KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Science

Introducing Redundancy into Numerical Computations

Computing with Frames

Roel Matthysen

Supervisor:
Prof. dr. ir. D. Huybrechs

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Computer Science

March 2018

Introducing Redundancy into Numerical Computations

Computing with Frames

Roel MATTHYSEN

Examination committee:
Prof. dr. ir. H. Hens, chair
Prof. dr. ir. D. Huybrechs, supervisor
Prof. dr. ir. D. Nuyens
Prof. dr. ir. M. Van Barel
Prof. dr. S. Olver
(Imperial College, London)
Prof. dr. M. Lyon
(University of New Hampshire)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Computer Science

March 2018

Preface

This thesis is about computing with frames, particular sets of functions that lend themselves very well to approximation problems. For certain frames, such as the Fourier extension frame, we show spectral properties that lead to efficient approximation algorithms. We then demonstrate the flexible nature of these algorithms as a building block for more complicated problems.

At the end of this research project I have many people to thank, this would not have been possible without their help and support. I first want to thank the members of the jury for their careful reading of this manuscript and their insightful questions and comments.

I want to thank my supervisor Daan for his infectious enthusiasm for frames and their wonderful properties. It is what initially convinced me to start working on this topic, and the ever growing frames group in our department shows this enthusiasm is still very much alive. I am also very grateful that his taste in road trip music turned out to be compatible with mine.

During these past years I was fortunate enough to be able to attend some of the most interesting conferences in this field. I want to thank the friendly and welcoming RBF and Approximation Theory communities for making these conferences fun as well as informative. A special thanks to Ben Adcock, Mark Lyon, Alex Townsend, Grady Wright, Jose-Luis Romero and Cécile Piret for their helpful advice and discussions.

Many thanks to the people I've shared an office with over these years: Peter, who helped me over administrative hurdles by conveniently clearing them a few weeks ahead of me, and Marcus, who supplemented his encouragement with self-brewed beer. There is also Marcus' predecessor Sam, who was always ready to provide a slightly too honest opinion on the student projects we supervised together. A special thanks to the people in the frames meetings: Daan, Marcus, Vincent and Andrew. When we're not getting carried away by wild conjectures there is some real progress every once in a while, and I am very proud some

of that builds on this thesis. As for the many other people in the department I've gotten to know over the years: thanks for the good company on PhDays weekends, the after work football games, and creating such a pleasant working environment.

Dank aan al mijn vrienden en familie, en vooral mijn ouders, voor wiens standbeeld er al een plaatsje in onze tuin wordt gezocht. Bedankt Karen, voor je geduld met mijn chronisch ontoereikende planning, en voor je peptalks als ik het even niet zag zitten. En last but not least is er mijn (voorlopig) kleinste supporter. Tijdens het schrijven van deze tekst was ze al op de achtergrond aanwezig, om dan met geweldig gevoel voor timing vlak na de preliminaire verdediging ten tonele te verschijnen. Dankzij haar was de stress van het schrijven en de verdediging altijd relatief.

Voor Ellie

Abstract

Function approximation is a basic building block in numerical analysis. A general function may hide much of its properties, by being only available through point evaluations or indirect measurements. However, for certain families of functions such as polynomials or trigonometric functions much more is known: explicit integrals and derivatives, symmetries, efficient discrete representations etc. The goal of this thesis is to come up with combinations of such known functions that approximate a given function very well, so the approximation is a very flexible mathematical object.

Many approximation schemes can be iteratively refined, by increasing the *degrees of freedom*, the number of functions used in the approximation. As the scheme is refined, we look for convergence, and minimal complexity. The convergence rate expresses the improvement in accuracy with increasing degrees of freedom. Fast convergence is not achievable for all functions, leading to convergence results for different classes of functions. On the other hand, the complexity expresses the computational costs involved. We want the cost to scale reasonably with the degrees of freedom.

This thesis focuses on computing approximations that are capable of yielding very high accuracy very quickly. Specifically, we cover approximation of functions in a frame, which is an extension of the concept of an orthonormal basis. Orthonormal bases are very convenient choices for the function family, and much effort has been put into efficient schemes using them to compute approximations. Frames on the other hand are a more recent development. Their use in approximation problems is therefore relatively new. The main advantage compared to orthonormal bases is the flexibility: there are many problems where an orthonormal basis is hard to find, or has suboptimal properties, but where a suitable frame can easily be constructed.

