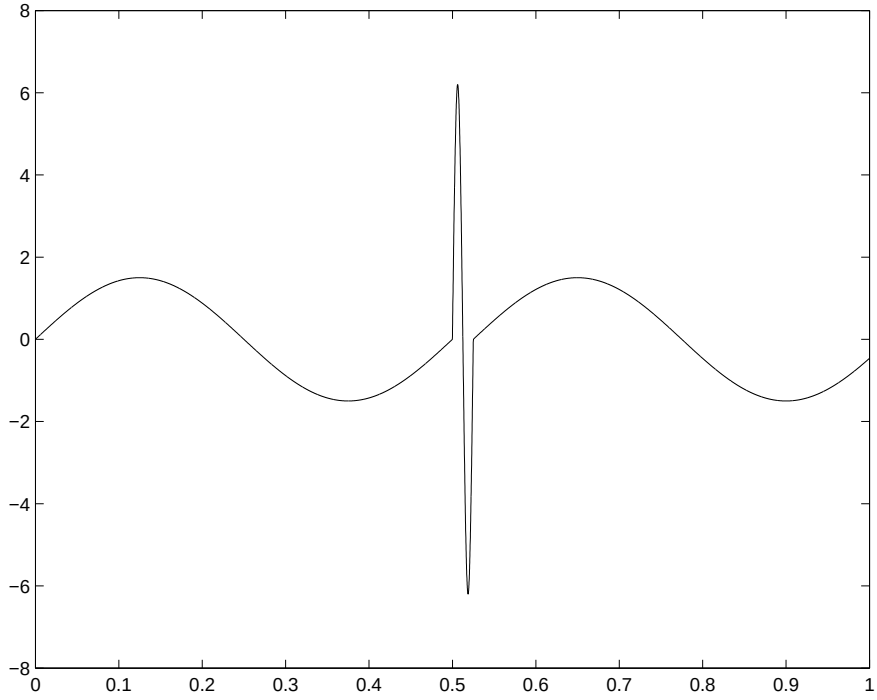


Figure 3.3: the function $f(x)$ of (3.2.28)

We consider the multiwavelet basis with vanishing moments of order 4*, that is,* $k = 4$*. Therefore, the* mother *multiwavelets, for* $k = 4$ *shown in Figure 3.4, are obtained from (3.2.10), using the functions* g_i*,* $i = 1, \ldots, k$ *in (3.2.14).*

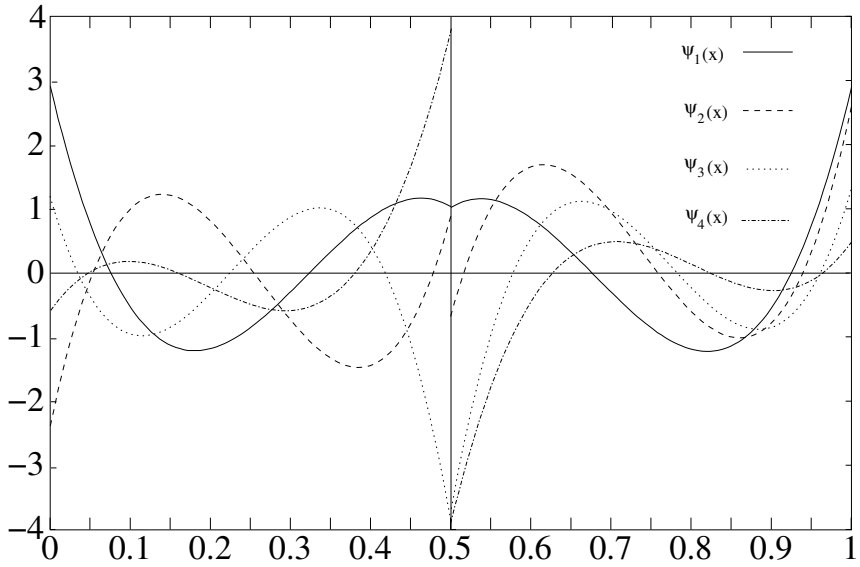


Figure 3.4: The Mother Wavelets

The corresponding scaling functions are

$$\begin{aligned}
\phi_1(x) &= 1, \\
\phi_2(x) &= \sqrt{3}(2x-1), \\
\phi_3(x) &= \frac{\sqrt{20}}{2}(6x^2-6x+1), \\
\phi_4(x) &= \frac{\sqrt{28}}{2}(20x^3-30x^2+12x-1).
\end{aligned} \tag{3.2.29}$$

We approximate the function $f(x)$ in the spaces V_M, for $M = 6, \ldots, 10$. Table 3.1 contains the results when the function is approximated using the scaling functions. The discretisation error is denoted $\|f - f_\phi\|$, the error introduced when setting elements less than a given tolerance to zero is denoted $\left\|f_\phi - \overline{f}_\phi\right\|$. The total error is denoted $\left\|f - \overline{f}_\phi\right\|$. The column $nz\ b4\ tol$ shows the number of non-zero coefficients before the tolerance is

applied, tol shows the tolerance applied and nz is the remaining number of non-zero coefficients.

M	$\|f - f_\phi\|$	nz *b4* tol	tol	$\|f_\phi - \overline{f}_\phi\|$	$\|f - \overline{f}_\phi\|$	nz
5	3.53×10^{-1}	128	1×10^{-1}	1.88×10^{-1}	3.94×10^{-1}	27
6	6.15×10^{-2}	256	1×10^{-2}	4.63×10^{-2}	7.87×10^{-2}	80
7	1.19×10^{-2}	512	2×10^{-3}	8.52×10^{-3}	1.37×10^{-2}	216
8	5.84×10^{-3}	1024	8×10^{-4}	5.07×10^{-3}	7.92×10^{-3}	416
9	1.50×10^{-3}	2048	2×10^{-4}	1.44×10^{-3}	1.96×10^{-3}	893
10	7.14×10^{-4}	4096	6×10^{-5}	5.54×10^{-4}	9.33×10^{-4}	1830

Table 3.1: Approximation of $f(x)$ using the scaling function basis

Table 3.2 contains the results when the function is approximated using the multiwavelet basis. The discretisation error is denoted $\|f - f_\psi\|$, the error introduced when setting elements less than a given tolerance to zero is denoted $\|f_\psi - \overline{f}_\psi\|$. The total error is denoted $\|f - \overline{f}_\psi\|$. From tables 3.1 and 3.2, it can clearly be seen that when using the

M	$\|f - f_\psi\|$	nz *b4* tol	tol	$\|f_\psi - \overline{f}_\psi\|$	$\|f - \overline{f}_\psi\|$	nz
5	3.53×10^{-1}	124	1×10^{-1}	1.46×10^{-1}	3.76×10^{-1}	14
6	6.15×10^{-2}	250	4×10^{-2}	5.81×10^{-2}	8.26×10^{-2}	21
7	1.19×10^{-2}	504	7×10^{-3}	9.93×10^{-3}	1.46×10^{-2}	33
8	5.84×10^{-3}	892	3×10^{-3}	4.81×10^{-3}	7.79×10^{-3}	37
9	1.50×10^{-3}	1537	6×10^{-4}	1.32×10^{-3}	1.88×10^{-3}	50
10	7.14×10^{-4}	2836	2×10^{-4}	4.71×10^{-4}	8.89×10^{-4}	60

Table 3.2: Approximation of $f(x)$ using the multiwavelet basis

multiwavelet basis significantly less terms are required to represent $f(x)$ than when the

scaling function basis is used.

Chapter Review

In this chapter we have reviewed the basic properties of wavelets. We have discussed the multiresolution analysis, where a scaling function basis $\mathbf{\Phi}_m = \{\phi_\lambda := 2^{\frac{m}{2}}\phi(2^m \cdot -l), |\lambda = \{m, l\}, l \in \Delta_m\}$ generates a sequence of nested subspaces of a Hilbert space H, such that

$$\ldots \subset V_{-1} \subset V_0 \subset V_1 \subset \ldots \subset H.$$

The difference space, W_m, between two nested subspaces V_m and V_{m+1} is spanned by the basis $\mathbf{\Psi}_m = \{\psi_\lambda := 2^{\frac{m}{2}}\psi(2^m \cdot -l), |\lambda = \{m, l\}, l \in \nabla_m\}$, where ψ is the *mother* wavelet.

We have shown that any space V_M, has two distinct bases, namely, the scaling function basis $\mathbf{\Phi}_M$ and the wavelet basis

$$\mathbf{\Psi}^M = \mathbf{\Phi}_0 \bigcup_{m=m_0}^{M-1} \mathbf{\Psi}_m.$$

Therefore, when approximating a function f in the space V_M we have two choices, either we can approximate f using the scaling function basis $\mathbf{\Psi}_M$,

$$f_M = \mathbf{\Phi}_M^T \boldsymbol{\alpha}_M,$$

or using the wavelet basis $\mathbf{\Psi}^M$,

$$f_M = \mathbf{\Phi}_0^T \boldsymbol{\alpha}_0 + \sum_{m=m_0}^{M-1} \mathbf{\Psi}_m^T \boldsymbol{\beta}_m.$$

Using the decomposition and reconstruction transformations, T_M and R_M, respectively, we are able to transform the scaling function approximation coefficients $\boldsymbol{\alpha}_M$ into the wavelet approximation coefficients $\begin{pmatrix} \boldsymbol{\beta}_{M-1} \\ \vdots \\ \boldsymbol{\beta}_0 \\ \boldsymbol{\alpha}_0 \end{pmatrix}$ and vice versa. Due to the vanishing moments property of wavelets, many of the coefficients $\boldsymbol{\beta}_m$ may be small and can therefore be discarded. This results in wavelets being widely used in applications ranging from image compression and data denoising to numerical analysis.

Following [10], we developed the basis functions we use in this thesis, namely, the multiwavelets. These have the advantage that an increase in the order of vanishing moments is obtained not by increasing the functions compact support, as with wavelets, but by increasing the number of *mother* wavelets.

Chapter 4

Multiwavelet Galerkin Methods

The approach we present in this chapter is in contrast to much of that in current use in the engineering community, where, because of the perceived and real complexity in directly using wavelet bases, many practitioners obtain the matrix compression offered by wavelets by adopting a two stage scheme. First, the standard boundary element matrix is computed using the scaling function bases $\mathbf{\Phi}_M$ for V_M^k. Then, a wavelet transform is applied to obtain the coefficient matrix with respect to the wavelet basis (3.2.8). The resulting matrix is then compressed by the application of a threshold, see [38]. Whilst this method results in some speed up of the solution time, its computational cost is still $\mathcal{O}(N^2)$. Here, we are interested in 'real' fast methods. By estimating the size of the matrix elements, we are able to decide a priori which elements are going to be too small to affect the accuracy of our approximation. This way we avoid computing them in the first place, resulting in a fast algorithm with computational cost $\mathcal{O}(N \log^p N)$. Whilst, here, we are only concerned with the Galerkin method, the collocation method with multiwavelets is considered in [39, 40].

In section 4.1 we consider the standard Galerkin method with respect to the multiwavelet

basis functions, whereas, in section 4.2 we introduce the so-called non-standard Galerkin method, see [6], with respect to the multiwavelet basis functions, [41, 42, 43]. For both methods we obtain bounds for the size of matrix elements. Using these bounds, compression strategies are developed. Finally complexity results are presented for both methods.

4.1 The Standard Galerkin Method

In this section we apply the Galerkin method using the multiwavelet basis $\mathbf{\Psi}^M$. We order the basis functions such that $\mathbf{\Psi}^M = \mathbf{\Psi}_{M-1} \cup \ldots \cup \mathbf{\Psi_1} \cup \mathbf{\Psi}_0 \cup \mathbf{\Phi}_0$. Therefore, the resulting coefficient matrix with elements $A_{\lambda,\lambda'} = \langle \mathcal{K}\psi_\lambda, \psi_{\lambda'} \rangle$ has the symmetric structure illustrated in Figure 4.1. Here, we see that the square diagonal blocks contain interactions between basis functions at the same resolution, whereas, off diagonal blocks contain interactions between multiwavelets at different resolutions.

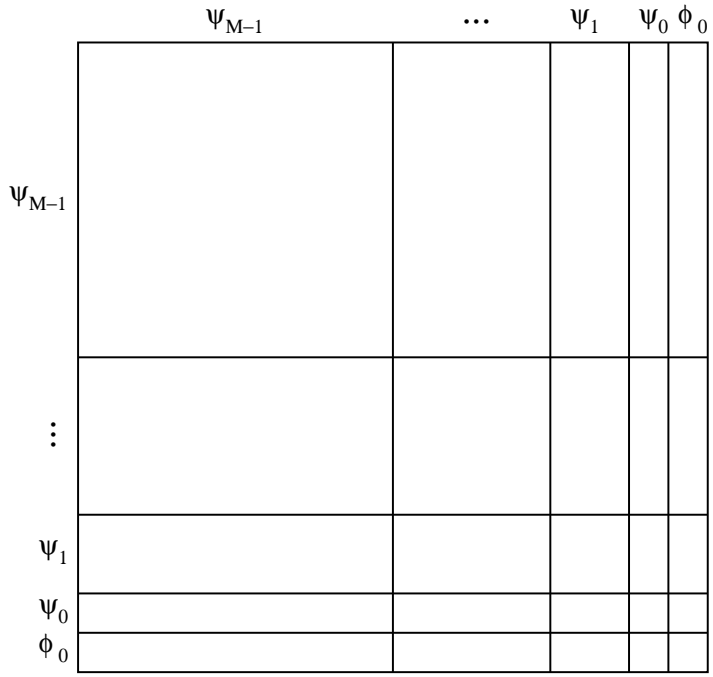


Figure 4.1: The standard multiwavelet Galerkin matrix

4.1.1 Matrix Element Bounds

We consider matrix elements of the form,

$$A_{\lambda,\lambda'} := \int_{I_{\lambda'}} \int_{I_\lambda} \overline{\mathcal{K}}(x,y)\psi_\lambda(y)\psi_{\lambda'}(x)dydx \tag{4.1.1}$$

where $\lambda = \{j, m, l\}$, with $j = 1, \ldots, k$; $m = M-1, \ldots, 0$ and $l = 0, \ldots, 2^m - 1$ and $\lambda' = \{j', m', l'\}$, with $j' = 1, \ldots, k$; $m' = M-1, \ldots, 0$ and $l' = 0, \ldots, 2^{m'} - 1$. I_λ is the support of the wavelet $\psi_\lambda(x)$ over the transformed boundary. Similarly, $I_{\lambda'}$ is the support of the wavelet $\psi_{\lambda'}$ over the transformed boundary. The kernel $\overline{\mathcal{K}}(x,y)$ is the so-called transformed kernel, when the integration domain is transformed from Γ to $[0,1]$. The kernels in many integral equations fall into the class of *analytically standard* functions.

Definition 4.1.1. The kernel $\mathcal{K}(\hat{x},\hat{y})$ is called *analytically standard* of order α if the transformed kernel $\overline{\mathcal{K}}(x,y)$ satisfies

$$\left|\partial_x^{\boldsymbol{l}}\partial_y^{\boldsymbol{m}}\overline{\mathcal{K}}(x,y)\right| \leq c\frac{(|\boldsymbol{l}|+|\boldsymbol{m}|)!}{dist\,(\hat{x},\hat{y})^{1+|\boldsymbol{l}|+|\boldsymbol{m}|+\alpha}}, \tag{4.1.2}$$

where

$$\overline{\mathcal{K}}(x,y) := \mathcal{K}\left(\kappa(x),\kappa(y)\right)|\kappa_x|\,|\kappa_y|\,, \tag{4.1.3}$$

with $\hat{x} := \kappa(x)$, $\hat{y} := \kappa(y)$ and $|\kappa_x|$, $|\kappa_y|$ are the Jacobians for the parametric map $\kappa : [0,1] \to \Gamma$, [44].

We now give a result bounding the size of the matrix elements $A_{\lambda,\lambda'}$.

Proposition 4.1.1. *Let $\mathcal{K}(\hat{x},\hat{y})$ be an analytically standard kernel. Then, the matrix elements $A_{\lambda,\lambda'}$, as described in (4.1.1), satisfy the bound*

$$|A_{\lambda,\lambda'}| \leq c\frac{2^{-(m+m')(k+\frac{1}{2})-2k}}{(2k+1)\;dist(\Gamma_{\lambda'},\Gamma_\lambda)^{1+2k+\alpha}}, \tag{4.1.4}$$

for $dist(\Gamma_{\lambda'}, \Gamma_\lambda) > 0$, where Γ_λ is the support of the multiwavelet ψ_λ on the boundary Γ.

Proof: First, we expand as a Taylor series our analytically standard kernel in (4.1.1), up to terms involving the $2k^{\text{th}}$ partial derivative of $\overline{\mathcal{K}}$, about the point (x_0, y_0), with x_0 taken as the midpoint of $I_{\lambda'}$ and y_0 as the midpoint of I_λ. Therefore, we obtain

$$\begin{aligned}\overline{\mathcal{K}}(x,y) =& \overline{\mathcal{K}}(x_0,y_0) + \left.\frac{\partial \overline{\mathcal{K}}}{\partial x}\right|_{(x_0,y_0)} (x-x_0) + \left.\frac{\partial \overline{\mathcal{K}}}{\partial y}\right|_{(x_0,y_0)} (y-y_0) \\ &+ \frac{1}{2} \sum_{i+j=2} \left.\frac{\partial^{i+j}\mathcal{K}}{\partial x^i \partial y^j}\right|_{(x_0,y_0)} (x-x_0)^i (y-y_0)^j + \dots \\ &+ \frac{1}{(2k-1)!} \sum_{i+j=2k-1} \left.\frac{\partial^{i+j}\overline{\mathcal{K}}}{\partial x^i \partial y^j}\right|_{(x_0,y_0)} (x-x_0)^i (y-y_0)^j + R, \end{aligned} \tag{4.1.5}$$

where R is the remainder of the truncated series. That is,

$$R = \frac{1}{(2k)!} \sum_{i+j=2k} \left.\frac{\partial^{i+j}\overline{\mathcal{K}}}{\partial x^i \partial y^j}\right|_{(t_x,t_y)} (x-x_0)^i (y-y_0)^j \ . \tag{4.1.6}$$

Therefore, substituting the Taylor expansion (4.1.5) into (4.1.1) and using the k vanishing moments property of the multiwavelets, in both x and y directions, we obtain

$$\begin{aligned}|A_{\lambda,\lambda'}| &= \frac{1}{(2k)!} \left| \int_{I_{\lambda'}} \int_{I_\lambda} \left(\left.\frac{\partial^{2k}\overline{\mathcal{K}}}{\partial x^k \partial y^k}\right|_{(t_x,t_y)} (x-x_0)^k (y-y_0)^k \right) \psi_\lambda(y) \psi_{\lambda'}(x) \ dydx \right| \\ &\leq \frac{1}{(2k)!} \left| \sup_{t_x \in I_{\lambda'}, t_y \in I_\lambda} \left.\frac{\partial^{2k}\overline{\mathcal{K}}}{\partial x^k \partial y^k}\right|_{(t_x,t_y)} \right| \times \\ &\qquad \left| \int_{I_{\lambda'}} \int_{I_\lambda} (x-x_0)^k (y-y_0)^k \psi_\lambda(y) \psi_{\lambda'}(x) \ dydx \right| . \end{aligned}$$

Then, using the Cauchy-Schwartz inequality yields,

$$|A_{\lambda,\lambda'}| \leq \frac{1}{(2k)!} \left| \sup_{t_x \in I_{\lambda'}, t_y \in I_\lambda} \frac{\partial^{2k} \overline{\mathcal{K}}}{\partial x^k \partial y^k} \right|_{(t_x,t_y)} \left| \left[\int_{I_{\lambda'}} (x - x_0)^{2k} \, dx \right]^{\frac{1}{2}} \times \left[\int_{I_\lambda} (y - y_0)^{2k} \, dy \right]^{\frac{1}{2}} \|\psi_\lambda\| \, \|\psi_{\lambda'}\| \, .$$

Now, we know that $\|\psi_\lambda\| = \|\psi_{\lambda'}\| = 1$ and

$$\begin{aligned} \int_{I_{\lambda'}} (x - x_0)^{2k} \, dx &= \frac{|I_{\lambda'}|^{2k+1}}{2^{2k}(2k+1)} = \frac{2^{-m'(2k+1)-2k}}{2k+1}, \\ \int_{I_\lambda} (y - y_0)^{2k} \, dy &= \frac{|I_\lambda|^{2k+1}}{2^{2k}(2k+1)} = \frac{2^{-m(2k+1)-2k}}{2k+1}. \end{aligned}$$

Therefore, we obtain

$$\begin{aligned} |A_{\lambda,\lambda'}| &\leq \frac{1}{(2k)!} \left| \sup_{t_x \in I_{\lambda'}, t_y \in I_\lambda} \frac{\partial^{2k} \overline{\mathcal{K}}}{\partial x^k \partial y^k} \right|_{(t_x,t_y)} \cdot \frac{2^{-\frac{1}{2}m(2k+1)-k}}{\sqrt{2k+1}} \cdot \frac{2^{-\frac{1}{2}m'(2k+1)-k}}{\sqrt{2k+1}} \\ &= \frac{1}{(2k)!} \left| \sup_{t_x \in I_{\lambda'}, t_y \in I_\lambda} \frac{\partial^{2k} \overline{\mathcal{K}}}{\partial x^k \partial y^k} \right|_{(t_x,t_y)} \cdot \frac{2^{-\frac{1}{2}(m+m')(2k+1)-2k}}{2k+1}. \end{aligned}$$

Since, the kernel $\mathcal{K}(\hat{x}, \hat{y})$ is analytically standard we use inequality (4.1.2) to obtain

$$\begin{aligned} |A_{\lambda,\lambda'}| &\leq \frac{1}{(2k)!} \cdot \frac{2^{-\frac{1}{2}(m+m')(2k+1)-2k}}{2k+1} . c \sup_{t_x \in I_{\lambda'}, t_y \in I_\lambda} \frac{(2k)!}{dist(\kappa(t_x), \kappa(t_y))^{1+2k+\alpha}} \\ &\leq c \frac{2^{-(m+m')(k+\frac{1}{2})-2k}}{(2k+1) \; dist(\Gamma_{\lambda'}, \Gamma_\lambda)^{1+2k+\alpha}}, \end{aligned}$$

which is the result we require. □

4.1.2 Compression Strategy

With the bound (4.1.4), we now follow [45] in developing a compression strategy for the multiwavelets. The coefficient matrix A_M is replaced by its sparse approximation A_M^d where

$$A^d_{\lambda,\lambda'} := \begin{cases} 0 & dist(\Gamma_{\lambda'}, \Gamma_\lambda) > d_{m,m'} \\ A_{\lambda,\lambda'} & \text{otherwise.} \end{cases} \tag{4.1.7}$$

That is, instead of solving the linear system

$$A_M \mathbf{u}_h = \mathbf{f}_h \tag{4.1.8}$$

we replace the matrix A_M by the matrix A_M^d and solve the modified linear system

$$A_M^d \mathbf{u}_h^d = \mathbf{f}_h. \tag{4.1.9}$$

Therefore, the error introduced by the modified system is $\left\|\mathbf{u}_h - \mathbf{u}_h^d\right\|_2$. We now find a bound for this error. Using equations (4.1.8) and (4.1.9) we obtain

$$\begin{aligned} A_M \mathbf{u}_h - A_M \mathbf{u}_h^d + A_M \mathbf{u}_h^d &= A_M^d \mathbf{u}_h^d \\ A_M \left(\mathbf{u}_h - \mathbf{u}_h^d\right) &= -\left(A_M - A_M^d\right) \mathbf{u}_h^d. \end{aligned}$$

Therefore, taking norms we obtain

$$\left\|\mathbf{u}_h - \mathbf{u}_h^d\right\|_2 \leq \left\|A_M^{-1}\right\|_2 \left\|A_M - A_M^d\right\|_2 \left\|\mathbf{u}_h^d\right\|_2.$$

Since A_M is a discretisation of $\mathcal{A} : H^s \rightarrow H^{s-\alpha}$ it can be shown that $\left\|A_M^{-1}\right\|_2 \leq c_1 \left(k2^M\right)^{|\alpha|}$, also we can show that $\|\mathbf{u}_h\|_2 \leq c_2$, we obtain

$$\left\|\mathbf{u}_h - \mathbf{u}_h^d\right\|_2 \leq d_1 \left(k2^M\right)^{|\alpha|} \left\|A_M - A_M^d\right\|_2,$$

where $d_1 = c_1 c_2$.