The main contributions in this thesis are useful algorithms to compute frame approximations, for classes of frames satisfying a condition on the

spectrum of their evaluation operator. We formulate algorithms, both one- and multidimensional, to compute good approximations in the frame in a least squares sense. These improve upon the complexity of straightforward least squares implementations. We also explain how these algorithms can be a building block for more general problems involving least squares approximations, including boundary value problems and approximations in more general frames. We briefly describe the design choices involved in implementing these algorithms, implementations that are available as an open source Julia package. We end with an overview of current research topics in numerical frame approximations that relate to the algorithms in this thesis.

Beknopte samenvatting

Het benaderen van functies is een fundamenteel onderdeel van de numerieke analyse. De eigenschappen van een generieke functie zijn vaak moeilijk te doorgronden indien de functie enkel geobserveerd kan worden in discrete punten of door indirecte metingen. Daartegenover staan families van functies zoals veeltermen of trigonometrische functies die al eeuwenlang bestudeerd worden en waarvan de integralen, afgeleiden, symmetrieën, efficiënte transformaties etc. gekend zijn. Het doel van deze thesis is om een gegeven functie te benaderen met elementen uit zo een familie, zodat de resulterende benadering een flexibel wiskundig object is met gekende eigenschappen.

Veel benaderingsschema's kunnen iteratief verfijnd worden door het aantal *vrijheidsgraden* op te voeren, het aantal functies dat gebruikt wordt in de benadering. Bij een stijgend aantal vrijheidsgraden verwachten we dat de fout daalt, en dat de rekenkost voor het algoritme waarmee we de benadering berekenen binnen de perken blijft. De snelheid waarmee de benaderingsfout daalt hangt af van de gegeven functie, deze is kwalitatief verschillend voor verschillende klassen van functies.

Deze thesis gaat over benaderingsschema's die zeer snel zeer nauwkeurige benaderingen kunnen berekenen. Meer specifiek bestaat de familie van functies waaruit we elementen gebruiken uit een frame, een generalisatie van een orthonormale basis. Orthonormale basissen zijn klassieke en zeer goede keuzes voor benaderingsproblemen, met een lange geschiedenis van theoretische en praktische resultaten. Frames zijn een recentere ontwikkeling, en relatief onbekend in benaderingsproblemen. Hun grote voordeel ten opzichte van orthonormale basissen zijn de flexibiliteit: er zijn problemen waarvoor een orthonormale basis zeer moeilijk te vinden is, of onbruikbaar is in algoritmes, terwijl een frame voor hetzelfde probleem eenvoudig te vinden is.

De voornaamste bijdragen van deze thesis zijn bruikbare algoritmes om benaderingen in frames te berekenen, voor frames waarvan het spectrum van de

evaluatiematrix voldoet aan bepaalde voorwaarden. We formuleren algoritmes in één en meerdere dimensies om benaderingen met een kleine kleinste kwadraten fout te berekenen. Deze algoritmes hebben een fundamenteel lagere rekenkost dan een voor de hand liggende aanpak. Daarnaast brengen we een aantal problemen aan waarvoor deze algoritmes een bouwsteen kunnen vormen, zoals randwaardeproblemen, en ruimere klassen frames. We beschrijven kort de genomen ontwerpbeslissingen bij het implementeren van deze algoritmes, deze zijn vrij verkrijgbaar als een open source JULIA pakket. We eindigen met een overzicht van toekomstig werk en openstaande problemen gerelateerd aan deze thesis.

List of Abbreviations

DCT Discrete Cosine Transform. 21, 28, 50

DFT Discrete Fourier Transform. 27, 35, 37

DPSS Discrete Prolate Spheroidal Sequence. ix, 33, 34, 37, 53, 62

DPSWF Discrete Prolate Spheroidal Wave Function. ix, 33, 35, 36

FC Fourier Continuation. 40

FE Fourier Extension. xviii, 17, 18, 25, 30, 34, 35, 39, 41–47, 51, 53, 54, 59, 60, 62, 69, 73, 75, 106, 118, 126, 128, 129, 137, 139

FFT Fast Fourier Transform. xvii, xviii, 21, 26–28, 58, 106, 125, 126, 128, 131

P-DPSS Periodic Discrete Prolate Spheroidal Sequence. ix, 35–37, 39, 51, 53, 56, 57, 60, 75–78, 137

PSWF Prolate Spheroidal Wave Function. ix, 29, 31, 32, 34–36, 39, 40

RBF Radial Basis Function. 102, 113

SVD Singular Value Decomposition. 16, 28, 35, 39, 49–51, 59, 70, 71, 78, 115, 126

T-SVD Truncated Singular Value Decomposition. xi, 16, 21, 22, 29, 41, 51, 52, 70, 73, 78, 101, 115–118

List of Symbols

Function Approximation

$\{\phi_i\}_{i=1}^{\infty}$ Set of functions

I_N Index set with N indices

$\mathcal{H}, \mathcal{H}_N$ Hilbert space, Hilbert space span$(\{\phi_i\}_{I_N})$

$\mathcal{P}, \mathcal{P}_N$ Orthogonal projection on $\mathcal{H}$, $\mathcal{H}_N$.