The values $d_{m,m'}$ are found so that the error in the solution with the modified matrix A_M^d, namely $\left\|\mathbf{u}_h - \mathbf{u}_h^d\right\|_2$, is of the same order as the discretisation error. We proceed by studying the norm $\left\|A_M - A_M^d\right\|_\infty$, since $\left\|A_M - A_M^d\right\|_2 \leq \sqrt{N} \left\|A_M - A_M^d\right\|_\infty$, see [46]. Therefore,

$$\left\|\mathbf{u}_h - \mathbf{u}_h^d\right\|_2 \leq d_1 \left(k2^M\right)^{|\alpha|+\frac{1}{2}} \left\|A_M - A_M^d\right\|_\infty. \tag{4.1.10}$$

Hence, in order to keep the error in the solution with modified matrix A_M^d, $\left\|\mathbf{u}_h - \mathbf{u}_h^d\right\|_2$, of the same order as the discretisation error, we find the values $d_{m,m'}$, such that the matrix $(A_M - A_M^d)$ satisfies,

$$\left\|A_M - A_M^d\right\|_\infty \leq \frac{\eta 2^{-M(|\alpha|+\frac{1}{2})}}{d_1 k^{|\alpha|+\frac{1}{2}}}, \tag{4.1.11}$$

where η is an estimate for the discretisation error. Theorem 2.4.4 states that for the Galerkin method, the discretisation error is of the same order as the approximation error by orthogonal projection. Moreover, Lemma 3.2.1 states that the approximation error when using multiwavelets to approximate the function u in the space V_M^k is bounded as

$$\left\|P_M^k u - u\right\|_{L_2} \leq 2^{-Mk} \frac{2}{4^k k!} \sup_{x\in[0,1]} \left|u^{(k)}(x)\right|.$$

Therefore, as an estimate for the discretisation error we use

$$\eta = \frac{d_2 2^{-Mk}}{4^k k!}. \tag{4.1.12}$$

We now proceed to find an estimate for $\left\|A_M - A_M^d\right\|_\infty$. Let us denote by $A^{mm'}$ the natural submatrices of A_M of size $k2^m$ by $k2^{m'}$ involving multiwavelet basis functions of W_m^k and $W_{m'}^k$. We similarly denote by $A^{mm',d}$ submatrices of A_M^d. We can now define the submatrices

$$\overline{A}^{mm'} = A^{mm'} - A^{mm',d}.$$

We bound the norm of the submatrices $\overline{A}^{mm'}$ and use this to find the values $d_{m,m'}$. Therefore, summing along each row of the submatrices $\overline{A}^{mm'}$ we obtain,

$$\begin{aligned}\sum_{\lambda'} \left|\overline{A}_{\lambda,\lambda'}\right| &= \sum_{\lambda' :\, dist(\Gamma_{\lambda'},\Gamma_\lambda)>d_{m,m'}} |A_{\lambda,\lambda'}| \\ &\leq \sum_{\lambda' :\, dist(\Gamma_{\lambda'},\Gamma_\lambda)>d_{m,m'}} \frac{c}{2k+1} 2^{-(m+m')(k+\frac{1}{2})-2k} dist(\Gamma_{\lambda'},\Gamma_\lambda)^{-(1+2k+\alpha)}.\end{aligned} \tag{4.1.13}$$

As we are bounding away from the diagonal it is reasonable to assume that $d_{m,m'} \geq \max\{2^{-m}, 2^{-m'}\}$. Therefore, we estimate the sum on the r.h.s of (4.1.13) by an integral to obtain,

$$\begin{aligned}\sum_{\lambda'} \left|\overline{A}_{\lambda,\lambda'}\right| &\leq \frac{c}{2k+1} 2^{-(m+m')(k+\frac{1}{2})-2k} . 2^{m'} . 2 \left| \int_{d_{m,m'}}^{\infty} |x|^{-(1+2k+\alpha)}\, dx \right| \\ &= \frac{c}{2k+1} 2^{-(m+m')(k+\frac{1}{2})-2k} 2^{m'+1} \frac{1}{2k+\alpha} d_{m,m'}^{-(2k+\alpha)}.\end{aligned}$$

Therefore, each submatrix $\overline{A}^{m,m'}$ satisfies,

$$\begin{aligned}\left\|\overline{A}^{mm'}\right\|_\infty &= \max_\lambda \sum_{\lambda'} \left|\overline{A}_{\lambda,\lambda'}\right| \\ &\leq \max_\lambda \left\{ \frac{c}{(2k+1)(2k+\alpha)} 2^{-(m+m')(k+\frac{1}{2})-2k} 2^{m'+1} d_{m,m'}^{-(2k+\alpha)} \right\} \\ &= \frac{c}{(2k+1)(2k+\alpha)} 2^{-(m+m')(k+\frac{1}{2})-2k} 2^{m'+1} d_{m,m'}^{-(2k+\alpha)}.\end{aligned} \tag{4.1.14}$$

Since we require, from (4.1.11) and (4.1.12), that

$$\left\|A_M - A_M^d\right\|_\infty \leq \max_{0\leq m\leq M-1} \sum_{m'=0}^{M-1} \left\|\overline{A}^{mm'}\right\|_\infty \leq \frac{d2^{-Mk}2^{-M(|\alpha|+\frac{1}{2})}}{4^k k! k^{|\alpha|+\frac{1}{2}}}, \tag{4.1.15}$$

we set,

$$\left\|\overline{A}^{mm'}\right\|_\infty \leq \frac{d2^{-Mk}2^{-M(|\alpha|+\frac{1}{2})}}{4^k k! k^{|\alpha|+\frac{1}{2}} M}, \tag{4.1.16}$$

where $d = \frac{d_2}{d_1}$. Then, the values $d_{m,m'}$ are found by equating the r.h.s. of inequalities (4.1.14) and (4.1.16),

$$d_{m,m'} = \left(\frac{cM4^k k! k^{\frac{1}{2}+|\alpha|}}{d(2k+1)(2k+\alpha)}\right)^{\frac{1}{2k+\alpha}} 2^{\frac{-2k(m+m')-2k+1+M\left(k+\frac{1}{2}+|\alpha|\right)}{2k+\alpha}}. \tag{4.1.17}$$

Theorem 4.1.1. *For $k > \alpha + |\alpha| + \frac{1}{2}$, the modified multiwavelet Galerkin matrix using the truncation value (4.1.17) has $\mathcal{O}(N \log N)$ non-zero elements.*

Proof: Since the structure of matrix A_M is symmetric along the diagonal, as shown in Figure 4.1, in this proof we consider only the upper block triangular part of matrix A_M, namely, the submatrices $A^{m,m'}$ where $m \geq m'$. Due to the truncation criterion (4.1.7), each row of the submatrix $A^{m,m'}$ contains at most $\mathcal{O}\left(2^{m'} d_{m,m'} + 1\right)$ non-zero elements.

Consider the submatrix $A^{(M-1)(M-1)}$. Each row of the submatrix contains at most $\mathcal{O}\left(2^{M-1} d_{M-1,M-1} + 1\right) = \mathcal{O}\left(M^{\frac{1}{2k+\alpha}} 2^{M-1} 2^{\frac{-M\left(3k-\frac{1}{2}-|\alpha|\right)}{2k+\alpha}} + 1\right) =$ $\mathcal{O}\left(M^{\frac{1}{2k+\alpha}} 2^{\frac{M(2k+\alpha)-M\left(3k-\frac{1}{2}-|\alpha|\right)}{2k+\alpha}} + 1\right) = \mathcal{O}\left(M^{\frac{1}{2k+\alpha}} 2^{\frac{-M\left(k-\frac{1}{2}-|\alpha|-\alpha\right)}{2k+\alpha}} + 1\right) =$ $\mathcal{O}\left(N^{\frac{-\left(k-\frac{1}{2}-|\alpha|-\alpha\right)}{2k+\alpha}} \log^{\frac{1}{2k+\alpha}} N + 1\right)$ non-zero elements. For $k > \alpha + |\alpha| + \frac{1}{2}$, $N^{\frac{-\left(k-\frac{1}{2}-|\alpha|\right)-\alpha}{2k+\alpha}}$ $\to 0$ as $N \to \infty$. Therefore, each row contains at most $\mathcal{O}(1)$ non-zero elements. The submatrix $A^{(M-1)(M-1)}$ has $\frac{N}{2}$ rows. Therefore, the submatrix contains at most $\mathcal{O}\left(\frac{N}{2}\right)$ non-zero elements.

We now consider the submatrix $A^{(M-1)(M-2)}$. Applying the truncation value $d_{M-1,M-2}$, each row of the submatrix $A^{(M-1)(M-2)}$ contains at most $\mathcal{O}\left(N^{\frac{-\left(k-\frac{1}{2}-|\alpha|-\alpha\right)}{2k+\alpha}}\log^{\frac{1}{2k+\alpha}}N+1\right)=\mathcal{O}(1)$ non-zero elements. The submatrix $A^{(M-1)(M-2)}$ has the same number of rows as the submatrix $A^{(M-1)(M-1)}$, namely, $\frac{N}{2}$ rows. Therefore, the submatrix $A^{(M-1)(M-2)}$ contains at most $\mathcal{O}\left(\frac{N}{2}\right)$ non-zero elements.

Using the same argument for the submatrices $A^{(M-1)m}$, with $m<M-2$, we infer that each row has asymptotically $\mathcal{O}(1)$ non-zero elements. Each of these submatrices has $\frac{N}{2}$ rows. Therefore, the submatrices $A^{(M-1)m}$, for $m<M-2$, contain at most $\mathcal{O}\left(\frac{N}{2}\right)$ non-zero elements. In Figure 4.1, we can see that there are $M=\log N-\log k$ submatrices $A^{(M-1)m}$, $m\leq M-1$. Therefore, the submatrix

$$\widehat{A}^{(M-1)}:=\begin{pmatrix} A^{(M-1)(M-1)} & \dots & A^{(M-1)0}\end{pmatrix},$$

contains at most $\mathcal{O}\left(\frac{N}{2}\log N\right)$ non-zero elements.

We now consider the submatrix $A^{(M-2)(M-2)}$. Using the truncation value $d_{M-2,M-2}$, each row of the submatrix $A^{(M-2)(M-2)}$ contains at most $\mathcal{O}\left(N^{\frac{-\left(k-\frac{1}{2}-|\alpha|-\alpha\right)}{2k+\alpha}}\log^{\frac{1}{2k+\alpha}}N+1\right)=\mathcal{O}(1)$ non-zero elements. The submatrix $A^{(M-2)(M-2)}$ has $\frac{N}{4}$ rows. Therefore, the submatrix contains at most $\mathcal{O}\left(\frac{N}{4}\right)$ non-zero elements. We now consider the submatrices $A^{(M-2)m}$ for $m<M-2$. Applying the same argument as above, each submatrix $A^{(M-2)m}$, for $m<M-2$, contains at most $\mathcal{O}\left(\frac{N}{4}\right)$ non-zero elements. In Figure 4.1, we can see that there are $M-1=\log N-\log k-1$ submatrices $A^{(M-2)m}$, with $m\leq M-2$. Therefore, the submatrix

$$\widehat{A}^{(M-2)}:=\begin{pmatrix} A^{(M-2)(M-2)} & \dots & A^{(M-2)0}\end{pmatrix},$$

contains at most $\mathcal{O}\left(\frac{N}{4}\log N\right)$ non-zero elements.

Applying the same argument to all levels, we infer that each submatrix $A^{mm'}$, for $m < m'$ contains at most $\mathcal{O}\left(\frac{N}{2^{M-m}}\right)$ non-zero elements. Then, each submatrix

$$\widehat{A}^m := \left(\begin{array}{ccc} A^{mm} & \cdots & A^{m0} \end{array} \right),$$

for $0 \leq m \leq M-1$, contains at most $\mathcal{O}\left(\frac{N}{2^{M-m}} \log N\right)$ non-zero elements. Therefore, summing over all submatrices $\widehat{A}^m$, for $0 \leq m \leq M-1$, the matrix A_M contains at most $\mathcal{O}\left(\frac{N}{2}\log N + \frac{N}{4}\log N + \ldots + \frac{N}{2^M}\log N\right) = \mathcal{O}\left(\left(\frac{1}{2}+\frac{1}{4}+\ldots+\frac{1}{2^M}\right) N \log N\right) = \mathcal{O}(N \log N)$ non-zero elements. □

4.2 The Non-Standard Galerkin Method

In this section, we use the so-called non-standard representation of an operator introduced in [6]. Here, the non-standard representation is used as a device to facilitate efficient matrix-vector multiplication in the solution of the classical Galerkin method. In the non-standard form all levels are decoupled, that is, we only have interactions between basis functions of the same resolution. However, the price we pay is that the non-standard representation is an over representation involving the multiwavelets and scaling functions on all levels.

We consider the projection operators $P_m^k : H^s[0,1] \rightarrow V_m^k$ and $Q_m^k : H^s[0,1] \rightarrow W_m^k$. The classical form of the Galerkin representation of a bounded linear operator $\mathcal{A} : H^s \rightarrow H^{s-\alpha}$ in the space V_M^k is the matrix $\mathrm{A}_M^c = P_M^k \mathcal{A} P_M^k$. Then, following [6] and using the

fact that $Q^k_{m-1} = P^k_m - P^k_{m-1}$, we rewrite $P^k_M \mathcal{A} P^k_M$ as a telescopic expansion,

$$\begin{aligned} P^k_M \mathcal{A} P^k_M &= \sum_{m=1}^{M} \left[P^k_m \mathcal{A} P^k_m - P^k_{m-1} \mathcal{A} P^k_{m-1} \right] + P^k_0 \mathcal{A} P^k_0 \\ &= \sum_{m=1}^{M} [\left(P^k_m - P^k_{m-1}\right) \mathcal{A} \left(P^k_m - P^k_{m-1}\right) + \left(P^k_m - P^k_{m-1}\right) \mathcal{A} P^k_{m-1} \\ &\qquad + P^k_{m-1} \mathcal{A} \left(P^k_m - P^k_{m-1}\right)] + P^k_0 \mathcal{A} P^k_0 \\ &= \sum_{m=1}^{M} \left[Q^k_{m-1} \mathcal{A} Q^k_{m-1} + Q^k_{m-1} \mathcal{A} P^k_{m-1} + P^k_{m-1} \mathcal{A} Q^k_{m-1} \right] + P^k_0 \mathcal{A} P^k_0, \end{aligned} \tag{4.2.1}$$

where

$$Q^k_m \mathcal{A} Q^k_m u = \sum_{\lambda,\lambda'} \beta^m_\lambda A^m_{\lambda,\lambda'} \psi_{\lambda'}(x), \tag{4.2.2}$$

$$Q^k_m \mathcal{A} P^k_m u = \sum_{\lambda,\lambda'} \alpha^m_\lambda B^m_{\lambda,\lambda'} \psi_{\lambda'}(x), \tag{4.2.3}$$

and

$$P^k_m \mathcal{A} Q^k_m u = \sum_{\lambda,\lambda'} \beta^m_\lambda C^m_{\lambda,\lambda'} \phi_{\lambda'}(x), \tag{4.2.4}$$

for $\lambda = \{j, m, l\}$, $\lambda' = \{j', m', l'\}$, $j, j' = 1, \ldots, k$; $m = m' = M-1, \ldots, 0$ and $l, l' = 0, \ldots, 2^m - 1$. The coefficients $\boldsymbol{\alpha}_m = \{\alpha^m_\lambda\}$ are the projection coefficients of a function u into the space V^k_m, likewise, the coefficients $\boldsymbol{\beta}_m = \{\beta^m_\lambda\}$ are the projection coefficients into the space W^k_m. That is,

$$P^k_m u = \sum_\lambda \langle u, \phi_\lambda \rangle \phi_\lambda = \sum_\lambda \alpha^m_\lambda \phi_\lambda$$

and

$$Q_m^k u = \sum_\lambda \langle u, \psi_\lambda \rangle \psi_\lambda = \sum_\lambda \beta_\lambda^m \psi_\lambda.$$

We define the coefficients $A^m_{\lambda,\lambda'}$, $B^m_{\lambda,\lambda'}$ and $C^m_{\lambda,\lambda'}$ as

$$A^m_{\lambda,\lambda'} := \langle \mathcal{A}\psi_\lambda, \psi_{\lambda'} \rangle, \tag{4.2.5}$$

$$B^m_{\lambda,\lambda'} := \langle \mathcal{A}\phi_\lambda, \psi_{\lambda'} \rangle, \tag{4.2.6}$$

and

$$C^m_{\lambda,\lambda'} := \langle \mathcal{A}\psi_\lambda, \phi_{\lambda'} \rangle, \tag{4.2.7}$$

where the superscript m, signifies that $m = m'$. Note that in section 4.1 submatrices are denoted by $A^{mm'}$ but here we only have submatrices where $m = m'$, therefore we denote them by A^m. We now, define the submatrices A^m of size $k2^m$ by $k2^m$ as

$$A^m = \begin{pmatrix} A^m_{1,m,0;1,m,0} & \cdots & A^m_{k,m,0;1,m,0} & \cdots & A^m_{1,m,2^m-1;1,m,0} & \cdots & A^m_{k,m,2^m-1;1,m,0} \\ \vdots & & \vdots & & \vdots & & \vdots \\ A^m_{1,m,0;k,m,0} & \cdots & A^m_{k,m,0;k,m,0} & \cdots & A^m_{1,m,2^m-1;k,m,0} & \cdots & A^m_{k,m,2^m-1;k,m,0} \\ \vdots & & \vdots & & \vdots & & \vdots \\ A^m_{1,m,0;1,m,2^m-1} & \cdots & A^m_{k,m,0;1,m,2^m-1} & \cdots & A^m_{1,m,2^m-1;1,m,2^m-1} & \cdots & A^m_{k,m,2^m-1;1,m,2^m-1} \\ \vdots & & \vdots & & \vdots & & \vdots \\ A^m_{1,m,0;k,m,2^m-1} & \cdots & A^m_{k,m,0;k,m,2^m-1} & \cdots & A^m_{1,m,2^m-1;k,m,2^m-1} & \cdots & A^m_{k,m,2^m-1;k,m,2^m-1} \end{pmatrix}.$$

The submatrices B^m and C^m are similarly defined. Furthermore, let $D^m_{\lambda,\lambda'} := \langle \mathcal{A}\phi_\lambda, \phi_{\lambda'} \rangle$. Then, the matrix representation of the operator $P_0^k \mathcal{A} P_0^k$ can be defined analogously to A^0

and we denote it by D^0.

We are now able to present the matrix representation of the telescopic expansion, namely the non-standard matrix. The non-standard matrix K_M, of size $2k(2^M-1)$ by $2k(2^M-1)$, is defined as

$$K_M := \begin{pmatrix} A^{M-1} & B^{M-1} & & & & & \\ C^{M-1} & & & & & & \\ & & A^{M-2} & B^{M-2} & & & \\ & & C^{M-2} & & & & \\ & & & & \ddots & & \\ & & & & & A^0 & B^0 \\ & & & & & C^0 & D^0 \end{pmatrix}. \tag{4.2.8}$$

Therefore, when using the non-standard matrix, rather than the standard matrix of section 4.1, we only have interactions between basis functions of the same resolution. That is, the matrix only involves the matrix blocks A^m and not the off diagonal blocks $A^{m,m'}$ of the standard matrix. Consequently, the domains of integration for the elements of the non-standard matrix are the same size.

In the non-standard method, the non-standard matrix (4.2.8) is used to efficiently compute the matrix-vector product $\mathbf{b}_M = \mathrm{A}_M^c\mathbf{x}$, where A_M^c is the classical Galerkin matrix, that is the coefficient matrix with respect to the scaling function basis. The elements of the vector $\mathbf{x}$ are with respect to the scaling function basis $\mathbf{\Phi}_M$. Hence, the vector $\mathbf{x}$ is of length $k2^M$. Therefore, the vector $\mathbf{x}$ needs to be transformed to the vector $\mathbf{y}$, of length $2k(2^M-1)$, whose elements are with respect to the over representation $\{\mathbf{\Psi}_{M-1} \cup \mathbf{\Phi}_{M-1} \cup \ldots \cup \mathbf{\Psi}_1 \cup \mathbf{\Phi}_1 \cup \mathbf{\Psi}_0 \cup \mathbf{\Phi}_0\}$.

The decomposition transformation T_m of section 3.1.3 transforms the vector $\boldsymbol{\alpha}_M$, of

length $k2^M$, to the vector $\boldsymbol{\beta}_{0,M-1}$, of length $k2^M$. That is,

$$T_M \boldsymbol{\alpha}_M = \boldsymbol{\beta}_{0,M-1},$$

where

$$\boldsymbol{\beta}_{0,M-1} = \begin{pmatrix} \boldsymbol{\beta}_{M-1} \\ \vdots \\ \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_0 \\ \boldsymbol{\alpha}_0 \end{pmatrix}.$$

Therefore, when using the non-standard method the decomposition transformation T_M cannot be applied. Instead we introduce the decomposition transformation $\overline{T}_M$. For $0 \leq m < M-1$, we define $[2k(2^M-1)-2k(2^{m+1}-1)] \times [2k(2^m-1)-2k(2^{m+1}-1)+k2^{m+1}]$ matrix

$$\overline{T}_{M,m} = \begin{pmatrix} \mathrm{I}_1 & 0 \\ 0 & \mathrm{I}_2 \\ 0 & \mathrm{G}_m \end{pmatrix}, \tag{4.2.9}$$

where I_1 is the identity matrix of size $2k(2^M - 1) - 2k(2^{m+1} - 1) - k2^{m+1}$ and I_2 is the identity matrix of size $k2^{m+1}$ and G_m is the matrix (3.1.28). For $m = M - 1$, $\overline{T}_{M,M-1} = T_{M,M-1}$. Then, the decomposition transformation $\overline{T}_M$ is defined as,

$$\overline{T}_M = \overline{T}_{M,0} \cdots \overline{T}_{M,M-1}.$$

The decomposition transformation $\overline{T}_M$ acts on the vector $\boldsymbol{\alpha}_M$, of length $k2^m$, and results

in the vector

$$\begin{pmatrix} \boldsymbol{\beta}_{M-1} \\ \boldsymbol{\alpha}_{M-1} \\ \vdots \\ \boldsymbol{\beta}_1 \\ \boldsymbol{\alpha}_1 \\ \boldsymbol{\beta}_0 \\ \boldsymbol{\alpha}_0 \end{pmatrix}, \tag{4.2.10}$$

of length $2k(2^M - 1)$.

We also require the reverse transformation, the reconstruction transformation $\overline{R}_M$. For $0 \leq m < M-1$, we define the $[2k(2^m - 1) - 2k(2^{m+1} - 1) + k2^{m+1}] \times [2k(2^M - 1) - 2k(2^{m+1} - 1)]$ matrix

$$\overline{R}_{M,m} = \begin{pmatrix} \mathrm{I}_1 & 0 & 0 \\ 0 & \mathrm{I}_2 & \mathrm{C}_m \end{pmatrix}, \tag{4.2.11}$$

where I_1 and I_2 are the identity matrices of size $2k(2^M - 1) - 2k(2^{m+1} - 1) - k2^{m+1}$ and $k2^{m+1}$, respectively, and C_m is the matrix (3.1.25). For $m = M-1$, $\overline{R}_{M,M-1} = R_{M,M-1}$. Then, the reconstruction transformation $\overline{R}_M$ is defined as,

$$\overline{R}_M = \overline{R}_{M,M-1} \cdots \overline{R}_{M,0}.$$

Using the decomposition and reconstruction transformations $\overline{T}_M$ and $\overline{R}_M$, respectively, the non-standard matrix can be used to efficiently compute the matrix-vector product $\mathbf{b}_M = A_M^c \mathbf{x}$, as follows:

1. The decomposition transformation $\overline{T}_M$ is applied to the vector **x**, of length $k2^M$. This results in a vector **y**, of length $2k(2^M - 1)$, whose elements are with respect to

the over representation $\{\mathbf{\Psi}_{M-1} \cup \mathbf{\Phi}_{M-1} \cup \ldots \cup \mathbf{\Psi}_1 \cup \mathbf{\Phi}_1 \cup \mathbf{\Psi}_0 \cup \mathbf{\Phi}_0\}$. That is,

$$\overline{T}_m \mathbf{x} = \mathbf{y}.$$

Note that in practice the matrices $\overline{T}_{M,m}$ are not formed. Instead, the decomposition filters (3.2.26) and (3.2.27) are applied; when the non-standard method is used, coefficients with respect to the bases $\mathbf{\Phi}_m$ are stored. However, when the filters (3.2.26) and (3.2.27) are used to apply T_M in the standard wavelet case, the coefficients with respect to the basis $\mathbf{\Phi}_m$ are not stored.

2. Then, the vector **y** is premultiplied by the non-standard matrix K_M, to obtain the vector **z**,

$$K_M \mathbf{y} = \mathbf{z}.$$

3. The reconstruction transformation $\overline{R}_M$ is applied to the vector **z**, of length $2k(2^M - 1)$, to obtain the required vector $\mathbf{b}_M = A_M^c \mathbf{x}$ of length $k2^M$. That is,

$$\overline{R}_M \mathbf{z} = \mathbf{b}_M.$$

Note that in practice the matrices $\overline{R}_{M,m}$ are not formed. Instead, we apply the filter (3.2.23) as follows. Consider the vector

$$\mathbf{z} = \begin{pmatrix} A^{M-1}\boldsymbol{\beta}_{M-1} + B^{M-1}\boldsymbol{\alpha}_{M-1} \\ C^{M-1}\boldsymbol{\beta}_{M-1} \\ \vdots \\ A^1\boldsymbol{\beta}_1 + B^1\boldsymbol{\alpha}_1 \\ C^1\boldsymbol{\beta}_1 \\ A^0\boldsymbol{\beta}_0 + B^0\boldsymbol{\alpha}_0 \\ C^0\boldsymbol{\beta}_0 + D^0\boldsymbol{\alpha}_0 \end{pmatrix}, \tag{4.2.12}$$

which is the vector (4.2.10) premultiplied by the non-standard matrix K_M. We note that $D^m\boldsymbol{\alpha}_m$ is only present for $m = 0$. By applying the reconstruction filter (3.2.23) we reconstruct $D^m\boldsymbol{\alpha}_m$, for $m > 0$. We start by applying the filter (3.2.23) to the vector

$$\begin{pmatrix} A^0\boldsymbol{\beta}_0 + B^0\boldsymbol{\alpha}_0 \\ C^0\boldsymbol{\beta}_0 + D^0\boldsymbol{\alpha}_0 \end{pmatrix},$$

to obtain the vector $\mathbf{b}_1$, of length $k2^1$. That is, for $j' = 1, \ldots, k$ and $l = 0, 1$,

$$\mathsf{b}_{\{j',1,2l\}} = \sum_{j=1}^{k} c_{j,j'} \left(\left[C^0\boldsymbol{\beta}_0\right]_{\{j,0,l\}} + \left[D^0\boldsymbol{\alpha}_0\right]_{\{j,0,l\}} \right) + d_{j,j'} \left(\left[A^0\boldsymbol{\beta}_0\right]_{\{j,0,l\}} + \left[B^0\boldsymbol{\alpha}_0\right]_{\{j,0,l\}} \right),$$

$$\mathsf{b}_{\{j',1,2l+1\}} = \sum_{j=1}^{k} c_{j,j'+k} \left(\left[C^0\boldsymbol{\beta}_0\right]_{\{j,0,l\}} + \left[D^0\boldsymbol{\alpha}_0\right]_{\{j,0,l\}} \right) + d_{j,j'+k} \left(\left[A^0\boldsymbol{\beta}_0\right]_{\{j,0,l\}} + \left[B^0\boldsymbol{\alpha}_0\right]_{\{j,0,l\}} \right).$$

The vector $\mathbf{b}_1$ is then added to the vector $C^1\boldsymbol{\beta}_1$. Therefore, we obtain the vector

$$\begin{pmatrix} A^{M-1}\boldsymbol{\beta}_{M-1} + B^{M-1}\boldsymbol{\alpha}_{M-1} \\ C^{M-1}\boldsymbol{\beta}_{M-1} \\ \vdots \\ A^1\boldsymbol{\beta}_1 + B^1\boldsymbol{\alpha}_1 \\ C^1\boldsymbol{\beta}_1 + \mathbf{b}_1 \end{pmatrix},$$

of length $2k(2^M - 1) - 2k$. The filter (3.2.23) is now applied to the vector

$$\begin{pmatrix} A^1\boldsymbol{\beta}_1 + B^1\boldsymbol{\alpha}_1 \\ C^1\boldsymbol{\beta}_1 + \mathbf{b}_1 \end{pmatrix},$$

and we obtain the vector $\mathbf{b}_2$, of length $k2^2$. That is, for $j' = 1, \ldots, k$ and $l = 0, 1, 2, 3$,

$$\mathrm{b}_{\{j',2,2l\}} = \sum_{j=1}^{k} c_{j,j'} \left(\left[C^1 \boldsymbol{\beta}_1 \right]_{\{j,1,l\}} + [\mathbf{b}_1]_{\{j,1,l\}} \right) + d_{j,j'} \left(\left[A^1 \boldsymbol{\beta}_1 \right]_{\{j,1,l\}} + \left[B^1 \boldsymbol{\alpha}_1 \right]_{\{j,1,l\}} \right),$$

$$\mathrm{b}_{\{j',2,2l+1\}} = \sum_{j=1}^{k} c_{j,j'+k} \left(\left[C^1 \boldsymbol{\beta}_1 \right]_{\{j,1,l\}} + [\mathbf{b}_1]_{\{j,1,l\}} \right) + d_{j,j'+k} \left(\left[A^1 \boldsymbol{\beta}_1 \right]_{\{j,1,l\}} + \left[B^1 \boldsymbol{\alpha}_1 \right]_{\{j,1,l\}} \right).$$

The vector $\mathbf{b}_2$ is then added to the vector $C^2 \boldsymbol{\beta}_2$. Therefore, we obtain the vector

$$\begin{pmatrix} A^{M-1} \boldsymbol{\beta}_{M-1} + B^{M-1} \boldsymbol{\alpha}_{M-1} \\ C^{M-1} \boldsymbol{\beta}_{M-1} \\ \vdots \\ A^2 \boldsymbol{\beta}_2 + B^2 \boldsymbol{\alpha}_2 \\ C^2 \boldsymbol{\beta}_2 + \mathbf{b}_2 \end{pmatrix},$$

of length $2k(2^M - 1) - 6k$. This is repeated for each level, until the vector $\mathbf{b}_M$, of length $k2^M$, is recovered.

Therefore, if we wish to solve the classical Galerkin system

$$\mathrm{A}_M^c \mathbf{u}_h = \mathbf{f}_h,$$

using an iterative solver, rather than forming the full matrix A_M^c we form the sparse non-standard matrix K_M. Then, for iteration i, the vector $\mathbf{b}^{(i)} = A_M^c \mathbf{u}_h^{(i)}$ can be computed as $\mathbf{b}^{(i)} = \overline{R}_M K_M \overline{T}_M \mathbf{u}_h^{(i)}$.

4.2.1 Matrix Element Bounds

We consider the matrix elements $A^m_{\lambda,\lambda'}$, $B^m_{\lambda,\lambda'}$ and $C^m_{\lambda,\lambda'}$ of (4.2.5), (4.2.6) and (4.2.7), respectively, where the operator $\mathcal{A}$ has the form

$$(\mathcal{A}u)(x) = \int_0^1 \overline{\mathcal{K}}(x,y)u(y)\,dy. \tag{4.2.13}$$

Then we have the following result.

Proposition 4.2.1. *Let $\mathcal{K}(\hat{x},\hat{y})$ be an analytically standard kernel. Then, the matrix elements $A^m_{\lambda,\lambda'}$ in (4.2.5) satisfy the bound*

$$\left|A^m_{\lambda,\lambda'}\right| \leq c\frac{2^{-m(2k+1)-2k}}{(2k+1)\;dist(\Gamma_{\lambda'},\Gamma_\lambda)^{1+2k+\alpha}}, \tag{4.2.14}$$

for $dist(\Gamma_{\lambda'},\Gamma_\lambda) > 0$. Moreover, the matrix elements $B^m_{\lambda,\lambda'}$ and $C^m_{\lambda,\lambda'}$ of (4.2.6) and (4.2.7), respectively, satisfy the bound

$$\left|B^m_{\lambda,\lambda'}\right|, \left|C^m_{\lambda,\lambda'}\right| \leq c\frac{2^{-m(k+\frac{1}{2})-k}}{\sqrt{2k+1}\;dist\,(\Gamma_{\lambda'},\Gamma_\lambda)^{1+k+\alpha}}, \tag{4.2.15}$$

for $dist(\Gamma_{\lambda'},\Gamma_\lambda) > 0$.

Proof: The bound (4.2.14) follows from proposition 4.1.1 by letting $m = m'$. To obtain the bound (4.2.15) for

$$B^m_{\lambda,\lambda} = \int_{I_{\lambda'}}\int_{I_{\lambda'}} \overline{\mathcal{K}}(x,y)\phi_\lambda(y)\psi_{\lambda'}(x)\;dydx \tag{4.2.16}$$

we rewrite the kernel $\overline{\mathcal{K}}(x,y)$ as a $(k+1)$-term Taylor expansion about the point (x_0,y),

with x_0 taken as the mid-point of $I_{\lambda'}$. Therefore, we obtain

$$\overline{\mathcal{K}}(x,y) = \overline{\mathcal{K}}(x_0,y) + \left.\frac{\partial\overline{\mathcal{K}}}{\partial x}\right|_{(x_0,y)} + \frac{1}{2!}\left.\frac{\partial^2\overline{\mathcal{K}}}{\partial x^2}\right|_{(x_0,y)}(x-x_0) + \ldots + \frac{1}{(k-1)!}\left.\frac{\partial^{k-1}\overline{\mathcal{K}}}{\partial x^{k-1}}\right|_{(x_0,y)}(x-x_0)^{k-1} + R \quad (4.2.17)$$

where R is the remainder of the truncated series. That is,

$$R = \frac{1}{k!}\left.\frac{\partial^k\overline{\mathcal{K}}}{\partial x^k}\right|_{(t_x,y)}(x-x_0)^k. \quad (4.2.18)$$

Therefore, substituting the Taylor expansion (4.2.17) into (4.2.16) and using the k vanishing moments property of the multiwavelets we obtain

$$\begin{aligned}|B_{\lambda,\lambda'}| &= \frac{1}{k!}\left|\int_{I_{\lambda'}}\int_{I_\lambda}\left.\frac{\partial^k\overline{\mathcal{K}}}{\partial x^k}\right|_{(t_x,y)}(x-x_0)^k\phi_{\lambda'}(y)\psi_\lambda(x)\,dydx\right| \\ &\leq \frac{1}{k!}\left|\sup_{t_x\in I_{\lambda'},y\in I_\lambda}\left.\frac{\partial^k\overline{\mathcal{K}}}{\partial x^k}\right|_{(t_x,y)}\right|\left|\int_{I_{\lambda'}}\int_{I_\lambda}(x-x_0)^k\phi_\lambda(y)\psi_{\lambda'}(x)\,dydx\right|.\end{aligned}$$

Then, using the Cauchy-Schwartz inequality yields,

$$\begin{aligned}|B_{\lambda,\lambda'}| &\leq \frac{1}{k!}\left|\sup_{t_x\in I_{\lambda'},y\in I_\lambda}\left.\frac{\partial^k\overline{\mathcal{K}}}{\partial x^k}\right|_{(t_x,y)}\right|\left[\int_{I_\lambda}dy\right]^{\frac{1}{2}}\left[\int_{I_{\lambda'}}(x-x_0)^{2k}dx\right]^{\frac{1}{2}}\|\phi_\lambda\|\,\|\psi_{\lambda'}\| \\ &\leq \frac{1}{k!}\left|\sup_{t_x\in I_{\lambda'},y\in I_\lambda}\left.\frac{\partial^k\overline{\mathcal{K}}}{\partial x^k}\right|_{(t_x,y)}\right|\frac{2^{-\frac{1}{2}m'(2k+1)-k}}{\sqrt{2k+1}}.2^{-\frac{1}{2}m},\end{aligned}$$

where we have used the facts that $\|\phi_\lambda\| = \|\psi_{\lambda'}\| = 1$ and

$$\int_{I_{\lambda'}}(x-x_0)^{2k}\,dx = \frac{|I_{\lambda'}|^{2k+1}}{2^{2k}(2k+1)} = \frac{2^{-m(2k+1)-2k}}{2k+1}.$$

Since, the kernel $\mathcal{K}(\hat{x}, \hat{y})$ is analytically standard we use inequality (4.1.2) to obtain

$$\begin{aligned}\left|B^m_{\lambda,\lambda'}\right| &\leq \frac{1}{k!}\frac{2^{-m(k+\frac{1}{2})-k}}{\sqrt{2k+1}}.c \sup_{t_x\in I_{\lambda'}, y\in I_\lambda} \frac{(k)!}{dist(\kappa(t_x),\kappa(y))^{1+k+\alpha}} \\ &\leq c\frac{2^{-m(k+\frac{1}{2})-k}}{\sqrt{2k+1}dist(\Gamma_{\lambda'},\Gamma_\lambda)^{1+k+\alpha}},\end{aligned}$$

which is the result we require. To prove the bound for matrix elements $C^m_{\lambda,\lambda'}$ we proceed as above for elements of $B^m_{\lambda,\lambda}$, except we consider a Taylor expansion about the point (x, y_0), with y_0 the mid-point of I_λ. □

4.2.2 Compression Strategy

With the bounds (4.2.14) and (4.2.15) we now develop a compression strategy for the non-standard Galerkin method with multiwavelet basis functions. The non-standard matrix K_M is replaced by its sparse approximation $K^{d,d'}_M$. That is, the submatrices A^m, B^m and C^m are replaced by the sparse submatrices $A^{m,d}$, $B^{m,d'}$ and $C^{m,d'}$ where

$$A^{m,d}_{\lambda,\lambda'} := \begin{cases} 0 & dist(\Gamma_{\lambda'},\Gamma_\lambda) > d_m \\ A^m_{\lambda,\lambda'} & \text{otherwise,}\end{cases} \tag{4.2.19}$$

$$B^{m,d'}_{\lambda,\lambda'} := \begin{cases} 0 & dist(\Gamma_{\lambda'},\Gamma_\lambda) > d'_m \\ B^m_{\lambda,\lambda'} & \text{otherwise,}\end{cases} \tag{4.2.20}$$

and

$$C^{m,d'}_{\lambda,\lambda'} := \begin{cases} 0 & dist(\Gamma_{\lambda'},\Gamma_\lambda) > d'_m \\ C^m_{\lambda,\lambda'} & \text{otherwise.}\end{cases} \tag{4.2.21}$$

The values d_m and d'_m are found so that the error in the solution with the modified matrix $K_M^{d,d'}$ is of the same order as the discretisation error.