$\mathcal{P}^\tau, \mathcal{P}_N^\tau$ Regularised projection on $\mathcal{H}$, $\mathcal{H}_N$ with cutoff parameter τ.

$\mathcal{L}^2$ Space of square integrable functions.

ℓ^2 Space of square integrable sequences.

$\mathcal{T}^*, \mathcal{T}_N^*$ Analysis operator for $\{\phi_i\}_{i=1}^{\infty}$, $\{\phi_i\}_{I_N}$

$\mathcal{T}, \mathcal{T}_N$ Synthesis operator for $\{\phi_i\}_{i=1}^{\infty}$, $\{\phi_i\}_{I_N}$

Prolate Spheroidal Wave Functions

$\eta(\tau, N_\Lambda)$ Number of singular values between $1-\tau$ and τ for N_Λ degrees of freedom

$\mathcal{D}_\Omega, D_\Omega$ Restriction operator to Ω, infinite and finite-dimensional

$\mathcal{B}_\Lambda, B_\Lambda$ Bandlimiting operator to Λ, infinite and finite-dimensional

ϑ Prolate Spheroidal Wave Function

Ψ Discrete Prolate Spheroidal Wave Function

ψ Discrete Prolate Spheroidal Sequence

φ Periodic Discrete Prolate Spheroidal Sequence

Time Domain

Ω	Domain
P_Ω	Set of Points in domain
I_R	Index set of bounding box
N_R	Number of points in the bounding box grid
n_R	Number of grid points along a single dimension
I_Ω	Index set of Points in domain
S_Ω	Tiling of Points in domain
N_Ω	Number of Points in domain
$\delta\Omega$	Domain boundary
$P_{\delta\Omega}$	Set of Points in domain boundary
$N_{\delta\Omega}$	Number of Points in domain boundary
R	Bounding box
P_R	Set of points in bounding box

Frequency Domain

Λ	Frequency domain
P_Λ	Set of points in frequency domain
N_Λ	Number of points in frequency domain
$\hat{R}$	Bounding box
$P_{\hat{R}}$	Set of points in bounding box
$N_{\hat{R}}$	Number of points in the bounding box grid
n_Λ	Number of grid points along a single dimension

Algorithms

G_{N_Λ}	Truncated Gram matrix
A	Oversampled collocation matrix
ϱ	Oversampling factor

τ Truncation parameter for the T-SVD.

$U_\alpha, S_\alpha, V_\alpha$ Singular vectors and values for which $\sigma > 1 - \tau$

$U_\gamma, S_\gamma, V_\gamma$ Singular vectors and values for which $\tau \geq \sigma$

$U_\beta, S_\beta, V_\beta$ Singular vectors and values for which $1 - \tau \geq \sigma > \tau$

T Ratio R/Ω

Contents

List of Figures

Chapter 1

Introduction

The central problem of this thesis is the approximation of functions in a frame. In order to understand frames as a generalisation of orthonormal bases we recall the most important approximation properties regarding these bases. Then we present some examples of frames like the Fourier extension frame and augmented frames, and why they are the natural choice in some situations. We also cover the most important recent results in frame approximation theory, most importantly regarding regularised projections. These provide error estimates based on the existence of a good approximation in the frame, serving as a motivation for using frames. We will cover the efficient computation of these regularised projections in Chapters 3 and 4, for the classes of frames described in Chapter 2.

Throughout this chapter, the different types of bases and frames are illustrated with simple examples. Through these examples, the most important characteristics of the different approximations are easily shown: accuracy and convergence speed, computational complexity, and flexibility. By no means do these examples cover this diverse topic, but they serve to contextualise the frame approximations.

To keep this chapter concise, theorems and statements are provided without proof. The classical results are found in textbooks on approximation or spectral methods such as [100, 10], and books on frame theory [26, 25], where most of the frame-specific notation and definitions originate. Near the end of the chapter the approximation properties of frames are the result of more recent work [1, 3].

1.1 Function Approximation

The functions we approximate are mappings of the form

$$f : \Omega \to \mathbb{C},$$

where $\Omega \in \mathbb{R}^d$, a mapping that takes elements of d-dimensional Euclidean space to complex numbers. We assume f is in some normed linear vector space V with norm $\|\cdot\| : V \to \mathbb{R}^+$. For an approximation $\tilde{f}$, we seek to minimise $\|f - \tilde{f}\|$.