The submatrices $\overline{A}^m$, $\overline{B}^m$ and $\overline{C}^m$ are defined as

$$\overline{A}^m := A^m - A^{m,d},$$
$$\overline{B}^m := B^m - B^{m,d'},$$

and

$$\overline{C}^m := C^m - C^{m,d'}.$$

We now bound the norm of the submatrices $\overline{A}^m$, $\overline{B}^m$ and $\overline{C}^m$. These are then used to find the values d_m and d'_m. Using the bound (4.1.14) with $m = m'$, we see that

$$\left\|\overline{A}^m\right\|_\infty \leq \frac{c}{(2k+1)(2k+\alpha)} 2^{-2k(m+1)+1} d_m^{-(2k+\alpha)}. \tag{4.2.22}$$

We now find a bound for $\left\|\overline{B}^m\right\|_\infty$,

$$\begin{aligned}\left\|\overline{B}^m\right\|_\infty &\leq \max_\lambda \left\{ \sum_{\lambda' :\, dist(\Gamma_{\lambda'},\Gamma_\lambda)>d'_m} B^m_{\lambda,\lambda'} \right\} \\ &\leq \max_\lambda \left\{ \sum_{\lambda' :\, dist(\Gamma_{\lambda'},\Gamma_\lambda)>d'_m} \frac{c\, 2^{-m(k+\frac{1}{2})-k}}{\sqrt{2k+1}}\, dist(\Gamma_{\lambda'},\Gamma_\lambda)^{-(1+k+\alpha)} \right\}. \end{aligned} \tag{4.2.23}$$

As in section 4.1.2, it is reasonable to assume that $d'_m \geq 2^{-m}$. Therefore, we estimate the

sum in (4.2.23) by an appropriate integral to obtain

$$\begin{aligned}\left\|\overline{B}^m\right\|_\infty &\leq \max_\lambda \left\{ \frac{c\ 2^{-m(k+\frac{1}{2})-k}}{\sqrt{2k+1}} 2^m 2. \left| \int_{d'_m}^{\infty} |x|^{-(1+k+\alpha)}\ dx \right| \right\} \\ &= \max_\lambda \left\{ \frac{c\ 2^{-k(m+1)+\frac{1}{2}m+1}}{\sqrt{2k+1}(k+\alpha)}\ {d'}_m^{-(k+\alpha)} \right\} \\ &= \frac{c\ 2^{-k(m+1)+\frac{1}{2}m+1}}{\sqrt{2k+1}(k+\alpha)}\ {d'}_m^{-(k+\alpha)}. \end{aligned} \tag{4.2.24}$$

Similarly, we obtain

$$\left\|\overline{C}^m\right\|_\infty \leq \frac{c\ 2^{-k(m+1)+\frac{1}{2}m+1}}{\sqrt{2k+1}(k+\alpha)}\ {d'}_m^{-(k+\alpha)}. \tag{4.2.25}$$

We now consider the larger submatrices

$$E^m := \begin{pmatrix} \overline{A}^m & \overline{B}^m \\ \overline{C}^m & 0 \end{pmatrix}, \tag{4.2.26}$$

which satisfy

$$\|E^m\|_\infty \leq \left\|\overline{A}^m\right\|_\infty + \left\|\overline{B}^m\right\|_\infty. \tag{4.2.27}$$

Therefore, the difference between the full non-standard matrix K_M and its sparse approximation can be written as

$$K_M - K_M^{d,d'} = \begin{pmatrix} E^{M-1} & & & & \\ & E^{M-2} & & & \\ & & \ddots & & \\ & & & E^1 & \\ & & & & E^0 \end{pmatrix}. \tag{4.2.28}$$

In order to keep the error in the solution with the modified non-standard matrix $K_M^{d,d'}$ to

be of the same order as the discretisation error, we require

$$\left\| K_M - K_M^{d,d'} \right\|_\infty \leq \frac{d2^{-Mk}2^{M(|\alpha|+\frac{1}{2})}}{M4^k k! k^{\frac{1}{2}-\alpha}},$$

where $d = \frac{d_2}{d_1}$. Since

$$\left\| K_M - K_M^{d,d'} \right\|_\infty = \max_{0 \leq m \leq M-1} \left\| E^m \right\|_\infty, \tag{4.2.29}$$

using relationship (4.2.27), we set,

$$\left\| E^m \right\|_\infty \leq \left\| \overline{A}^m \right\|_\infty + \left\| \overline{B}^m \right\|_\infty \leq \frac{d2^{-Mk}2^{M(|\alpha|+\frac{1}{2})}}{M4^k k! k^{\frac{1}{2}-\alpha}}. \tag{4.2.30}$$

Therefore, to find the values d_m and d'_m we set,

$$\left\| \overline{A}^m \right\|_\infty \leq \frac{d2^{-Mk}2^{M(|\alpha|+\frac{1}{2})}}{2M4^k k! k^{\frac{1}{2}-\alpha}} \tag{4.2.31}$$

and

$$\left\| \overline{B}^m \right\|_\infty \leq \frac{d2^{-Mk}2^{M(|\alpha|+\frac{1}{2})}}{2M4^k k! k^{\frac{1}{2}-\alpha}}. \tag{4.2.32}$$

Then, equating the r.h.s of (4.2.22) and (4.2.31) we obtain,

$$d_m = \left(\frac{2c4^k k! M k^{\frac{1}{2}+|\alpha|}}{d(2k+1)(2k+\alpha)} \right)^{\frac{1}{2k+\alpha}} 2^{\frac{-2k(m+1)+1+M(k+\frac{1}{2}+|\alpha|)}{2k+\alpha}}. \tag{4.2.33}$$

Similarly, the r.h.s of (4.2.24) and (4.2.32) we obtain,

$$d'_m = \left(\frac{2c4^k k! M k^{\frac{1}{2}+|\alpha|}}{d(2k+1)(k+\alpha)} \right)^{\frac{1}{k+\alpha}} 2^{\frac{-k(m+1)+\frac{1}{2}m+1+M(k+\frac{1}{2}+|\alpha|)}{k+\alpha}}. \tag{4.2.34}$$

Theorem 4.2.1. *Let $|\alpha| > k - \frac{1}{2}$. Then, the modified multiwavelet non-standard matrix using the truncation values (4.2.33) and (4.2.34) has $\mathcal{O}\left(N^{1+\frac{1+|\alpha|}{k+\alpha}} \log^{\frac{2}{k+\alpha}} N\right)$ non-zero elements, where $N = k2^M$.*

Proof:

First we consider the submatrix E^{M-1}. Applying the truncation value d_{M-1} to the submatrix A^{M-1}, each row contains at most $\mathcal{O}\left(M^{\frac{1}{2k+\alpha}} 2^{\frac{-M\left(k-\frac{1}{2}-|\alpha|\right)}{2k+\alpha}} + 1\right) = \mathcal{O}\left(N^{\frac{-\left(k-\frac{1}{2}-|\alpha|\right)}{2k+\alpha}} \log^{\frac{1}{2k+\alpha}} N + 1\right)$ non-zero elements. For $|\alpha| < k - \frac{1}{2}$, $N^{\frac{-\left(k-\frac{1}{2}+|\alpha|\right)}{2k+\alpha}} \to 0$ as $N \to \infty$. Therefore, each row contains at most $\mathcal{O}(1)$ non-zero elements. The submatrix A^{M-1} has $\frac{N}{2}$ rows. Therefore, the submatrix contains at most $\mathcal{O}\left(\frac{N}{2}\right)$ non-zero elements. Applying the truncation value d'_{M-1} to the submatrices B^{M-1} and C^{M-1}, each row contains at most $\mathcal{O}\left(N^{\frac{1+|\alpha|}{k+\alpha}} \log^{\frac{1}{k+\alpha}} N\right)$ non-zero elements. Since the submatrices B^{M-1} and C^{M-1} both have $\frac{N}{2}$ rows, the submatrices B^{M-1} and C^{M-1} contain at most $\mathcal{O}\left(\frac{N^{1+\frac{1+|\alpha|}{k+\alpha}}}{2} \log^{\frac{1}{k+\alpha}} N\right)$ non-zero elements. Therefore, the submatrix E^{M-1} contains at most $\mathcal{O}\left(\frac{N^{1+\frac{1+|\alpha|}{k+\alpha}}}{2} \log^{\frac{1}{k+\alpha}} N\right)$ non-zero elements.

We now consider the submatrix E^{M-2}. Applying the truncation value d_{M-2} to the submatrix A^{M-2}, each row contains at most $\mathcal{O}\left(M^{\frac{1}{2k+\alpha}} 2^{\frac{-M\left(k-\frac{1}{2}-|\alpha|\right)}{2k+\alpha}} + 1\right) = \mathcal{O}(1)$ non-zero elements. The submatrix A^{M-2} has $\frac{N}{4}$ rows. Therefore, the submatrix contains at most $\mathcal{O}\left(\frac{N}{4}\right)$ non-zero elements. Applying the truncation value d'_{M-2} to the submatrices B^{M-2} and C^{M-2}, each row contains at most $\mathcal{O}\left(N^{\frac{1+|\alpha|}{k+\alpha}} \log^{\frac{1}{k+\alpha}} N\right)$ non-zero elements. Since the submatrices B^{M-2} and C^{M-2} both have $\frac{N}{4}$ rows, the submatrices B^{M-2} and C^{M-2} contain at most $\mathcal{O}\left(\frac{N^{1+\frac{1+|\alpha|}{k+\alpha}}}{4} \log^{\frac{1}{k+\alpha}} N\right)$ non-zero elements. Therefore, the submatrix E^{M-2} contains at most $\mathcal{O}\left(\frac{N^{1+\frac{1+|\alpha|}{k+\alpha}}}{4} \log^{\frac{1}{k+\alpha}} N\right)$ non-zero elements.

Using the same argument the submatrices E^m, for $m < M - 2$, contain at most

$\mathcal{O}\left(\frac{N^{1+\frac{1+|\alpha|}{k+\alpha}}}{2^{M-m}}\log^{\frac{1}{k+\alpha}} N\right)$ non-zero elements. Therefore, summing over all submatrices E^m for $m = 0, \ldots, M-1$, the non-standard matrix $K_M^{d,d'}$ contains at most $\mathcal{O}\left(\left(\frac{1}{2}+\frac{1}{4}+\ldots+\frac{1}{2^{M-m}}\right) N^{1+\frac{1+|\alpha|}{k+\alpha}}\log^{\frac{1}{k+\alpha}} N\right) = \mathcal{O}(N^{1+\frac{1+|\alpha|}{k+\alpha}}\log^{\frac{1}{k+\alpha}} N)$ non-zero elements. □

Chapter Review

In this chapter we have introduced the standard and non-standard Galerkin methods with multiwavelet basis functions. For the standard Galerkin method, in proposition 4.1.1 we found an upper bound for the size of matrix elements. Using this bound in section 4.1.2 we have developed a compression strategy where the error introduced by setting small matrix elements to zero is of the same order as the discretisation error. Using this strategy we only compute and store $\mathcal{O}(N \log N)$ elements.

For the non-standard Galerkin method, In proposition 4.2.1 we found upper bounds for the size of matrix elements of the submatrices A^m, B^m and C^m. Then, with these bounds in section 4.2.2 we have developed a compression strategy where the error introduced by the compression is of the same order as the discretisation error. Using this strategy we only compute and store $\mathcal{O}(N^{1+\frac{1+|\alpha|}{k+\alpha}}\log^{\frac{1}{k+\alpha}} N)$ elements.

These methods can then be combined with a conjugate gradient type scheme to solve $\mathcal{A}u = f$, [47]. In the next two chapters we apply the methods developed here to the radiosity problem and Laplace's equation with Neumann and Dirichlet boundary conditions.

Chapter 5

The Radiosity Problem

For the past two decades it has been the aim of researches in computer graphics to create images of non-existent environments, see [48, 49, 50, 51]. Practical applications for such methods range from industrial and architectural design to advertising and entertainment. The creation of an image by evaluating a model of light propagation is called *image synthesis*. Early image synthesis models were based on local illumination where each surface is considered separately from all other surfaces. Greater realism requires that global illumination models are used, which take account of the inter-reflection of light between surfaces. In a global illumination model, when we consider a given surface in an environment, all other surfaces can be considered as light emitters. An early method for solving the global illumination problem was the ray tracing method. However, when using this method if the position of the viewer is changed the solution has to be recomputed. Later methods applied the radiosity techniques of radiant heat transfers to the global illumination problem. Using these techniques the global illumination problem can be modelled mathematically by a second kind integral equation, the solution of which is viewer independent. In this chapter we use the results of chapter 4 to solve a second kind integral

equation which is the mathematical model for the global illumination or radiosity problem.

In section 5.1 we introduce the radiometric quantities that describe the movement of light in an environment. Then, using the radiometric quantities we derive the radiosity equation in section 5.2. Finally in section 5.2.1 we present several numerical examples using the multiwavelet Galerkin methods of chapter 4.

5.1 Radiometric Quantities

In this section we introduce the physical quantities that characterize radiant energy transfers.

The physical quantity used to describe the transfer of radiant energy is *radiance*, denoted by L. Radiance is defined as the amount of energy travelling at some point in a specified direction, per unit time, per unit area perpendicular to the direction of travel, per unit solid angle. Therefore, the energy radiated in a solid angle $d\omega$, from differential area $d\mathbf{p}$, during time interval dt is,

$$L(\mathbf{p}, \theta_i, \varphi_i)\ d\mathbf{p} \cos\theta_i\ d\omega dt \tag{5.1.1}$$

and the power P radiated in this direction satisfies

$$d^2P = L(\mathbf{p}, \theta_i, \varphi_i)\ d\mathbf{p} \cos\theta_i\ d\omega. \tag{5.1.2}$$

Due to it's "per unit solid angle" definition, radiance does not decay with distance. Therefore, a knowledge of the radiance leaving all surfaces is all that is required to create a image of an environment from any viewer position.

In (5.1.2) we describe the power leaving a point on the surface in a specific direction. To obtain the total power leaving a point on the surface we integrate (5.1.2) over a hemisphere Ω,

$$\begin{aligned} dP &= \int_{\Omega} d^2P \\ &= d\mathbf{p} \int_{\Omega} L(\mathbf{p}, \theta_i, \varphi_i) \cos\theta_i \; d\omega. \end{aligned} \tag{5.1.3}$$

Dividing this by $d\mathbf{p}$ we obtain the power per unit area at a point **p**, or the *radiosity*, denoted by B, at point **p**,

$$\begin{aligned} B(\mathbf{p}) &= \frac{dP}{d\mathbf{p}} \\ &= \int_{\Omega} L(\mathbf{p}, \theta_i, \varphi_i) \cos\theta_i \; d\omega. \end{aligned} \tag{5.1.4}$$

To describe the light sources in an environment we introduce the quantity *exitance*. Exitance is defined as the energy radiated per unit time, per unit area. Exitance is similar to radiosity in that it can be expressed as the integral of the emitted radiance,

$$E(\mathbf{p}) = \int_{\Omega} L_e(\mathbf{p}, \theta_i, \varphi_i) \cos\theta_i \; d\omega. \tag{5.1.5}$$

To be able to fully describe the transfer of light within an environment, in addition to the quantities already described we require a knowledge of the reflective properties of all the surfaces in the environment. The reflecting properties of a given material are described by the concept of reflectance, specifying the characteristics of the reflected light. The most general expression of reflectance is the *bidirectional reflectance distribution function* (BRDF). The BRDF is the ratio of the radiance in the reflected direction and the radiant flux density (power per unit area) in the incident direction. It is a function of both the

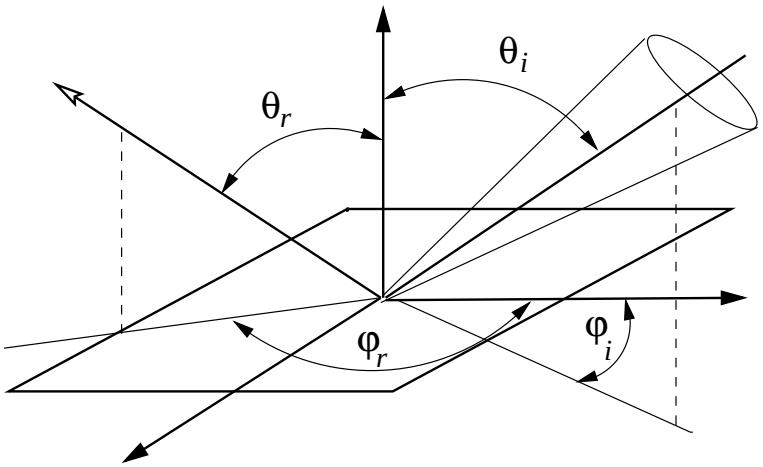


Figure 5.1: Notation for the definition of the BRDF

incident and reflected directions and is denoted ρ_{bd}. The incident radiant flux density coming from a differential solid angle $d\omega$ around the direction (θ_i, φ_i), as shown in Figure 5.1, is $d\Theta_i = L_i(\mathbf{p}, \theta_i, \varphi_i) \cos\theta_i \, d\omega$. Therefore, we can write the BRDF as

$$\rho_{bd}(\theta_r, \varphi_r, \theta_i, \varphi_i) = \frac{L(\mathbf{p}, \theta_r, \varphi_r)}{L_i(\mathbf{p}, \theta_i, \varphi_i) \cos\theta_i \, d\omega}. \tag{5.1.6}$$

5.2 The Radiosity Equation

In the general case, the energy equilibrium for a set of radiating surfaces is expressed by the following integral equation,

$$\underbrace{L(\mathbf{p}, \theta_r, \phi_r)}_{\text{total radiance}} = \underbrace{L_e(\mathbf{p}, \theta_r, \phi_r)}_{\text{emitted radiance}} + \underbrace{\int_\Omega \rho_{bd}(\mathbf{p}, \theta_r, \phi_r, \theta_i, \phi_i) L_i(\mathbf{p}, \theta_i, \phi_i) \cos\theta_i \, d\omega}_{\text{reflected radiance}}. \tag{5.2.1}$$

The first term on the r.h.s. of equation (5.2.1) represents the light emitted by the surface, this is only non-zero for light sources. The second term on the r.h.s. represents the effect of light reflected from other points on the surface. Using the radiosity method we

solve a simplified version of equation (5.2.1), under the assumption that all surfaces are *ideal diffuse reflectors*. That is, the surface reflects light equally in all directions. Then, the BRDF is independent of both the incident and reflected directions and reduces to a function of position only,

$$\rho_{bd}\left(\mathbf{p}, \theta_r, \varphi_r, \theta_i, \varphi_i\right) \equiv \rho_{bd}(\mathbf{p}).$$

As we now show, we can now use radiosity to describe the light in an environment, rather than the radiance. Under the assumption that the surfaces are ideal diffuse reflectors, radiance is a function of position only, that is

$$L\left(\mathbf{p}, \theta_i, \varphi_i\right) \equiv L(\mathbf{p}).$$

Then, substituting this into equation (5.1.4) we obtain,

$$\begin{aligned} B(\mathbf{p}) &= L(\mathbf{p}) \int_{\Omega} \cos\theta_i \; d\omega \\ &= L(\mathbf{p}) \int_0^{\pi} \int_0^{2\pi} \cos\theta_i \sin\theta_i \; d\theta_i d\varphi_i \\ &= \pi L(\mathbf{p}). \end{aligned} \tag{5.2.2}$$

Thus, radiosity is proportional to radiance and they can be used interchangeably to characterize light leaving ideal diffuse surfaces. Similarly, we obtain

$$E(\mathbf{p}) = \pi L_e(\mathbf{p}) \tag{5.2.3}$$

and therefore, we can interchange exitance and emitted radiance. If we now substitute

relationships (5.2.2) and (5.2.3) into equation (5.2.1) we obtain,

$$B(\mathbf{p}) = E(\mathbf{p}) + \pi\rho_{bd}(\mathbf{p}) \int_{\Omega} L_i\left(\mathbf{p}, \theta_i, \varphi_i\right) \cos\theta_i \; d\omega. \tag{5.2.4}$$

Let **q** be a point visible from the point **p** in the direction (θ_i, φ_i), then the point **p** is also visible from the point **q** in the direction (θ_i', φ_i'). Therefore, the invariance of radiance along a line of sight states that

$$L_i\left(\mathbf{p}, \theta_i, \varphi_i\right) = L\left(\mathbf{q}, \theta_i', \varphi_i'\right).$$

Then, using relation (5.2.2) we have that the incident radiance at point **p** is proportional to the radiosity at point **q**, that is,

$$L_i\left(\mathbf{p}, \theta_i, \varphi_i\right) = \frac{B(\mathbf{q})}{\pi}. \tag{5.2.5}$$

The integral in equation (5.2.4) is now written as an integral over all surfaces in an environment by expanding the differential solid angle,

$$d\omega = \begin{cases} \frac{\cos\theta_i' d\Gamma_{\mathbf{q}}}{|\mathbf{p}-\mathbf{q}|^2} & \text{in 3D} \\ \frac{\cos\theta_i' d\Gamma_{\mathbf{q}}}{|\mathbf{p}-\mathbf{q}|} & \text{in 2D} \end{cases}, \tag{5.2.6}$$

and setting the domain of integration to be the set of all surfaces in the environment that are visible from point **p**. This is achieved by including a visibility function $V(\mathbf{p}, \mathbf{q})$, such that

$$V(\mathbf{p}, \mathbf{q}) = \begin{cases} 1 & \text{if } \mathbf{p} \text{ and } \mathbf{q} \text{ are mutually visible,} \\ 0 & \text{otherwise.} \end{cases} \tag{5.2.7}$$

Thus, in two dimensions, equation (5.2.4) reduces to the radiosity equation,

$$B(\mathbf{p}) = E(\mathbf{p}) + \rho_{bd}(\mathbf{p}) \int_{\Gamma} \frac{\cos\theta_i \cos\theta_i'}{|\mathbf{p}-\mathbf{q}|} V(\mathbf{p},\mathbf{q}) B(\mathbf{q}) d\Gamma_{\mathbf{q}}, \tag{5.2.8}$$

where the exitance $E(\mathbf{p})$ is known and the radiosity $B(\mathbf{p})$ is unknown.

5.2.1 Numerical Results

We now present the numerical results for a radiosity problem on an ellipse of circumference 4π, with minor axis 1.29704815 and major axis 2.5940936 centered at the origin. We consider the environment when two light sources are placed at $\mathbf{p}_0 = \left(1.2477, \frac{3}{5}\pi\right)$ and $\mathbf{p}_1 = \left(1.1909, \frac{7}{5}\pi\right)$, with strengths 1.5 and 1, respectively. We first consider the solution of the problem using the non-standard Galerkin method discussed in section 4.2. We then consider a standard Galerkin method using a multiwavelet basis. In practice for the radiosity problem we do not know the exact level of discretisation error, η. Therefore, we estimate the discretisation error as $\eta = k2^{-M}$.

Non-standard Results

In tables 5.1-5.4 the column $\|u_h - \overline{u}_h\|$ is the L_2 error introduced when using the non-standard Galerkin method. The column nz is the number of non-zero elements of the matrix that are computed. The column $\%$ gives the percentage of the matrix entries that have not been computed without any detrimental effect to the solution. Here, the conjugate gradient square (CGS) method, [47], was used to solve the linear system, and never required more than 16 iterations to converge.

M	$\|u_h - \overline{u}_h\|$	nz	%
4	5.3428×10^{-1}	164	35.94
5	3.3755×10^{-1}	420	59.98
6	1.9688×10^{-1}	1412	65.53
7	8.6546×10^{-2}	5224	68.12
8	4.7271×10^{-2}	19016	70.98
9	3.0266×10^{-2}	66576	74.60
10	1.7667×10^{-2}	240080	77.10

Table 5.1: $k = 1$

M	$\|u_h - \overline{u}_h\|$	nz	%
4	1.2856×10^{-1}	528	48.44
5	2.8012×10^{-2}	1936	52.73
6	7.1220×10^{-3}	6816	58.40
7	2.1748×10^{-3}	23968	63.43
8	6.1847×10^{-4}	89312	65.93
9	1.5865×10^{-4}	352192	66.41
10	6.5420×10^{-5}	1283136	69.41

Table 5.2: $k = 2$

M	$\|u_h - \overline{u}_h\|$	nz	%
4	1.3822×10^{-2}	1476	35.94
5	1.5213×10^{-3}	4356	52.73
6	2.1048×10^{-4}	15480	58.01
7	2.5693×10^{-5}	59112	59.91
8	3.6313×10^{-6}	232056	60.66
9	4.0529×10^{-7}	882288	62.60
10	4.90529×10^{-8}	3509712	62.81

Table 5.3: $k = 3$

M	$\|u_h - \overline{u}_h\|$	nz	%
4	1.9113×10^{-3}	2624	35.94
5	1.1138×10^{-4}	7744	52.73
6	5.6804×10^{-6}	29312	55.27
7	3.4306×10^{-7}	109440	58.25
8	3.9489×10^{-8}	420736	59.88
9	1.2869×10^{-9}	1619200	61.40

Table 5.4: $k = 4$

The sparsity patterns of the non-standard matrices for $k = 3$, $M = 8$ and $k = 4$, $M = 8$ are shown in Figures 5.2 and 5.3, respectively. Studying the structure of the non-standard matrices in Figures 5.2 and 5.3 we can clearly see that the multiwavelet levels have been fully decomposed, that is, there are only interactions between multiwavelets and scaling functions of the same resolution. We also note, that due to the presence of two multiwavelets the banding of the submatrices A^m is significantly tighter than the banding of the submatrices B^m and C^m.

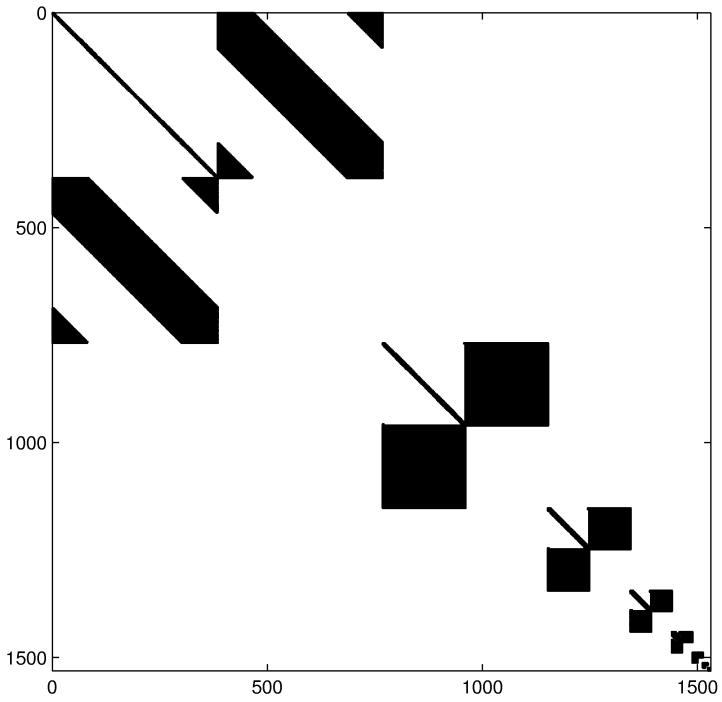


Figure 5.2: Non-standard matrix: $k = 3$, $M = 8$

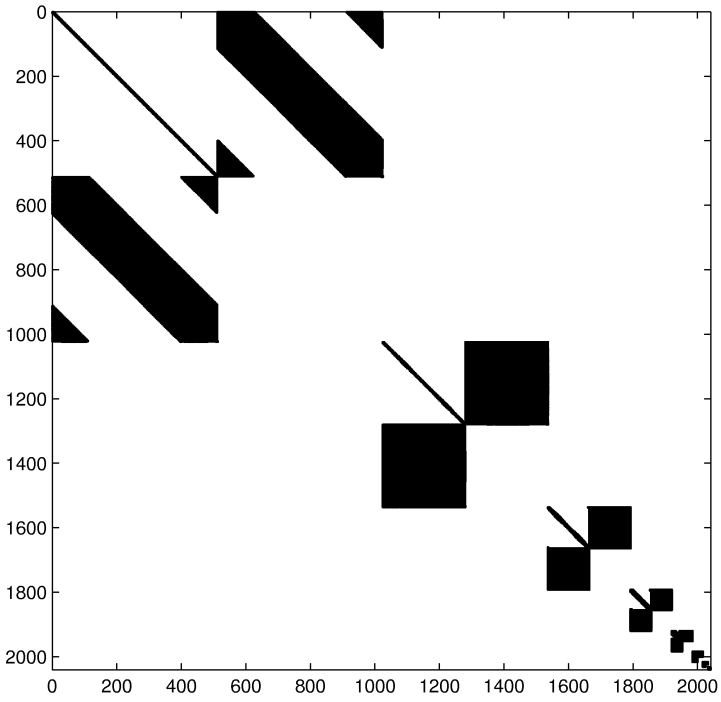


Figure 5.3: Non-standard matrix: $k = 4$, $M = 8$

We note that in tables 5.1-5.4, the $\%$ of matrix entries that have not been computed without any detrimental effect to the solution, is higher for $k = 1$. This is due to the fact that as k increases the approximate discretisation error decreases rapidly to the level of computer accuracy. For a given problem, once we decide on the number of *mother* wavelets k and the level of discretisation M, the non-standard and standard matrices, K_M and A_M, respectively, are fixed. Moreover, using Theorem 2.4.4 and Lemma 3.2.1 we see that the discretisation error is bounded as

$$\|u - u_h\| \leq ch^k \sup_{x\in[0,1]} \left|u^{(k)}(x)\right|. \tag{5.2.9}$$

Now, if the problem is "difficult" the bound (5.2.9) is large. Therefore, we can set many of the entries of the non-standard and standard matrices, K_M and A_M, respectively, to

zero. If the problem is "less difficult" the bound (5.2.9) is smaller. Therefore, less entries of the non-standard and standard matrices, K_M and A_M, respectively, can be set to zero. Hence, due to the fixed accuracy of computers the optimal complexity estimate may not always be observable. In Figures 5.4 and 5.5 we plot the non zero elements of the non-stand matrix for $k = 3$, $M = 8$, when entries less than 10^{-8} and 10^{-6}, respectively, have been set to zero.

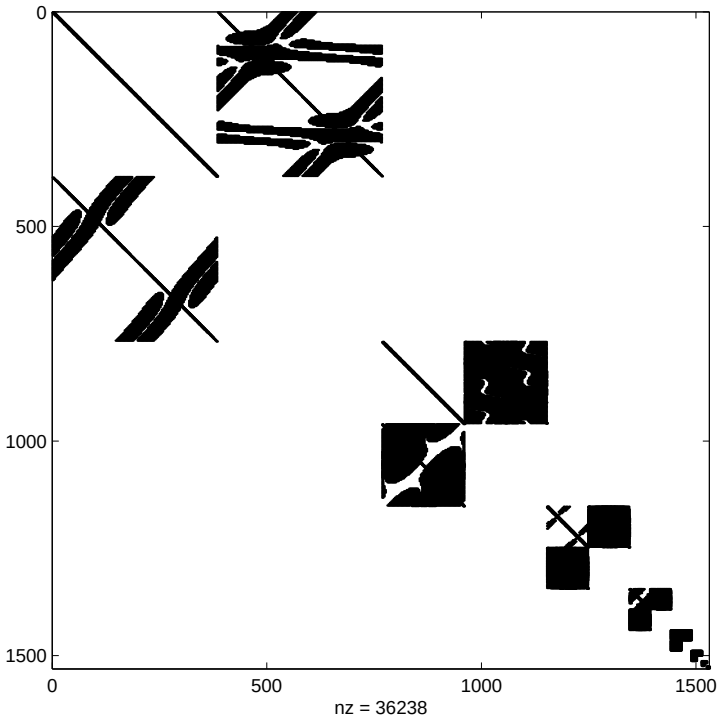


Figure 5.4: Non-standard matrix with a threshold 10^{-8}: $k = 3$, $M = 8$

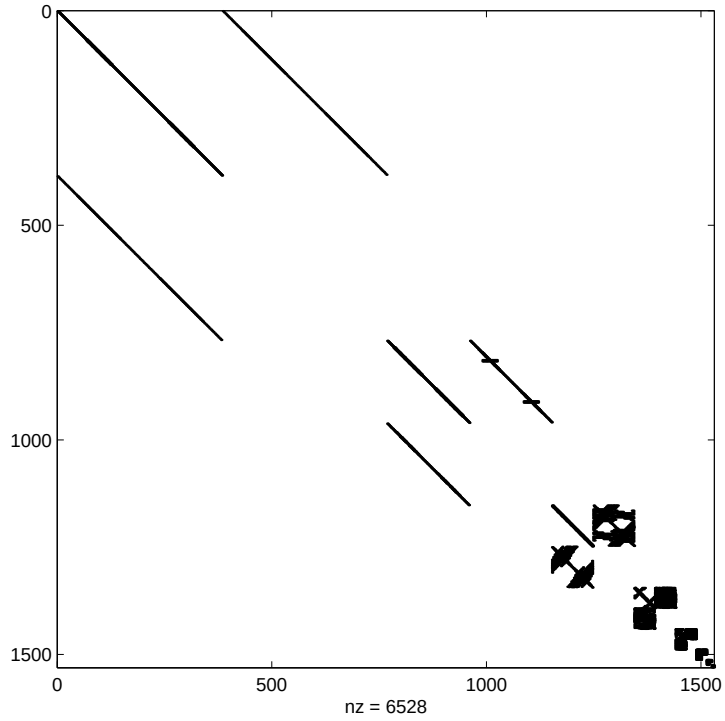


Figure 5.5: Non-standard matrix with a threshold 10^{-6}: $k = 3$, $M = 8$

Standard Results

In tables 5.5-5.8 the L_2 norm of the error of the compressed system is denoted by $\|u - \overline{u}_h\|$ and *nz* is the number of non-zero elements. The column $\%$ gives the percentage of the matrix entries that can be set to zero without any detrimental effect. Here, the CGS method was used to solve the linear system, and never required more than 16 iterations to converge.

M	$\|u_h - \overline{u}_h\|$	nz	%
4	6.9223×10^{-1}	30	88.28
5	3.3883×10^{-1}	146	85.74
6	1.3372×10^{-1}	442	89.21
7	5.7805×10^{-2}	1078	93.42
8	3.4555×10^{-2}	2486	96.21
9	2.3982×10^{-2}	4718	98.20
10	1.4199×10^{-2}	8102	99.23

Table 5.5: $k = 1$

M	$\|u_h - \overline{u}_h\|$	nz	%
4	1.3481×10^{-1}	168	83.59
5	2.5724×10^{-2}	416	89.84
6	6.8517×10^{-3}	1300	92.07
7	2.1018×10^{-3}	2354	96.41
8	6.0852×10^{-4}	5882	97.76
9	1.4283×10^{-4}	14534	98.61
10	6.3136×10^{-5}	23824	99.43

Table 5.6: $k = 2$

M	$\|u_h - \overline{u}_h\|$	nz	%
4	4.5796×10^{-2}	138	94.01
5	1.0363×10^{-3}	1134	97.70
6	2.2526×10^{-4}	3192	91.34
7	2.3683×10^{-5}	8284	94.38
8	3.8771×10^{-6}	18940	96.79
9	4.0126×10^{-7}	46397	98.03
10	4.5481×10^{-8}	109532	98.84

Table 5.7: $k = 3$

M	$\|u_h - \overline{u}_h\|$	nz	%
4	1.1632×10^{-3}	773	81.13
5	1.2851×10^{-4}	2457	85.00
6	6.5478×10^{-6}	6693	89.79
7	2.7598×10^{-7}	19158	92.69
8	2.9482×10^{-8}	48010	95.42
9	1.8780×10^{-9}	116126	97.23

Table 5.8: $k = 4$

The sparsity patterns of the standard matrices for $k = 3$, $M = 8$ and $k = 4$, $M = 8$ are shown in Figures 5.6 and 5.8, respectively. The sparsity patterns display the so-called 'finger' structure, this occurs when the kernel of the integral equation has non-polynomial like behaviour along the diagonal. Figures 5.7 and 5.9 show the eigenvalues of the respective matrices clustering about 1. In this case we know that conjugate gradient type schemes have fast convergence, in $\mathcal{O}(1)$ iterations.

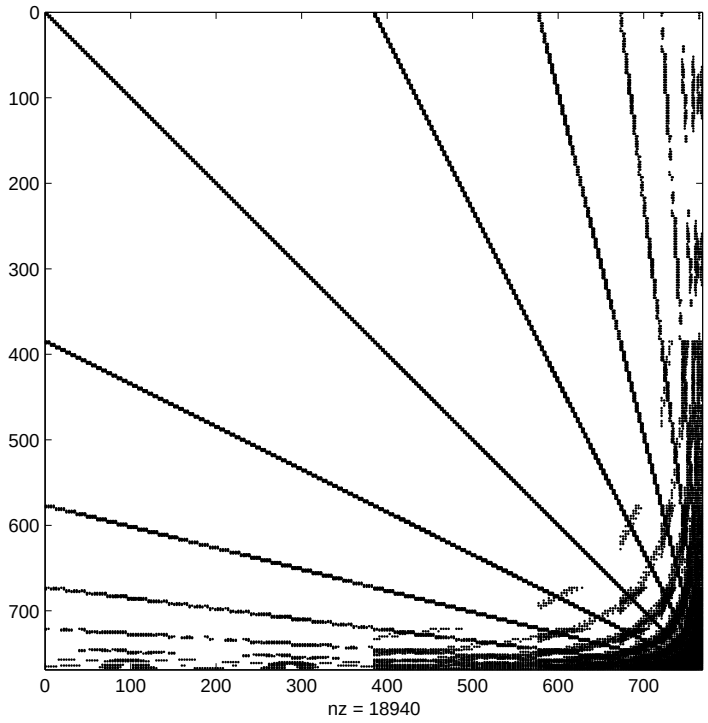


Figure 5.6: Standard Matrix: $k = 3$, $M = 8$

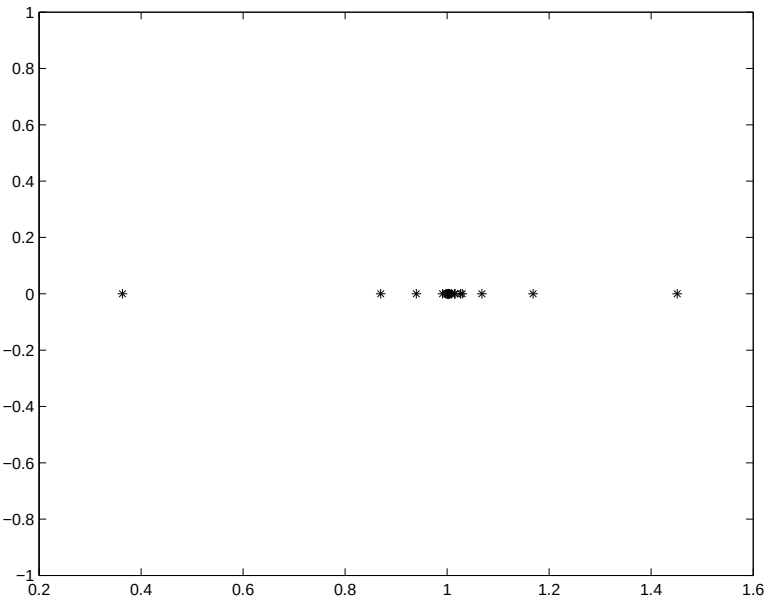


Figure 5.7: Standard Matrix Eigenvalues: $k = 3$, $M = 8$

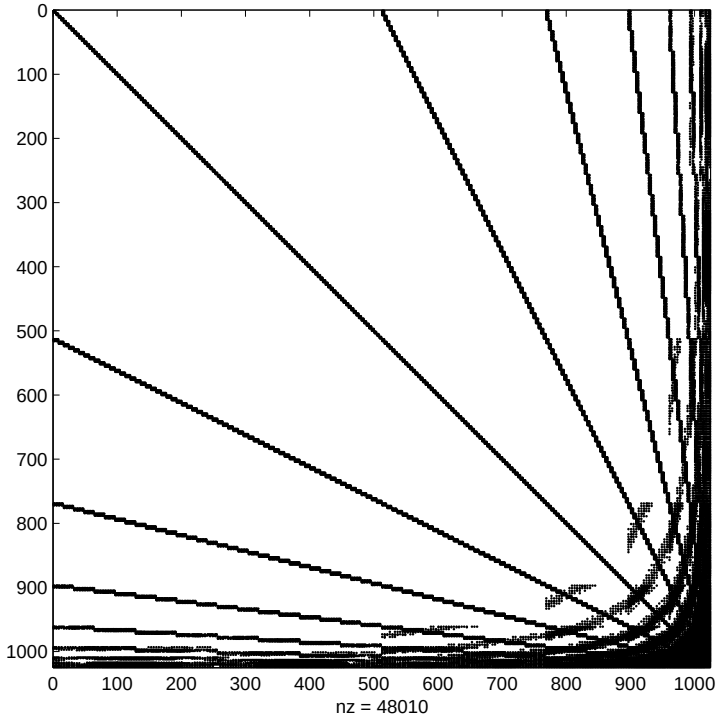


Figure 5.8: Standard Matrix: $k = 4$, $M = 8$

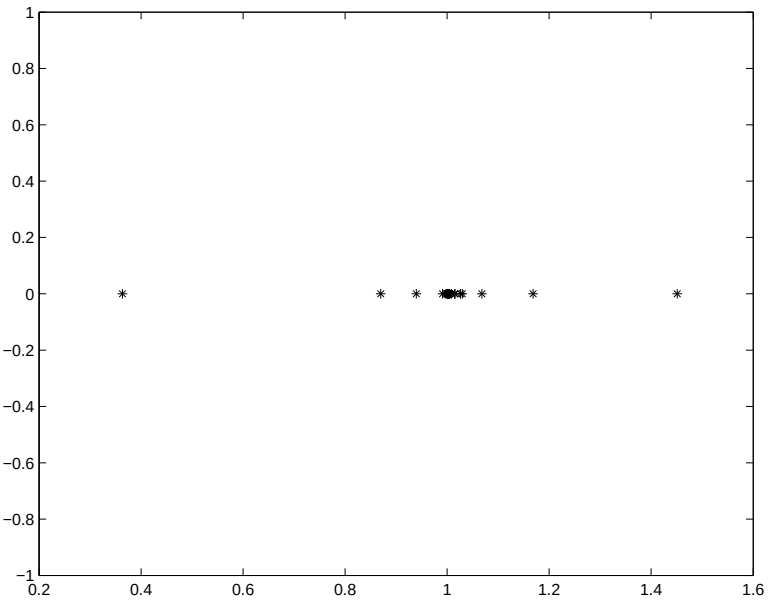


Figure 5.9: Standard Matrix Eigenvalues: $k = 4$, $M = 8$

In Figures 5.10 and 5.11 we plot the non zero elements of the standard matrix for $k = 3$, $M = 8$, when entries less than 10^{-8} and 10^{-6}, respectively, have been set to zero.

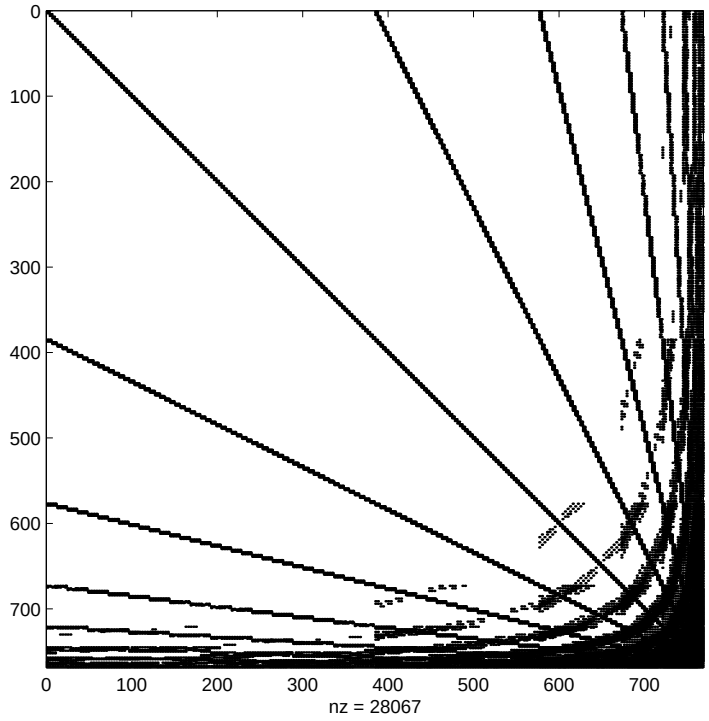


Figure 5.10: Standard matrix with a threshold 10^{-8}: $k = 3$, $M = 8$

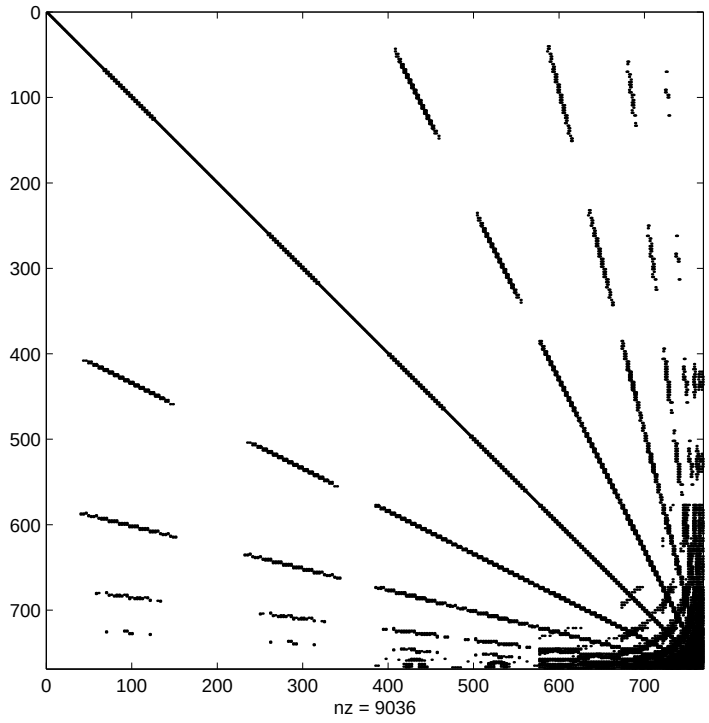


Figure 5.11: Standard matrix with a threshold 10^{-6}: $k = 3$, $M = 8$

Chapter Review

In this chapter we have introduced image synthesis, the creation of an image by evaluating a model of light propagation. Image synthesis methods have practical applications ranging from industrial and architectural design to advertising and entertainment.