If the vector space V has an inner product $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{C}$, and is complete with respect to the induced distance metric, it is called a Hilbert space, denoted by $\mathcal{H}$. This inner product is conjugate symmetric, linear in the first element and the inner product of an element with itself is positive. This last property induces the norm $\|f\|^2 = \langle f, f \rangle$.

The starting point of our approximations is the representation of f by a linear combination of functions from an ordered function sequence $\{\phi_i\}_{i=1}^{\infty}$:

$$f(x) = \sum_{i=1}^{\infty} c_i \phi_i(x) \tag{1.1}$$

If for all $f \in \mathcal{H}$ coefficients c_i exist so that $\|f - \sum_{i=1}^{\infty} c_i \phi_i\| = 0$ we say $\{\phi_i\}_{i=1}^{\infty}$ has the *expansion property*. This representation leads to the sequence of approximations

$$f_N(x) = \sum_{i=1}^{N} c_i \phi_i(x), \qquad N = 1, 2, \ldots \tag{1.2}$$

Because N coefficients can be chosen independently, the approximation f_N is said to have N *degrees of freedom*. This sequence is said to converge to f if

$$\lim_{N \to \infty} \|f - f_N\| = 0. \tag{1.3}$$

Function sequences that have a unique approximation for every element of $\mathcal{H}$ are called (Schauder) bases.

Definition 1.1. *A sequence of functions $\{\phi_i\}_{i=1}^{\infty}$ in $\mathcal{H}$ is a Schauder basis for $\mathcal{H}$ if, for each $f \in \mathcal{H}$, there exist unique scalar coefficients $\{c_i(f)\}_{i=1}^{\infty}$ such that*

$$f = \sum_{i=1}^{\infty} c_i(f) \phi_i. \tag{1.4}$$

(1.4) should be understood as convergence as in (1.3). Note that this may depend on the ordering of $\{\phi_i\}_{i=1}^{\infty}$. If convergence is unconditional, i. e. independent of this ordering, $\{\phi_i\}_{i=1}^{\infty}$ is called an *unconditional* basis. In practice we truncate (1.1) based on a series of index sets $I_N \subset I_{N+1} \subset \ldots$, with shorthand notation $\{\phi_i\}_{I_N} = \{\phi_i\}_{i \in I_N}$. Convergence should be understood in this ordering.

The uniqueness of the coefficients requires some notion of independence of $\{\phi_i\}_{i=1}^{\infty}$. A N-dimensional set $\{\phi_i\}_{I_N}$ is linearly independent if

$$\sum_{i \in I_N} c_i \phi_i = 0 \Leftrightarrow c_i = 0 \quad \forall i.$$

In infinite-dimensional spaces, there are different ways of defining independence.

Definition 1.2. *If $\{\phi_i\}_{i=1}^{\infty}$ is a sequence in $\mathcal{H}$, then*

1. *$\{\phi_i\}_{i=1}^{\infty}$ is linearly independent if every finite subset of $\{\phi_i\}_{i=1}^{\infty}$ is linearly independent.*
2. *$\{\phi_i\}_{i=1}^{\infty}$ is ω-independent if $\sum_{i=1}^{\infty} c_i \phi_i = 0$ implies $c_i = 0, \forall i$.*

It is clear that for the coefficients c_i in (1.1) to be unique, $\{\phi_i\}_{i=1}^{\infty}$ should be both linearly and ω-independent. The distinction between these types of independence will play an important role when defining frames in §1.4.

In practice, the function set is finite, and we look at the successive truncations of $\{\phi_i\}_{i=1}^{\infty}$ to N elements. If $\{\phi_i\}_{i=1}^{\infty}$ is a linearly independent basis, every truncation is a linearly independent basis for its span; denote span by $\mathcal{H}_N = \text{span}(\{\phi_i\}_{I_N})$. The best approximation to f in $\mathcal{H}_N$ exists and is uniquely defined by the orthogonal projection of f on $\mathcal{H}_N$.

Definition 1.3. *Let $\mathcal{H}$ be some Hilbert space, and $\mathcal{H}_N$ be a complete linear subspace. Then $\mathcal{P}_N$ is an orthogonal projection onto $\mathcal{H}_N$ if and only if for all $f, g \in \mathcal{H}$,*

$$\langle \mathcal{P}_N f, g \rangle = \langle \mathcal{P}_N f, \mathcal{P}_N g \rangle = \langle f, \mathcal{P}_N g \rangle.$$

So far, we have established that (1.1) has a unique solution if $\{\phi_i\}_{i=1}^{\infty}$ is a basis, and that for such a basis the best approximation (1.2) in a subspace is given by the orthogonal projection onto that subspace. However, it remains unclear how to choose $\{\phi_i\}_{i=1}^{\infty}$, and how to compute the successive approximations. The following sections give concrete examples of $\{\phi_i\}_{i=1}^{\infty}$ and accompanying constructions for $\mathcal{P}_N f$.