We have discussed the physical quantities that characterize radiant energy transfers, namely, radiance, radiosity and exitance. The transfer of light in an environment is governed by the equilibrium equation,

$$\underbrace{L(\mathbf{p},\theta_r,\phi_r)}_{\text{total radiance}} = \underbrace{L_e(\mathbf{p},\theta_r,\phi_r)}_{\text{emitted radiance}} + \underbrace{\int_\Omega \rho_{bd}(\mathbf{p},\theta_r,\phi_r,\theta_i,\phi_i) L_i(\mathbf{p},\theta_i,\phi_i)\cos\theta_i \; d\omega}_{\text{reflected radiance}} .$$

We assume that all the surfaces in the environment are ideal diffuse surfaces. That is, they reflect light equally in all directions. Under this assumption, radiance is proportional to radiosity and emitted radiance is proportional to exitance. Therefore, radiosity can be used to describe the transfer of light in an environment, which is now governed by the radiosity equation,

$$B(\mathbf{p}) = E(\mathbf{p}) + \rho_{bd}(\mathbf{p}) \int_\Gamma \frac{\cos\theta_i \cos\theta_i'}{|\mathbf{p}-\mathbf{q}|} V(\mathbf{p},\mathbf{q}) B(\mathbf{q}) d\Gamma_{\mathbf{q}}.$$

We have presented numerical results for the solution of the radiosity equation, for both the standard and non-standard methods have been obtained.

Chapter 6

Numerical Solution of Laplace's Equation

In this chapter we consider the numerical solution of Laplace's equation on the exterior of a domain Ω, with either Neumann or Dirichlet boundary conditions. Let $\Gamma = \partial\Omega$ be a smooth closed curve. Then , as shown in section 2.3, Laplace's equation on the exterior domain,

$$\begin{aligned} \nabla^2 u(\mathbf{p}) &= 0, \qquad \mathbf{p} \in \Omega_+ \\ \lim_{\mathbf{p}\to\infty} |u(\mathbf{p})| &= 0, \end{aligned} \tag{6.0.1}$$

can be reformulated as a boundary integral equation,

$$\left(-\frac{1}{2}\mathcal{I} + \mathcal{M}\right) u(\mathbf{p}) = \mathcal{L}\frac{\partial u}{\partial n}(\mathbf{p}), \qquad \mathbf{p} \in \Gamma, \tag{6.0.2}$$

on the boundary Γ, for the single- and double-layer operators $\mathcal{L}$ and $\mathcal{M}$, respectively. We wish to solve (6.0.2) using the multiwavelet Galerkin methods of chapter 4. The resulting

linear system will be solved by an iterative technique such as conjugate gradient squares (CGS) or generalized minimal residual (GMRES), [52]. The use of an iterative solver is efficient if the linear system is well conditioned and its eigenvalues cluster at a point different from zero.

When we consider equation (6.0.2) with Neumann boundary conditions, that is $\frac{\partial u}{\partial n}$ is given and u is unknown, the resulting linear system is well conditioned and the eigenvalues cluster about $-\frac{1}{2}$. In this case, an iterative method of the conjugate gradient type can converge in $\mathcal{O}(1)$ iterations. In section 6.1 we present numerical results for the solution of several Neumann test problems. However, when we consider equation (6.0.2) with Dirichlet boundary conditions, that is u is given and $\frac{\partial u}{\partial n}$ is unknown, the resulting linear system is ill conditioned. This is because the single-layer operator $\mathcal{L}$ is of order -1 and therefore its eigenvalues cluster at zero. Therefore, in order to use an iterative solver we precondition the linear system, [53]. In section 6.2 we discuss the preconditioner used when wavelet bases are employed, we then extend this to our case where multiwavelet bases are employed. In section 6.2.4 we present numerical results for a Dirichlet test problems.

6.1 The Neumann Problem

In this section we consider the solution of equation (6.0.2) with Neumann boundary conditions. we need to solve the second kind equation

$$-\frac{1}{2}u(\mathbf{p}) + \int_\Gamma u(\mathbf{q})\frac{\partial G(\mathbf{p},\mathbf{q})}{\partial n_{\mathbf{q}}}d\Gamma_{\mathbf{q}} = \int_\Gamma G(\mathbf{p},\mathbf{q})\frac{\partial u(\mathbf{q})}{\partial n_{\mathbf{q}}}d\Gamma_{\mathbf{q}}, \quad \mathbf{p}\in\Gamma, \tag{6.1.1}$$

for the unknown Dirichlet boundary condition u. Once u is found on the boundary we can use the integral representation (2.3.18), namely,

$$u(\mathbf{p}) = \int_\Gamma u(\mathbf{p}) \frac{\partial G(\mathbf{p},\mathbf{q})}{\partial n_\mathbf{q}} d\Gamma_\mathbf{q} - \int_\Gamma G(\mathbf{p},\mathbf{q}) \frac{\partial u(\mathbf{q})}{\partial n_\mathbf{q}} d\Gamma_\mathbf{q}, \quad \mathbf{p} \in \Omega_+,$$

to find u anywhere in the exterior domain Ω_+.

6.1.1 Numerical Results

Problem One

Here, we consider the numerical solution of equation (6.1.1) exterior to an ellipse of circumference 4π, with major axis 2.5940938 and minor axis 1.2970468 centered at the origin. We consider a Neumann problem, equivalent to that generated by three interior point sources placed at $\mathbf{p}_0 = \left(1.3611, \frac{3}{5}\pi\right)$, $\mathbf{p}_1 = \left(1.1909, \frac{7}{5}\pi\right)$ and $\mathbf{p}_2 = \left(1.1342, \frac{8}{5}\pi\right)$ with strengths 2, 1.5 and 1.5, respectively. The field generated is $u(\mathbf{p}) = -\frac{1}{\pi}\ln|\mathbf{p}-\mathbf{p}_0| - \frac{1.5}{2\pi}\ln|\mathbf{p}-\mathbf{p}_1| - \frac{1.5}{\pi}\ln|\mathbf{p}-\mathbf{p}_2|$. We first consider the solution of the problem using the non-standard Galerkin method discussed in section 4.2. Then, we consider a standard Galerkin method using a multiwavelet basis.

Non-Standard Results

We denote by $\overline{u}_h$ the solution of the compressed system. Then, in tables 6.1-6.4, the L_2 norm of the error of the compressed system is denoted by $\|u - \overline{u}_h\|$ and *nz* is the number of non-zero elements of the non-standard matrix that are computed. The column $\%$ gives the percentage of the matrix entries that have not been computed without any detrimental

effect to the solution. Here, the CGS method was used to solve the linear system, and never required more than 14 iterations to converge.

M	$\|u - \overline{u}_h\|$	nz	$\%$
4	6.4281×10^{-2}	124	51.56
5	3.2171×10^{-2}	308	69.92
6	1.6862×10^{-2}	900	78.03
7	8.3855×10^{-3}	2248	86.28
8	4.0197×10^{-3}	7944	87.88
9	2.4887×10^{-3}	25488	90.28
10	1.1519×10^{-3}	100048	90.46

Table 6.1: $k = 1$

M	$\|u - \overline{u}_h\|$	nz	$\%$
4	7.2992×10^{-3}	528	48.44
5	2.5857×10^{-3}	1616	60.55
6	1.1753×10^{-3}	4384	73.24
7	3.5843×10^{-4}	11552	82.37
8	7.7036×10^{-5}	43584	83.27
9	2.9272×10^{-5}	186944	82.17
10	4.8206×10^{-6}	653952	84.41

Table 6.2: $k = 2$

M	$\|u - \overline{u}_h\|$	nz	%
4	2.7309×10^{-3}	1188	48.44
5	9.8955×10^{-4}	2772	69.92
6	1.3717×10^{-4}	9252	74.90
7	1.5517×10^{-5}	29736	79.83
8	1.9597×10^{-6}	118440	79.92
9	2.4560×10^{-7}	437616	81.45
10	3.0737×10^{-8}	1790928	81.02

Table 6.3: $k = 3$

M	$\|u - \overline{u}_h\|$	nz	%
4	1.4305×10^{-3}	2112	48.44
5	1.6489×10^{-4}	4928	69.92
6	9.7781×10^{-6}	16512	74.80
7	1.1039×10^{-6}	52608	79.93
8	8.3695×10^{-8}	204672	80.48
9	4.2303×10^{-9}	836864	80.05
10	2.6474×10^{-10}	3331840	80.15

Table 6.4: $k = 4$

The sparsity patterns of the non-standard matrices for problem on with $k = 3$, $M = 8$ and $k = 4$, $M = 8$ are shown in Figures 6.1 and 6.2, respectively.

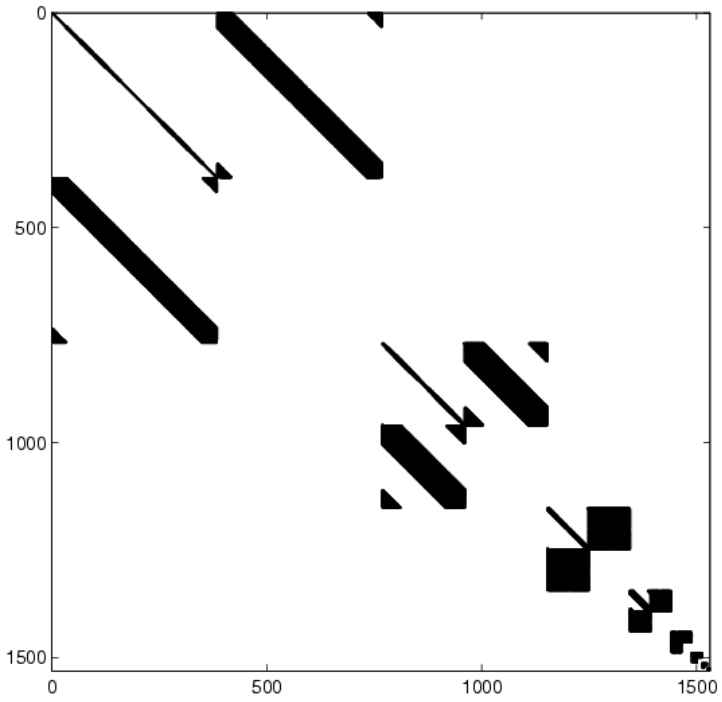


Figure 6.1: Non-standard matrix: $k = 3$, $M = 8$

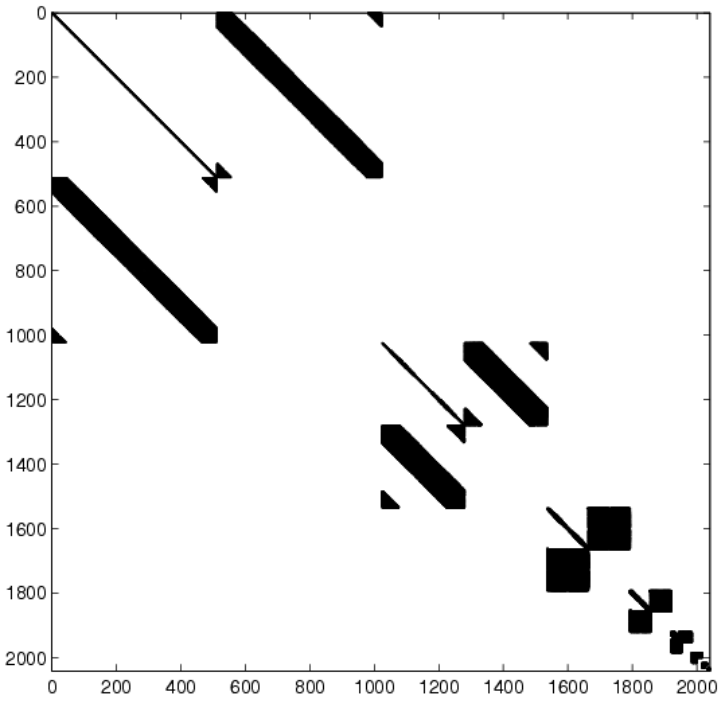


Figure 6.2: Non-standard matrix: $k = 4$, $M = 8$

Standard Results

In tables 6.5-6.8 the L_2 norm of the error of the compressed system is denoted by $\|u - \overline{u}_h\|$ and *nz* is the number of non-zero elements. The column $\%$ gives the percentage of the matrix entries that can be set to zero without any detrimental effect. Here, the CGS method was used to solve the linear system, and never required more than 14 iterations to converge.

M	$\|u - \overline{u}_h\|$	nz	%
4	5.4698×10^{-2}	30	88.28
5	2.8932×10^{-2}	46	95.51
6	1.5063×10^{-2}	114	97.22
7	8.1767×10^{-3}	234	98.57
8	4.0179×10^{-3}	606	99.08
9	1.9259×10^{-3}	1462	99.44
10	1.1253×10^{-3}	2290	99.78

Table 6.5: $k = 1$

M	$\|u - \overline{u}_h\|$	nz	%
4	7.1011×10^{-3}	84	91.80
5	3.2568×10^{-3}	120	97.07
6	1.0855×10^{-3}	358	97.81
7	3.2906×10^{-4}	706	98.92
8	7.5767×10^{-5}	1686	99.36
9	1.8639×10^{-5}	3826	99.64
10	4.6889×10^{-6}	8250	99.80

Table 6.6: $k = 2$

M	$\|u - \overline{u}_h\|$	nz	%
4	3.5872×10^{-3}	84	96.35
5	9.9750×10^{-4}	228	97.53
6	1.3878×10^{-4}	600	98.37
7	1.7144×10^{-5}	1276	99.13
8	2.2674×10^{-6}	3002	99.49
9	3.2507×10^{-7}	5458	99.77
10	3.5897×10^{-8}	13162	99.86

Table 6.7: $k = 3$

M	$\|u - \overline{u}_h\|$	nz	%
4	1.6417×10^{-3}	106	97.41
5	2.1813×10^{-4}	272	98.34
6	1.1242×10^{-5}	804	98.77
7	1.4569×10^{-6}	1622	99.38
8	7.1202×10^{-8}	4690	99.55
9	5.4075×10^{-9}	8534	99.80
10	3.7089×10^{-10}	17470	99.90

Table 6.8: $k = 4$

The sparsity patterns of the standard matrices for problem one with $k = 3$, $M = 8$ and $k = 4$, $M = 8$ are shown in Figures 6.3 and 6.5, respectively. We note, that since the kernel is polynomial like along the diagonal we do not obtain the so-called 'finger' structure. Instead since the kernel is smooth everywhere, so we only have significant interactions involving the lowest levels. Figures 6.4 and 6.6 show the eigenvalues of the respective standard matrices rapidly clustering about $-\frac{1}{2}$.

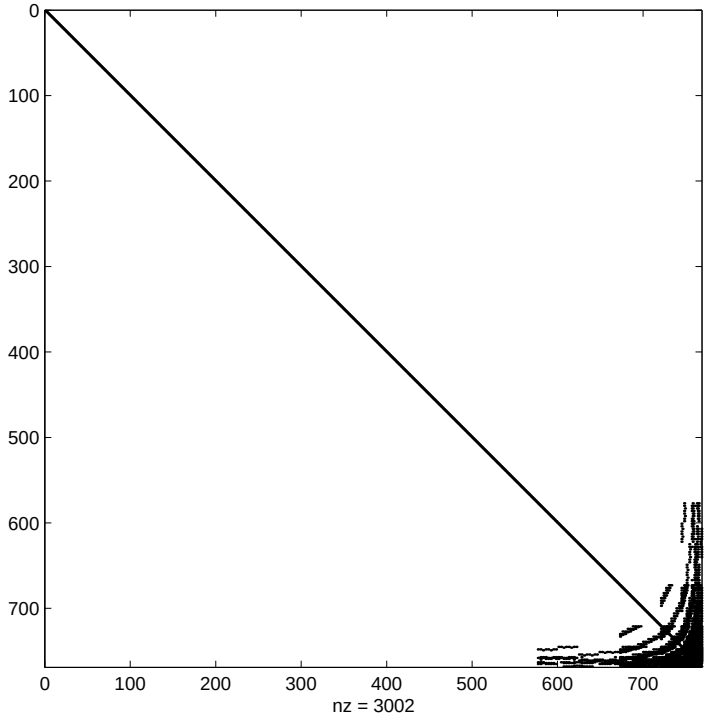


Figure 6.3: Standard matrix: $k = 3$, $M = 8$

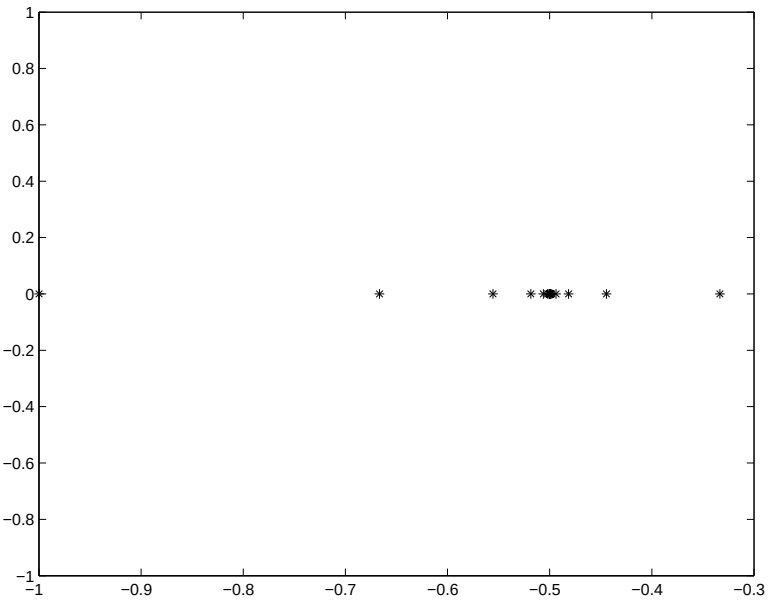


Figure 6.4: Standard Matrix Eigenvalues: $k = 3$, $M = 8$

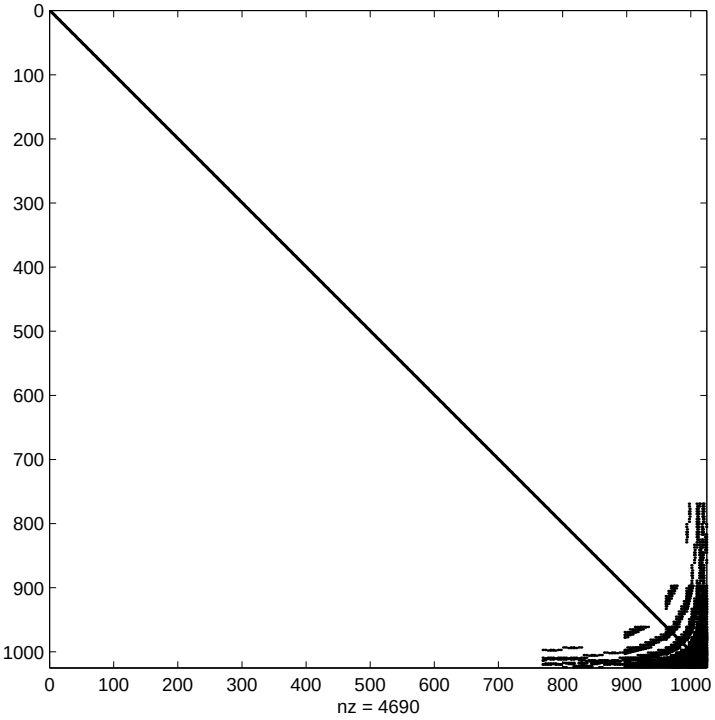


Figure 6.5: Standard matrix: $k = 4$, $M = 8$

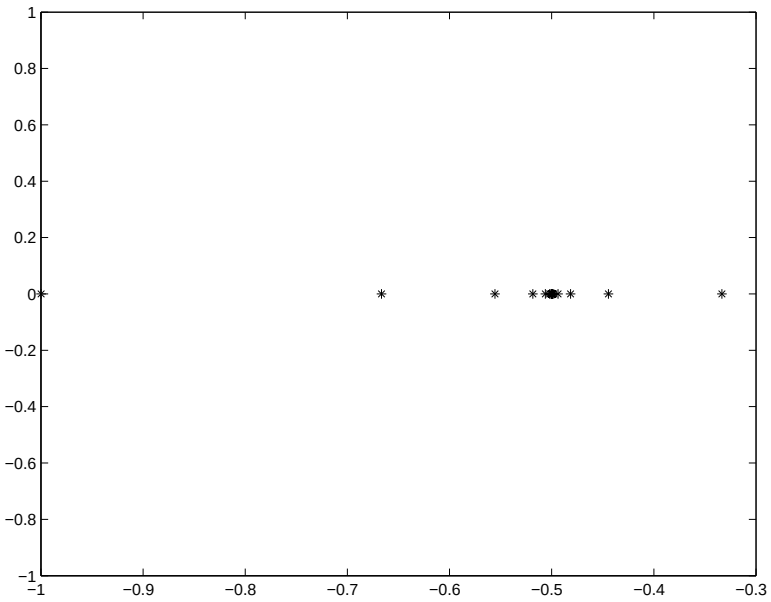


Figure 6.6: Standard Matrix Eigenvalues: $k = 4$, $M = 8$

Problem Two

In problem two we again consider the numerical solution of equation (6.1.1) exterior to an ellipse of circumference 4π, with major axis 2.5940938 and minor axis 1.2970468 centered at the origin. Here, we consider a Neumann problem, equivalent to that generated by two interior point sources placed at $\mathbf{p}_0 = \left(1.7981, \frac{7}{10}\pi\right)$ and $\mathbf{p}_1 = \left(1.4178, \frac{7}{5}\pi\right)$ both with strength 2. These point sources are closer to the boundary than those in problem one, this leads to a much "nastier" solution. The field generated is $u(\mathbf{p}) = -\frac{1}{\pi}\ln|\mathbf{p} - \mathbf{p}_0| - \frac{1}{2\pi}\ln|\mathbf{p} - \mathbf{p}_1|$. We first consider the solution of the problem using the non-standard Galerkin method discussed in section 4.2. Then, we consider a standard Galerkin method using a multiwavelet basis.

Non-standard Results

As before, in tables 6.9-6.12 the L_2 norm of the error is denoted by $\|u - \overline{u}_h\|$ and *nz* is the number of non-zero elements of the non-standard matrix that are computed. The column $\%$ gives the percentage of the matrix entries that have not been computed without any detrimental effect on the solution. Here, the CGS method was used to solve the linear system, and never required more than 14 iterations to converge.

M	$\|u - \overline{u}_h\|$	nz	$\%$
4	9.7429×10^{-2}	124	51.56
5	5.9653×10^{-2}	268	73.83
6	3.2539×10^{-2}	564	86.23
7	1.7673×10^{-2}	1544	90.58
8	8.9169×10^{-3}	5000	92.37
9	4.4959×10^{-3}	16272	93.79
10	2.3293×10^{-3}	37712	96.40

Table 6.9: $k = 1$

M	$\|u - \overline{u}_h\|$	nz	%
4	4.9823×10^{-2}	528	48.44
5	1.9560×10^{-2}	1232	69.92
6	9.3892×10^{-3}	2640	83.92
7	2.0211×10^{-3}	5600	91.46
8	8.2200×10^{-4}	14112	94.62
9	2.0150×10^{-4}	43488	95.85
10	5.1577×10^{-5}	135584	96.77

Table 6.10: $k = 2$

M	$\|u - \overline{u}_h\|$	nz	%
4	3.4996×10^{-2}	1116	51.56
5	1.1482×10^{-2}	2412	73.83
6	2.4024×10^{-3}	5076	86.23
7	5.6894×10^{-4}	10584	92.82
8	5.7816×10^{-5}	30600	94.81
9	9.5992×10^{-6}	104040	95.59
10	1.2087×10^{-6}	365904	96.12

Table 6.11: $k = 3$

M	$\|u - \overline{u}_h\|$	nz	$\%$
4	2.5927×10^{-2}	1984	51.56
5	3.7868×10^{-3}	4416	73.05
6	1.3000×10^{-3}	9536	85.45
7	7.2456×10^{-5}	19840	92.43
8	1.3431×10^{-5}	56704	94.59
9	6.9589×10^{-7}	181632	95.67
10	4.2181×10^{-8}	698112	95.87

Table 6.12: $k = 4$

The sparsity patterns of the non-standard matrices for $k = 3$, $M = 8$ and $k = 4$, $M = 8$ are shown in Figures 6.7 and 6.8, respectively.

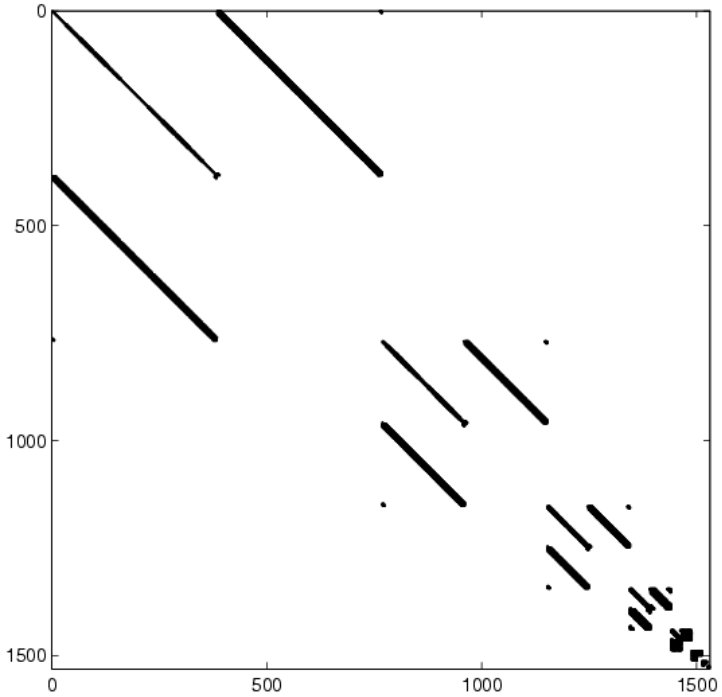


Figure 6.7: Non-standard matrix: $k = 3$, $M = 8$

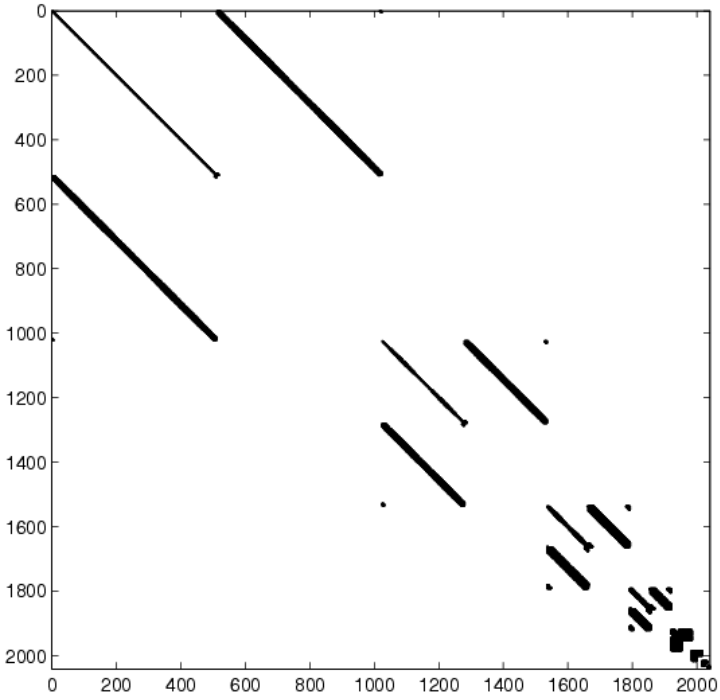


Figure 6.8: Non-standard matrix: $k = 4$, $M = 8$

Standard Results

In tables 6.13-6.16 the L_2 norm of the error of the compressed system is denoted by $\|u - \overline{u}_h\|$ and *nz* is the number of non-zero elements. The column $\%$ gives the percentage of the matrix entries that can be set to zero without any detrimental effect. Here, the CGS method was used to solve the linear system, and never required more than 14 iterations to converge.

M	$\|u - \overline{u}_h\|$	nz	%
4	9.5533×10^{-2}	30	88.28
5	5.4961×10^{-2}	46	95.51
6	2.8759×10^{-2}	114	97.22
7	1.6583×10^{-2}	234	98.57
8	8.8686×10^{-3}	454	99.31
9	4.1247×10^{-3}	1642	99.44
10	2.2959×10^{-3}	2054	99.80

Table 6.13: $k = 1$

M	$\|u - \overline{u}_h\|$	nz	%
4	5.1578×10^{-2}	54	94.73
5	1.9581×10^{-2}	116	97.17
6	9.3315×10^{-3}	192	98.83
7	2.0100×10^{-3}	486	99.26
8	6.0769×10^{-4}	1130	99.57
9	1.5951×10^{-4}	2038	99.81
10	4.2436×10^{-5}	4002	99.90

Table 6.14: $k = 2$

M	$\|u - \overline{u}_h\|$	nz	%
4	3.5131×10^{-2}	76	96.70
5	1.1436×10^{-2}	132	98.57
6	2.4208×10^{-3}	324	99.12
7	5.3742×10^{-4}	664	99.55
8	4.9299×10^{-5}	1918	99.67
9	9.7179×10^{-6}	3026	99.87
10	1.3187×10^{-6}	5854	99.94

Table 6.15: $k = 3$

M	$\|u - \overline{u}_h\|$	nz	%
4	2.6259×10^{-2}	84	97.93
5	3.7294×10^{-3}	240	98.54
6	1.3171×10^{-3}	372	99.43
7	6.9790×10^{-5}	950	99.64
8	1.3926×10^{-5}	1840	99.82
9	6.6913×10^{-7}	4098	99.90
10	4.8726×10^{-8}	8030	99.95

Table 6.16: $k = 4$

Tables 6.13-6.16 show that we achieve better compression as k and M increase. The sparsity patterns of the standard matrices for $k = 3$, $M = 8$ and $k = 4$, $M = 8$ are shown in Figures 6.9 and 6.10, respectively.

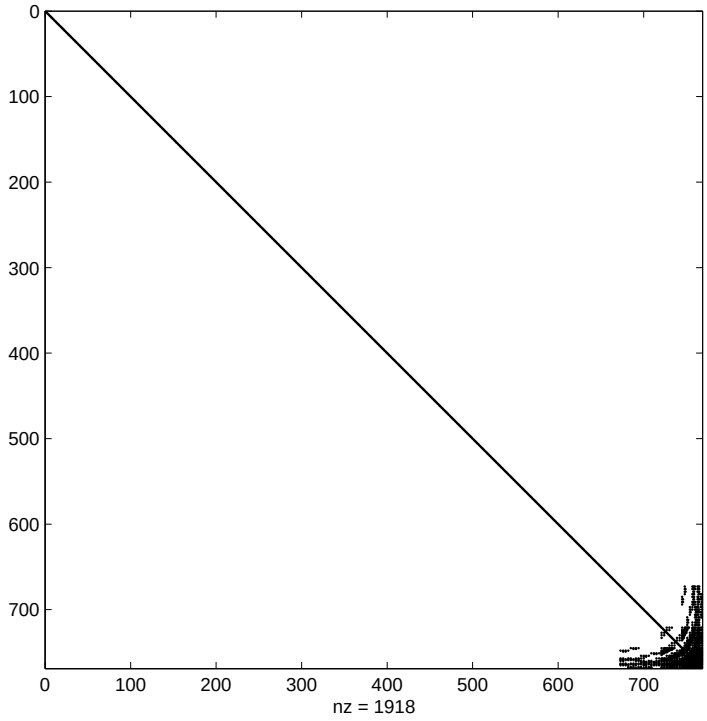


Figure 6.9: Standard matrix: $k = 3$, $M = 8$

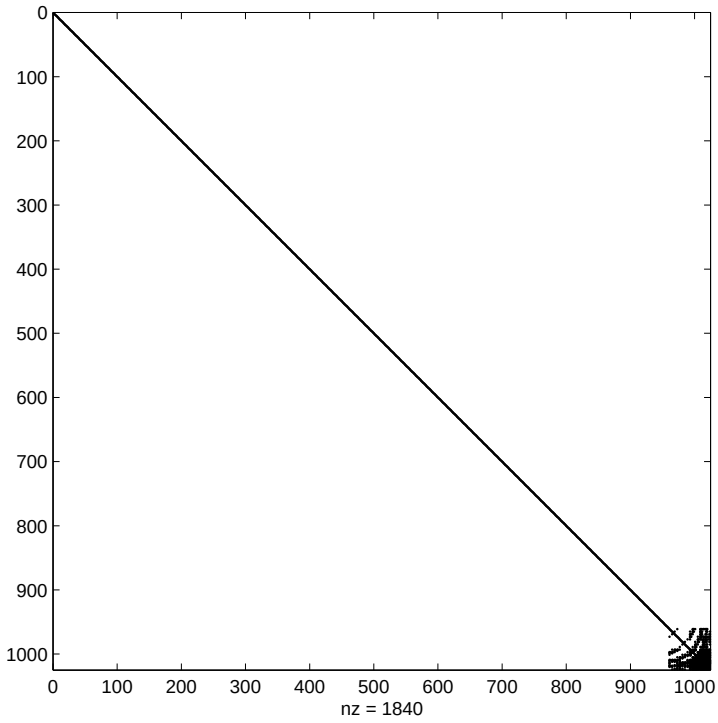


Figure 6.10: Standard matrix: $k = 4$, $M = 8$

6.2 The Dirichlet Problem

In this section we consider the solution of equation (6.0.2) with Dirichlet boundary conditions. We need to solve the first kind equation

$$\int_\Gamma G(\mathbf{p},\mathbf{q})\frac{\partial u(\mathbf{q})}{\partial n_{\mathbf{q}}}d\Gamma_{\mathbf{q}} = -\frac{1}{2}u(\mathbf{p}) + \int_\Gamma u(\mathbf{q})\frac{\partial G(\mathbf{p},\mathbf{q})}{\partial n_{\mathbf{q}}}d\Gamma_{\mathbf{q}}, \quad \mathbf{p}\in\Gamma, \tag{6.2.1}$$

for the unknown Neumann boundary condition $\frac{\partial u}{\partial n}$. Again, once $\frac{\partial u}{\partial n}$ is found on the boundary we use the integral representation (2.3.18) to find $\frac{\partial u}{\partial n}$ anywhere in the domain Ω_+.

We can show that the discretisation of an operator of order α, will in general have condi-

tion number $\mathcal{O}(N^{|\alpha|})$ and eigenvalues $\mathcal{O}(N^{\alpha})$. Therefore, since the single-layer operator $\mathcal{L}$ is of order -1, the eigenvalues of $A_M = \mathcal{O}(N^{-1})$ and $\kappa_2(A_M) = \mathcal{O}(N)$. Therefore, in order to use an iterative method efficiently we must precondition the matrix A_M.

6.2.1 Preconditioning

The convergence of Krylov subspace iterative methods is rapid if the matrix has a small condition number and the eigenvalues are clustered. If the original system matrix $\mathrm{A} \in \mathbb{C}^{N\times N}$ does not satisfy these conditions then it may be possible to find a preconditioner D such that $\mathrm{D}^{-1}\mathrm{A}$ has the desired properties. The Krylov subspace methods can then be applied to the preconditioned system

$$\mathrm{B}\mathbf{x} = \mathbf{y}, \tag{6.2.2}$$

where $\mathrm{B} = \mathrm{D}^{-1}\mathrm{A}$ and $\mathbf{y} = \mathrm{D}^{-1}\mathbf{b}$. Within conjugate gradient type methods the coefficient matrix is required only in matrix-vector products, therefore B is never explicitly formed. Suppose the matrix-vector product $\mathbf{z} = \mathrm{B}\mathbf{v}$ is required, where $\mathbf{v} \in \mathbb{C}^N$ is known. Then $\mathbf{z} = \mathrm{D}^{-1}\mathrm{A}\mathbf{v} = \mathrm{D}^{-1}\mathbf{t}$, where

$$\mathrm{D}\mathbf{z} = \mathbf{t}. \tag{6.2.3}$$

Hence to find $\mathbf{z}$, we first find $\mathbf{t} = \mathrm{A}\mathbf{v}$ and then solve (6.2.3).

Therefore, a good preconditioner D must satisfy two (often conflicting) requirements.

- Firstly, D must be a good approximation to A, that is the eigenvalues of $\mathrm{D}^{-1}\mathrm{A}$ should be clustered near 1. Therefore a conjugate gradient type algorithm applied to (6.2.2) should converge faster than for the original system.

- Secondly, the solution of (6.2.3) must be cheap.

6.2.2 Wavelet Preconditioning

Consider the pseudodifferential operator equation

$$\mathcal{A}u = f \tag{6.2.4}$$

for $A : H^s \to H^{s-\alpha}$. We denote its Galerkin discretisation by the biorthogonal wavelet basis by

$$A_M \mathbf{u}_h = \mathbf{f}_h. \tag{6.2.5}$$

In a recent paper by Dahmen (see also references within) [11] it has been proved that

$$\kappa_2(A_M) = \mathcal{O}(2^{|\alpha|M}) = \mathcal{O}(N)^{|\alpha|}. \tag{6.2.6}$$

Furthermore, they state and prove the following result:

The matrices

$$\mathrm{B}_M := \mathrm{D}^l A_M \mathrm{D}^l \tag{6.2.7}$$

where

$$\left(D^l\right)_{\lambda,\lambda'} = 2^{ml}\delta_{\lambda,\lambda'}, \tag{6.2.8}$$

have uniformly bounded condition numbers

$$\kappa_2(\mathrm{B}_M) = \|\mathrm{B}_M\| \left\|\mathrm{B}_M^{-1}\right\| = \mathcal{O}(1). \tag{6.2.9}$$

This implies that in place of solving

$$A_M \mathbf{u}_h = \mathbf{f}_h,$$

we solve

$$B_M \mathbf{x} = \mathbf{y},$$

where $\mathrm{D}^l \mathbf{x} = \mathbf{u}_h$ and $\mathrm{D}^{-l} \mathbf{y} = \mathbf{f}_h$. In the next section we wish to use this preconditioner with the multiwavelet basis.

6.2.3 Multiwavelet Preconditioning

The result in the previous section requires the biorthogonality of wavelet basis; our wavelets of course are orthogonal. The result above can be used to establish uniform boundedness of preconditioned matrices. However, we show that the natural extension we employ here can result insignificant improvement of the condition number. For the multiwavelet basis, $\Psi^M = \{\psi_\lambda |\ \lambda = \{k, m, l\}\}$, $m = M-1, \ldots, 0$ and $l = 0, \ldots, 2^m - 1$, with k *mother* wavelets the natural generalization of the preconditioner in section 6.2.2 is to use the natural $k \times k$ dimensional diagonal blocks. Let D_M be the matrix containing the diagonal block entries of

$$\mathrm{A}_M := \langle \mathcal{A} \mathbf{\Psi}^M, \mathbf{\Psi}^M \rangle^{\mathrm{T}}. \tag{6.2.10}$$

Then, we propose the use D_M as a preconditioner for A_M in the form below

$$\mathrm{B}_M := \mathrm{D}_M^{-\frac{1}{2}} \mathrm{A}_M \mathrm{D}_M^{-\frac{1}{2}}. \tag{6.2.11}$$

That is to say, in place of solving

$$A_M \mathbf{u}_h = \mathbf{f}_h$$

we solve

$$\mathrm{B}_M \mathbf{x} = \mathbf{y},$$

where $\mathrm{D}_M^{-\frac{1}{2}} \mathbf{x}_= \mathbf{u}_h$ and $\mathrm{D}_M^{\frac{1}{2}} \mathbf{y}_= \mathbf{f}_h$.

The condition number of A_M is $\mathcal{O}(N)$. Our numerical results show a big improvement with precondition systems, namely,

$$\|\mathrm{B}_M\| \left\|\mathrm{B}_M^{-1}\right\| = \mathcal{O}\left(\log^2 N\right). \tag{6.2.12}$$

6.2.4 Numerical Results

In this section we present numerical results for the Laplace problem exterior to an ellipse of circumference 4π, with major axis 2.5940936 and minor axis 1.2970468, centered at the origin. We consider a Dirichlet problem, equivalent to that generated by three interior point sources placed at $\mathbf{p}_0 = (1.445288, \frac{3}{5}\pi)$, $\mathbf{p}_1 = (2.264285, \frac{8}{5}\pi)$ and $\mathbf{p}_2 = (2.2478149, \frac{53}{50}\pi)$ with strengths 1, 1.3 and 2, respectively. In table 6.17, cond(A_M) is the condition number of the unpreconditioned matrix, 'unpre its' is the number of GMRES(10) iterations required, where GMRES(*l*) is the so-called "GMRES with restarts" after every *l* iterations, [52]. The column cond(B_M) gives the condition number of the preconditioned, 'pre its' is the number of GMRES(10) iterations required. The L_2 norm of the error of the compressed system is denoted by $\left\| \frac{\partial u}{\partial n_{\mathbf{q}}} - \left(\overline{\frac{\partial u}{\partial n_{\mathbf{q}}}} \right)_h \right\|$ and *nz* is the number of non-zero elements remaining after compressing the matrix. The column % gives the percentage of the matrix entries that can be set to zero without any detrimental effect. Our matrices are of size $k2^M$.

M	cond(A)	unpre its	cond(B_M)	pre its	$\left\| \frac{\partial u}{\partial n_{\mathbf{q}}} - \left(\overline{\frac{\partial u}{\partial n_{\mathbf{q}}}} \right)_h \right\|$	*nz*	%
3	51.0	30	7.2	22	7.810×10^{-2}	466	54.5
4	110.0	41	10.5	29	3.207×10^{-2}	1334	67.4
5	233.4	65	14.5	38	3.621×10^{-3}	3948	76.0
6	482.8	86	19.2	46	6.715×10^{-4}	12072	81.6
7	982.8	120	24.6	53	4.814×10^{-5}	28744	89.0
8	1983.2	164	30.7	70	2.948×10^{-6}	75030	92.8
9	3984.4	220	37.5	91	1.407×10^{-7}	223256	94.7
10	7986.9	250	45.0	111	7.299×10^{-9}	455060	97.3

Table 6.17: $k = 4$

Figure 6.11 shows the eigenvalues of the standard matrix A_M for $k = 4$, $M = 8$, before preconditioning. As expected the eigenvalues rapidly cluster about 0. In Figure 6.12, we show the eigenvalues of the preconditioned matrix B_M. In the preconditioned case, we see that the eigenvalues do not cluster as rapidly as in the unpreconditioned case, however, they are all away from zero.