Before continuing, we mention the two most common Hilbert spaces throughout this chapter. First, the space of square integrable functions on Ω:

Definition 1.4. *$\mathcal{L}^2_\Omega$ contains all functions f for which*

$$\int_\Omega |f(x)|^2 dx < \infty,$$

with inner product

$$\langle f, g\rangle = \int_\Omega f(x)\overline{g(x)}dx.$$

Second, the closely related space of square integrable sequences.

Definition 1.5. *ℓ^2 contains all sequences $\{c_k\}_{k=1}^\infty$ for which*

$$\sum_{k=1}^\infty |c_k|^2 < \infty, \tag{1.5}$$

with inner product

$$\langle c, d\rangle = \sum_{k=1}^\infty c_k\overline{d_k}.$$

Unless specified otherwise, $\langle\cdot,\cdot,\rangle$ and $\|\cdot\|$ should be understood as this inner product and norm for functions and sequences respectively.

1.2 Orthonormal Bases

Projecting an element of $\mathcal{H}$ onto a sequence of spaces spanned by basis elements is substantially easier if the basis has orthogonal elements.

Definition 1.6. *A basis sequence $\{\phi_i\}_{i=1}^\infty$ is orthonormal if*

$$\langle\phi_i, \phi_j\rangle = \delta_{ij}, \qquad i, j = 1, \ldots, \infty$$

with $\delta_{ij} = 1$ if $i = j$ and 0 otherwise.

If a Hilbert space has a countable orthonormal basis, like $\mathcal{L}^2$ and ℓ^2, it is called separable. For an orthonormal basis

$$f = \sum_{i=1}^\infty \langle f, \phi_i\rangle\phi_i.$$

$$\mathcal{P}_N f = \sum_{i\in I_N} \langle f, \phi_i\rangle\phi_i(x).$$

To see this, note that

$$\langle f, \phi_i \rangle = \langle \mathcal{P}_N f, \phi_i \rangle \qquad \forall i \in I_N.$$

Then

$$\langle \mathcal{P}_N f, \phi_j \rangle = \sum_{i \in I_N} c_i \langle \phi_i, \phi_j \rangle = c_i \qquad \forall i \in I_N.$$

The norm of the coefficients $c_i = \langle f, \phi_i \rangle$ is related to the norm of f through the Parseval identity

$$\forall f \in \mathcal{H} : \sum |\langle f, \phi_i \rangle|^2 = \|f\|^2. \tag{1.6}$$

Now define the *analysis operator* for an orthonormal basis $\{\phi_i\}_{i=1}^{\infty}$ as

$$\mathcal{T}^* : f \to \{\langle f, \phi_i \rangle\}. \tag{1.7}$$

Through Parseval's identity this operator is an isometric isomorphism from $\mathcal{L}^2$ to ℓ^2. The adjoint operator is called the *synthesis* operator:

$$\mathcal{T} : \{c_i\} \to \sum c_i \phi_i, \tag{1.8}$$

an isometric isomorphism from ℓ^2 to $\mathcal{L}^2$. Their composition $\mathcal{T}\mathcal{T}^*$ is the identity operator $\mathcal{I}$. More generally, an orthonormal basis for a Hilbert space can always be identified with ℓ^2, and as such all orthonormal bases for $\mathcal{H}$ are equivalent up to unitary transformation [25, Theorem 3.2.7]. In what follows the analysis and synthesis operators for $\mathcal{H}_N$ are denoted by $\mathcal{T}_N^*$ and $\mathcal{T}_N$ respectively.

A classical result sometimes attributed to Riesz and Fischer states that for an orthonormal basis $\|f - \mathcal{P}_N f\|$ converges as $N \to \infty$. Equivalently, for every $\epsilon > 0$ there exists an N so that $\|f - \mathcal{P}_N f\| < \epsilon$. The speed of this convergence for classes of f is an important measure for the effectiveness of an approximation scheme. To discern between these we define three qualitatively different convergence speeds

Definition 1.7. *A sequence $\{s_n\}_{n \in \mathbb{N}}$ decays with order k if*

$$s_n = \mathcal{O}(n^{-k}), \qquad n \gg 1$$

Definition 1.8. *A sequence $\{s_n\}_{n \in \mathbb{N}}$ decays superalgebraically if*

$$\forall k : s_n = o(n^{-k}), \qquad n \gg 1 \tag{1.9}$$

Definition 1.9. *A sequence $\{s_n\}_{n \in \mathbb{N}}$ decays exponentially if*

$$s_n = \mathcal{O}(\mathrm{e}^{cn}) \tag{1.10}$$

for some constant c.

An example of superalgebraic decay that is not exponential is so-called *root-exponential decay*, where

$$s_n = \mathcal{O}(\mathrm{e}^{c\sqrt{n}}).$$

An approximation is said to converge at a certain algebraic or exponential rate if the residual $\|f - f_N\|$ decays at this rate. Due to the Parseval identity the decay rate of the orthogonal projection is directly related to decay of the coefficients c_i.

$$\|f - \mathcal{P}_N f\|^2 = \sum_{i \notin I_N} |c_i|^2$$

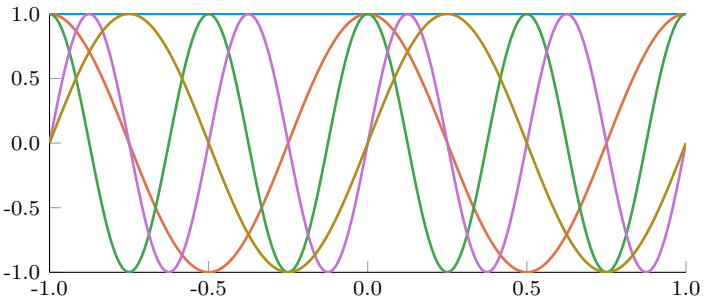
For example, if the coefficients for successive truncations decay exponentially, then

$$\begin{aligned}\|f - \mathcal{P}_N f\| &\leq \sqrt{C^2 \sum_{i \geq N} \rho^{2N}} \\ &= C \frac{\rho^N}{\sqrt{1-\rho^2}}.\end{aligned}$$

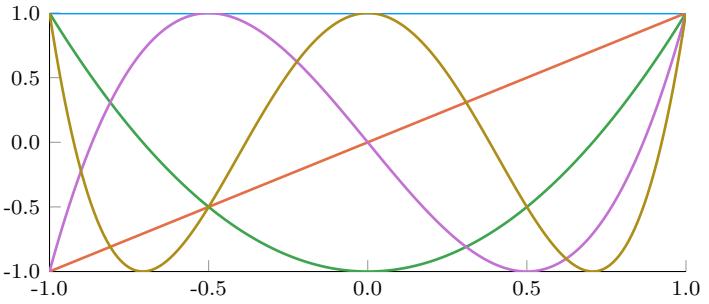
Exponential convergence of coefficients thus directly implies exponential convergence of $\mathcal{P}_N f$ to f in the norm, at some rate. Algebraic decay of coefficients with rate k similarly leads to algebraic convergence results at slightly lower rates [10, Sec. 2.12]. More precisely, if the coefficients decay as $\mathcal{O}(n^{-k})$, adding the tail of the expansion yields $\mathcal{O}(n^{-k+1})$ pointwise convergence and $\mathcal{O}(n^{-k+1/2})$ convergence in the $\mathcal{L}^2$ norm.

Remark 1.10. We only very briefly touch upon the topic of convergence of sequences in Hilbert spaces in this chapter, a subject that should be treated with care. However, if the synthesis operator $\mathcal{T}$ for the sequence is a bounded operator from ℓ^2 to $\mathcal{L}^2$, then $\{\phi_i\}_{i=1}^{\infty}$ is called a *Bessel sequence*. For a Bessel sequence, $\sum_{k=1}^{\infty} c_i \phi_i$ converges unconditionally for all $\{c_i\} \in \ell^2$, i. e. independent of any reordering [25, Corollary 3.1.5]. All orthonormal bases, Riesz bases and frames are Bessel sequences.

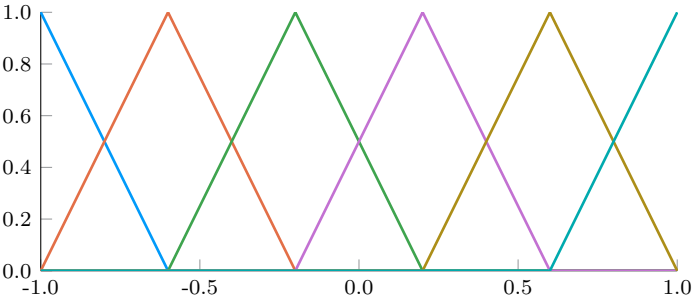
The practical conclusion is that the orthogonal projection $\mathcal{P}_N$ on $\mathcal{H}_N$ can be constructed by calculating the inner products $\langle f, \phi_i \rangle$. These integrals can sometimes be evaluated analytically, but are often approximated numerically. We will return to this in §1.5. The next paragraphs will cover some often used function sets $\{\phi_i\}_{i=1}^{\infty}$, illustrated in Fig. 1.1.