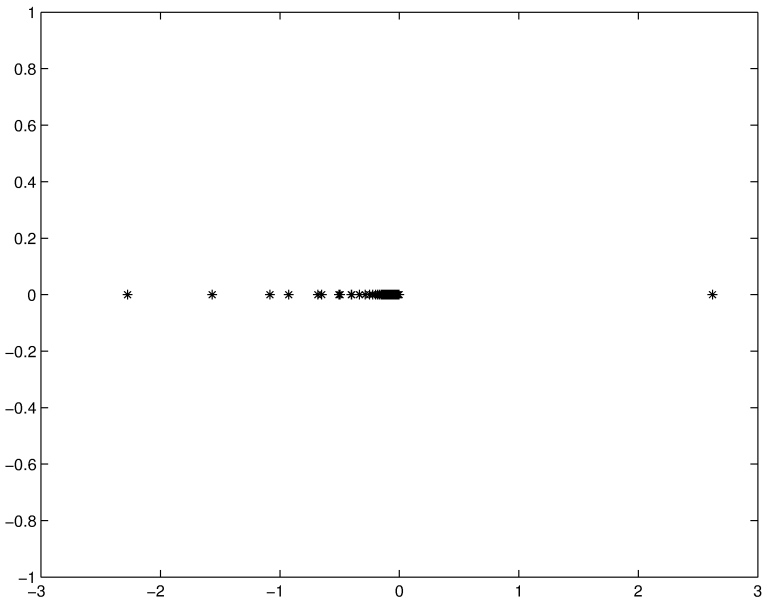


Figure 6.11: Eigenvalues of the unpreconditioned standard matrix for $k = 4$, $M = 8$

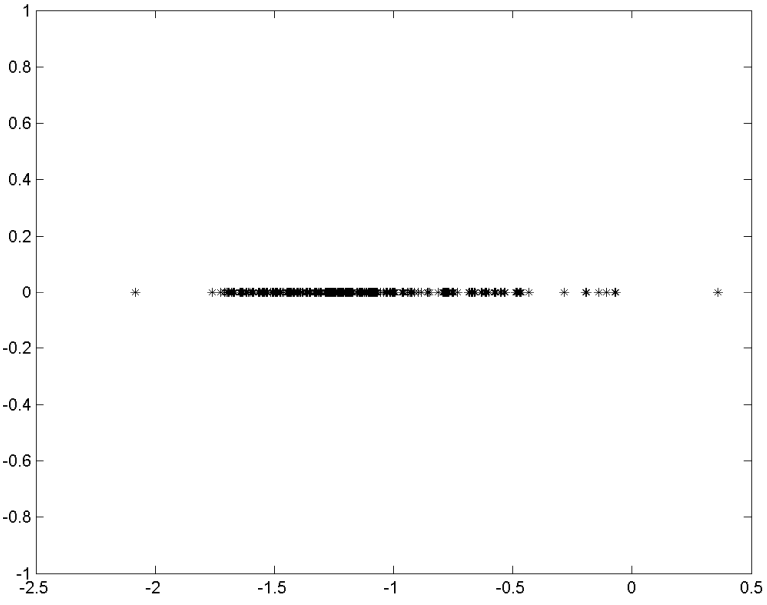


Figure 6.12: Eigenvalues of the preconditioned standard matrix for $k = 4$, $M = 8$

The sparsity pattern of the standard matrix for $k = 4$, $M = 8$ is shown in Figure 6.13.

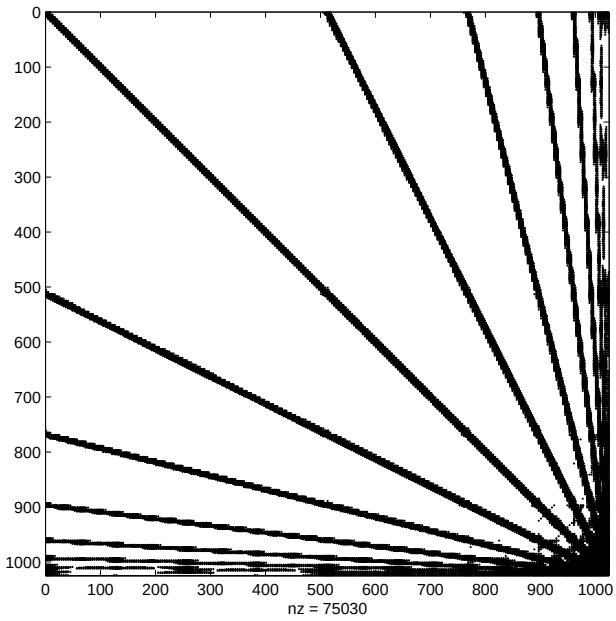


Figure 6.13: The standard matrix: $k = 4$, $M = 8$

Figure 6.14 shows the growth of the number of iterations for both the preconditioned and unpreconditioned methods using GMRES(10).

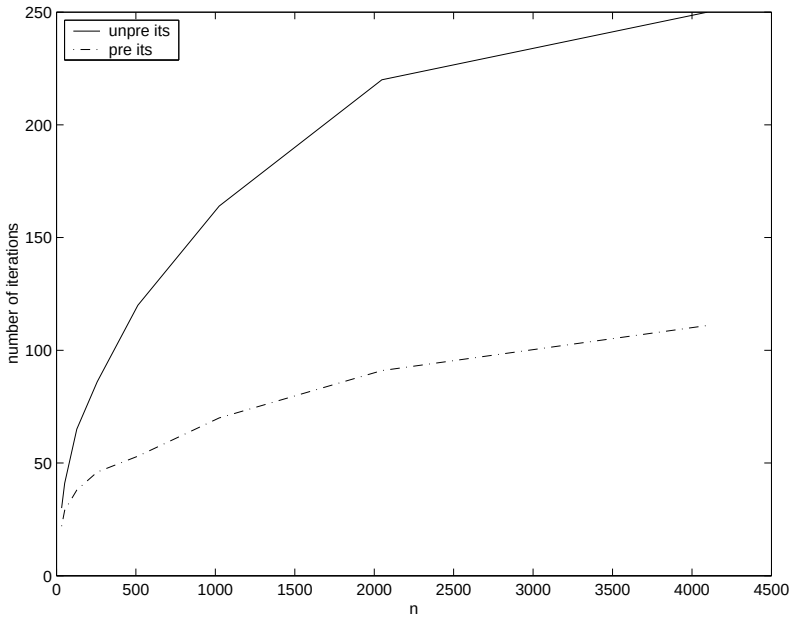


Figure 6.14: Iteration numbers for the preconditioned and unpreconditioned methods

Figure 6.15 shows the growth of the condition number for both the preconditioned and unpreconditioned methods. In Figure 6.16 we show the $\mathcal{O}\left(\log^2 N\right)$ growth of the condition number in the preconditioned case.

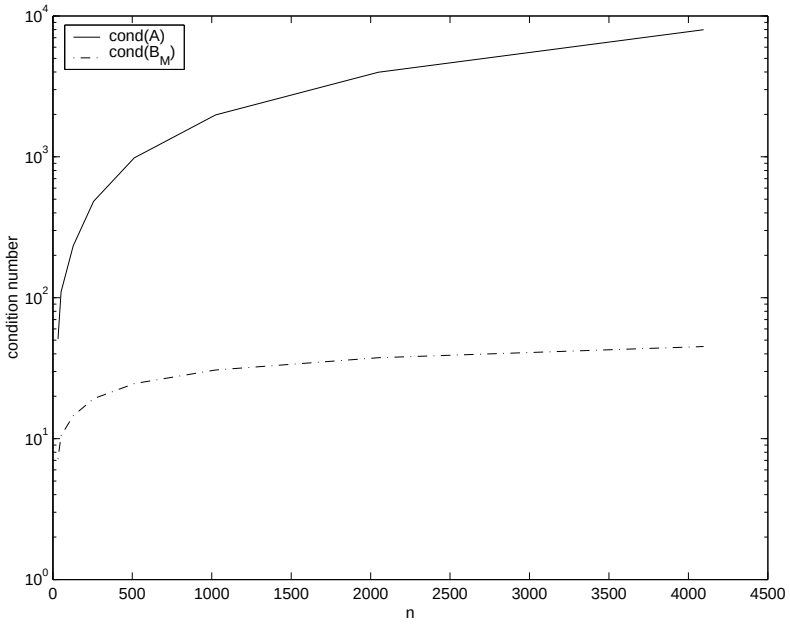


Figure 6.15: Condition numbers for both the preconditioned and unpreconditioned methods

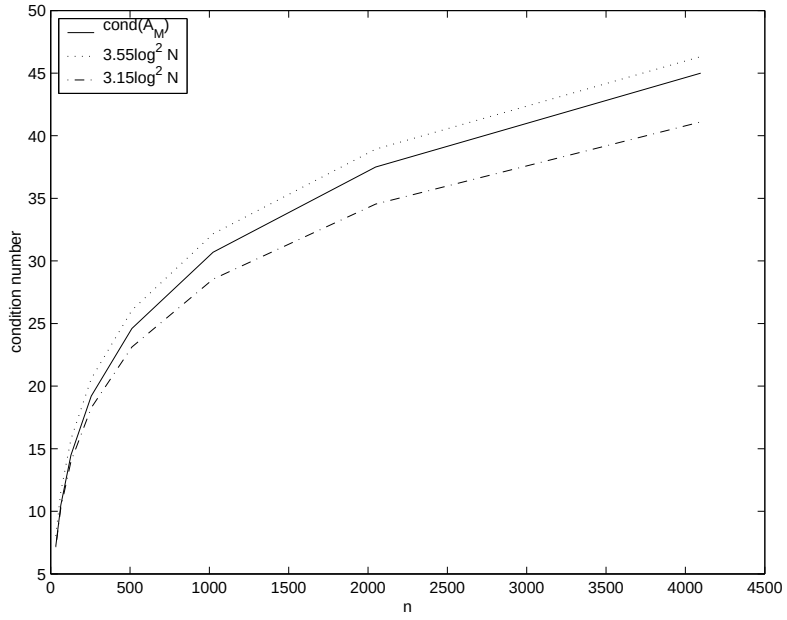


Figure 6.16: Condition number growth in the preconditioned case

Chapter Review

In this chapter we have presented the numerical results of two test problems for Laplace's equation with Neumann boundary conditions. Results for both the standard and non-standard methods have been obtained.

In section 6.2 we have considered Laplace's equation with Dirichlet boundary conditions. As the resulting coefficient matrix is ill-conditioned, we have discussed matrix preconditioning. In particular, we introduced the wavelet preconditioner suggested in [11]. Here, we have extended the use of the wavelet preconditioner to multiwavelet bases. We have presented numerical results, which show that our multiwavelet preconditioner reduces the growth of the matrix condition number to $\mathcal{O}(\log^2 N)$, and significantly reduces the number of GMRES(10) iterations required.

reduces the growth of the matrix condition number to $\mathcal{O}(\log^2 N)$, and significantly reduces the number of GMRES(10) iterations required.

Chapter 7

Conclusion and Further work

In this thesis, we have been concerned with the so-called wavelet algorithm for the solution of boundary integral equations. In chapter 2 we have briefly reviewed the methods and techniques required when partial differential equations are reformulated as boundary integral equations. In chapter 3, we discussed the *multiresolution* framework for wavelets, as well as, our choice of basis functions for this thesis, namely, the *multiwavelets* of [10].

In chapter 4, we developed the *standard* and *non-standard* Galerkin methods for multiwavelets. For both methods applied to operators of the standard analytical class, bounds are found for the size of matrix elements. Using these bounds compression strategies have been developed which only require the computation and storage of the significant matrix elements. We have shown that there are only $\mathcal{O}(N \log^p N)$ such significant elements.

In chapters 5 and 6 we have applied the standard and non-standard Galerkin methods to several test problems of varying "difficultly". In chapter 5, we concentrated on the radiosity problem of image synthesis, whereas, in chapter 6 we concentrated on the boundary integral reformulation of Laplace's equation. However, when we consider Laplace's equa-

tion with Dirichlet boundary conditions the resulting coefficient matrix is ill-conditioned. This is because the single-layer operator $\mathcal{L}$ is of order -1 and therefore its eigenvalues cluster at zero. Therefore, in order to use an iterative solver efficiently we precondition the linear system. We introduced the wavelet preconditioner suggested by Dahmen [11]. Then, we extend the preconditioner for use with multiwavelet basis functions. Our numerical results show that the multiwavelet preconditioner reduces the growth of the matrix condition number from $\mathcal{O}(N)$ to $\mathcal{O}\left(\log^2 N\right)$.

Many difficulties with the application of multiwavelets bases still remain. These include:

- The development of quadrature rules for the efficient numerical integration of multiwavelets over large supports.

- Further development of multiwavelet preconditioners to increase the clustering of eigenvalues and reduce further the $\mathcal{O}\left(\log^2 N\right)$ growth of the condition number.

- Due to the prevalence of collocation methods in the engineering community, the development of multiwavelet collocation methods, analogous to the standard and non-standard Galerkin methods.

Bibliography

[1] G. Chen and J. Zhou. *Boundary Element Methods*. Academic Press, London, 1992.

[2] P. W. Partridge, C. A. Brebbbia, and L. C. Wrobel. *The dual reciprocity boundary element method*. Elsevier, 1992.

[3] W. Hackbusch and Z. P. Nowak. On the fast matrix multiplication in the boundary element method by panel clustering. *Numerical Mathematics*, 54:436–491, 1989.

[4] A. T. J. Profit. *On fast multipole methods for the solution of the* H*elmholtz equation*. Ph.D. Thesis, Dept. of Maths., University of Salford, Salford, UK, 1999.

[5] S. Amini and A. T. J. Profit. Multi-level fast multipole solution of the scattering problem. *Engineering Analysis with Boundary Elements*, 27:547–546, 2003.

[6] G. Beylkin, R. Coifman, and V. Roklin. Fast wavelet transforms and numerical algorithms I. *Communications on Pure and Applied Mathematics*, 44:141–183, 1991.

[7] Peter Schröder. *Wavelet Algorithms for Illumination Computations*. Ph.D. Thesis, Princeton University, 1994.

[8] R. Wagner and W. Chew. A study of wavelets for the solution of electromagnetic integral equations. *IEEE Transactions on Antennas and Propagation*, 43(8):802–810, 1995.

[9] Christian Lage and Christoph Schwab. Wavelet Galerkin algorithms for boundary integral equations. *SIAM Journal of Scientific Computing*, 20(6):2195–2222, 1999.

[10] B. K. Alpert. *Sparse Representation of Smooth Linear Operators*. Ph.D. Thesis, Yale University, 1990.

[11] W. Dahmen. Wavelet and multiscale methods for operator equations. *Acta Numerica*, 6:55–228, 1997.

[12] R. Adams. *Sobolev Spaces*. Academic Press, 1975.

[13] W. McLean. *Strongly Elliptic Systems and Boundary Integral Equations*. Cambridge University Press, 2000.

[14] R. Kress. *Linear Integral Equations*, volume 82 of *Applied Mathematical Sciences*. Springer-Verlang, 1989.

[15] W. L. Wendland. Boundary element methods for elliptic problems. In A.H. Schatz, V. Thomée, and W. L. Wendland, editors, *Mathematical Theory of Finite and Boundary Element Methods*, pages 219–276. Birkhäuser Verlag, 1990.

[16] M. E. Taylor. *Pseudodifferential Operators and Nonlinear PDE*. Birkhäuser, 1991.

[17] Xavier Saint Raymond. *Elementary Introduction to the Theory of Pseudodifferential Operators*. Studies in Advanced Mathematics. CRC Press, 1991.

[18] Ram P. Kanwal. *Linear Integral Equations: theory and technique*. Academic Press, second edition, 1997.

[19] K. Atkinson. *A Survey of numerical methods for the solution of Fredholm integral equations*. Society for Industrial and Applied Mathematics, 1976.

[20] D.N. Arnold and W.L. Wendland. The convergence of spline collocation for strongly elliptic equations on curves. *Numerical Mathematics*, 47:317–341, 1985.

[21] C. K. Chui, editor. *Wavelets: A Tutorial in Theory and Applications*. Academic Press, 1992.

[22] M. B. Ruskai et al. *Wavelets and their applications*. Jones and Bartlet Publishers, 1992.

[23] G. G. Walter and X. Shen. *Wavelets and Other Orthogonal Systems*. Chapman and Hall/CRC, 2001.

[24] L. L. Schumaker and G. Webb, editors. *Topics in the Theory and Applications of Wavelets*. Academic Press, 1994.

[25] S. Jaffard and Ph. Laurençot. *Wavelets - A Tutorial in Theory and Applications*, volume 2 of *Wavelet Analysis and its applications*, chapter Orthonormal Wavelets, Analysis of Operators and Applications, pages 543–602. Academic Press, 1992.

[26] I. Daubechies. *Ten Lectures on Wavelets*, volume 61. CBMS-NSF Regional Conference Series in Applied Mathematics, 1992.

[27] A. Cohen, I. Daubechies, and J. C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 45:485–560, 1992.

[28] W. Dahmen, A. Kunoth, and K. Urban. Biorthogonal spline-wavelets on the interval - stability and moment conditions. Technical Report 129, RWTH Aachen, 1996.

[29] J. M. Carnicer, W. Dahmen, and J. M. Peña. Local decomposition of refinable spaces. *Applied and Computational Harmonic Analysis*, 3(2):127–153, 1996.

[30] W. Dahmen. Some remarks on multiscale transformations, stability and biorthogonality. In *Wavelets, Images and Surface Fitting*, pages 157–188. AK Peters Ltd, 1994.

[31] P. Auscher. Wavelets with boundary conditions on the interval. In C. K. Chui, editor, *Wavelets: A Tutorial in Theory and Applications*, pages 217–236. Academic Press, 1992.

[32] L. Anderson, N. Hall, B. Jawerth, and G. Peters. Wavelets on closed subsets of the real line. In L. L. Schumaker and G. Webb, editors, *Topics in the Theory and Applications of Wavelets*, pages 1–61. Academic Press, 1994.

[33] C. K. Chui and E. Quak. Wavelets on a bounded interval. In D. Braess and L. L. Schumaker, editors, *Numerical Methods of Approximation Theory*, pages 53–75. Birkhäuser, 1992.

[34] A. Cohen, I. Daubechies, and P. Vial. Wavelets on the interval and fast wavelet transforms. *Applied and Computational Harmonic Analysis*, 1:54–81, 1993.

[35] W. Dahmen, S. Prossdorf, and R. Schneider. Wavelet approximation methods for pseudodifferential equations II: Matrix compression and fast solution. *Advances in Computational Mathematics*, 1:259–335, 1993.

[36] C. K. Chui. *An introduction to Wavelets*. Academic Press, 1992.

[37] Stefano Pistillo. Multiwavelet solution of boundary integral equations - application to the radiosity problem. M.sc., University of Salford, 1997.

[38] H. F. Buchera, L. C. Wrobel, W. J. Mansur, and C. Magluta. A fast solution technique for large BEM systems of equations using block wavelet transforms. In *Advances in Boundary Element Techniques IV*, pages 307–313, 2003.

[39] S. Amini and S. P. Nixon. A comparison between collocation and Galerkin multiwavelet solution of Laplace's equation. In *Fourth UK Conference on Boundary Integral Methods*, pages 254–263, 2003.

[40] S. Amini, R. Elmazuzi, and S. P. Nixon. Multiwavelet collocation boundary element solution of Laplace's equation. In *International Conference on Computational & Experimental Engineering and Sciences*, 2004.

[41] S. Amini, E. A. Chadwick, S. P. Nixon, and S. Pistillo. Multiwavelet solution of boundary integral equations - applications to the radiosity problem. In *Third UK Conference on Boundary Integral Methods*, pages 99–108, 2001.

[42] S. Amini and S. P. Nixon. Multiwavelet Galerkin boundary integral method for Laplace's equation. In *Fourth UK Conference on Boundary Integral Methods*, pages 127–136, 2003.

[43] S. Amini and S. P. Nixon. Multiwavelet boundary integral solution of Laplace's equation. In *Advances in Boundary Element Techniques IV*, pages 295–300, 2003.

[44] H. Harbrecht, M. Konik, and R. Schneider. Fully discrete wavelet Galerkin schemes. *Engineering Analysis with Boundary Elements*, 27:423–437, 2003.

[45] W. Dahmen, H. Harbrecht, and R. Schneider. Compression techniques for boundary integral equations - optimal complexity estimates. Preprint, Tecnische Universität Chemnitz, SFB393/02-06, 2002.

[46] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore and London, 1991.

[47] R. W. Freund, G. H. Golub, and N. M. Nachtigal. Iterative solution of linear systems. *Acta Numerica*, 1:57–100, 1992.

[48] M. F. Cohen and J. R. Wallace. *Radiosity and Realistic Image Synthesis*. Academic Press, 1993.

[49] Francois X. Sillion and Claude Puech. *Radiosity and Global Illumination*. Morgan Kaufmann Publishers, 1994.

[50] Steven J. Gortler, Peter Schröder, Michael F. Cohen, and Pat Hanrahan. Wavelet radiosity. *Computer Graphics (SIGGRAPH '93 Proceedings)*, 27(4):pp. 221–230, August 1993.

[51] Sumanta N. Pattanaik and Kadi Bouatouch. Fast wavelet radiosity method. *Eurographics*, 13(3), 1994.

[52] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.

[53] S. Amini, S. P. Nixon, and P. R. Rajaguru. Block diagonal preconditioning for multiwavelet Galerkin boundary integral solution of Laplace's equation. In *Advances in Boundary Element Techniques V*, 2004.