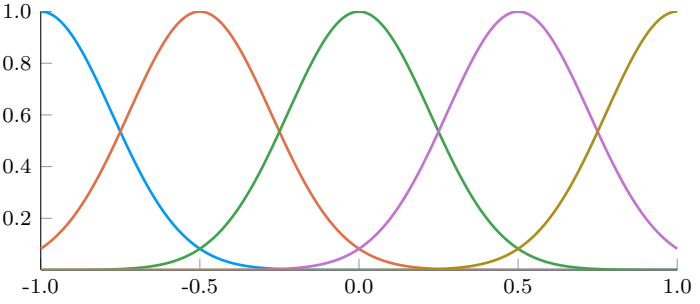
(a) Fourier Series



(b) Chebyshev Polynomials



(c) Piecewise polynomials



(d) Radial Basis Functions

Figure 1.1: $\{\phi_i\}_{i\in I_5}$ for some frequently used bases.

1.2.1 Fourier Series

A classical orthonormal basis for $\mathcal{L}^2_{[a,b]}$, the square integrable functions on the interval $[a,b]$, is the Fourier basis

$$\{\phi_k\}_{k=-\infty}^{\infty}, \quad \phi_k = \mathrm{e}^{\frac{\mathrm{i}k2\pi x}{b-a}}$$

The resulting expansion is the Fourier series

$$f = \sum_{k=-\infty}^{\infty} \hat{f}[k]\mathrm{e}^{\frac{\mathrm{i}k2\pi x}{b-a}},$$

where the coefficients are the inner products with the basis functions

$$\hat{f}[k] = \int_a^b f(x)\mathrm{e}^{-\frac{\mathrm{i}k2\pi x}{b-a}}\,dx. \tag{1.11}$$

Due to the basis set taking both positive and negative indices, we define the index set as

$$I_N = \begin{cases} -\frac{N-1}{2}, \ldots, \frac{N-1}{2} & \text{for } N \quad \text{odd} \\ -\frac{N}{2}+1, \ldots, \frac{N}{2} & \text{for } N \quad \text{even} \end{cases}. \tag{1.12}$$

Figure 1.1 shows the equivalent representation in sines and cosines.

The convergence of Fourier series is dependent on the smoothness and periodicity of the function f, and can be estimated through integration by parts. A classical result states that if f is $p-1$ times continuously differentiable, with the original function and each derivative periodic on $[a,b]$, and the derivative of order p is of bounded variation on $[a,b]$, then

$$|\hat{f}[k]| = \mathcal{O}(|k|^{-p-1}), \qquad k \gg 1 \tag{1.13}$$

If $f \in C^\infty$ and periodic, the coefficients decay superalgebraically. For analytic f, i. e. the Taylor series at each point in $[a,b]$ converges at least in a small neighborhood of that point, decay is exponential. The rate is determined by the location of the singularities in the complex plane.

Theorem 1.11. *[10, Theorem 5] Let z_i be the singularities in the complex plane of f, and denote by*

$$\rho = \min_i |\,\Im(z_i)|$$

the minimum distance of a singularity to the real line. Then

$$\hat{f}[k] = \mathcal{O}(\rho^{-k}).$$

When f is not periodic on $[a, b]$, the continuity requirements for (1.13) are not fulfilled. The coefficients only converge as $\mathcal{O}(|k|^{-1})$. For example, for the function $f(x) = x$ on $[-1, 1]$, some approximations are shown in Fig. 1.2. The approximation converges pointwise almost everywhere, with the exception of the discontinuity, where it converges to zero. However, the overshoot present in the oscillations near the discontinuity does not decay. In fact

$$\lim_{N\to\infty} \mathcal{P}_N f(-1 + \frac{1}{N}) = -1 - 2\dot{(}0.089489872236\ldots).$$

This persistent overshoot near discontinuous functions is known as the *Gibbs phenomenon* [47, 43].

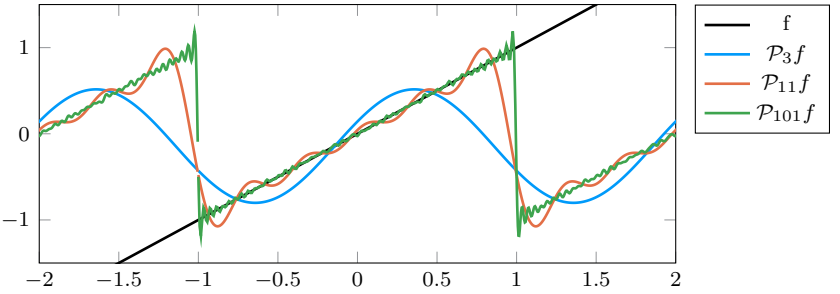


Figure 1.2: Fourier series of $f(x) = x$ for increasing degrees of freedom. The occurrence of the Gibbs phenomenon means the peak of the oscillation near the discontinuity never decreases in magnitude.

The requirement of periodicity makes the Fourier series and related methods unsuitable for non-periodic problems if fast convergence is required.

1.2.2 Orthogonal polynomials

As variants of $\mathcal{L}^2$, we can look at the weighted spaces $\mathcal{L}^2_{[-1,1],\omega}$. For these spaces the inner product is taken with respect to some positive weight function $\omega(x)$:

$$\langle f, g\rangle_{[-1,1],\omega} = \int_{-1}^{1} f(x)g(x)\omega(x)dx.$$

For each of these spaces, there exists a family of orthogonal polynomials. When normalised these constitute an orthonormal basis for $\mathcal{L}^2_{[-1,1],\omega}$. The orthogonal polynomials resulting from $\omega(x) = \left(\sqrt{1-x^2}^{-1}\right)$ are the Chebyshev polynomials.

Denote the Chebyshev polynomial of order k by $T_k(x)$, and the expansion by

$$f(x) = \sum_{k=0}^{\infty} c_k T_k(x).$$

Through the mapping

$$T_k(\cos\theta) = \cos k\theta \tag{1.14}$$

a Chebyshev series is related to a Fourier cosine series. Unsurprisingly, the convergence of the expansion is governed by similar principles.

Theorem 1.12. *[100, Theorem 7.1] Let $f, f', \ldots, f^{(p-1)}$ be absolutely continuous for some $p \geq 1$, and let $f^{(p)}$ be a function of bounded variation. Then*

$$|c_k| = \mathcal{O}(k^{-p-1}), k \gg 1. \tag{1.15}$$

For analytic functions, the exponential decay rate is determined by the singularities, where the distance is measured in terms of ellipses.

Theorem 1.13. *[100, Theorem 8.1] If f is analytic and bounded in the Bernstein ellipse with foci -1 and 1 with semimajor and semiminor axis lengths summing to ρ, then*

$$\|f - \mathcal{P}_N f\| = \mathcal{O}(\rho^{-N}), \qquad N \to \infty.$$

1.2.3 Higher dimensions and Tensor product bases

Consider $\Omega \subset \mathbb{R}^d$ when $d > 1$. Although orthonormal bases necessarily exist for any $\mathcal{L}^2_\Omega$, they are generally not obvious. As an example, the eigenfunctions of the Laplace operator $\Delta f = \lambda f$ on Ω form an orthonormal basis for $\mathcal{L}^2_\Omega$, though they can be hard to compute with.

On the other hand, certain Ω's do lend themselves to orthonormal bases, when there is structure to be exploited. For example, when Ω is a square $[-1,1] \times [-1,1]$, a tensor product basis

$$\{(x,y) \to \phi_i(x)\phi_j(y)\}, \qquad 1 \leq i \leq N_x, 1 \leq j \leq N_y$$

is orthonormal in $\mathcal{L}^2_\Omega$ if $\{\phi_i\}_{i=1}^{\infty}$ is an orthonormal basis for $\mathcal{L}^2_{[-1,1]}$. For some other domains Ω bases are known explicitly, such as the spherical harmonics for the n-sphere [7].

1.3 Non-Orthogonal Bases

The previous sections have demonstrated the desirable properties of orthonormal bases. Orthogonal projection simply follows from the definitions, and convergence is in general only limited by the smoothness of the approximant. Despite the advantages, this approach may not always be the best choice. For example:

- The approximant might not be very smooth, or maybe even discontinuous.
- As in §1.2.3, an orthogonal basis $\{\phi_i\}_{i=1}^{\infty}$ might be hard to come up with or compute with.
- The bases might need to satisfy additional requirements.

This section covers more general bases than the orthonormal ones. For these bases the representation

$$f = \sum \langle f, \phi_i \rangle \phi_i \tag{1.16}$$

may not hold in general. Define the analysis and synthesis operator as before. If the basis is a Bessel sequence, the composition of $\mathcal{T}^*$ and $\mathcal{T}$ is a bounded operator from ℓ^2 to ℓ^2. In matrix representation it is referred to as the *Gram* matrix and given by

$$\mathcal{G} = \mathcal{T}^*\mathcal{T} = \{\langle \phi_i, \phi_j \rangle\}_{i=1,j=1}^{\infty}. \tag{1.17}$$

Note that for $f = \sum_{i=1}^{\infty} c_i \phi_i$,

$$\mathcal{G}\{c_i\}_{i=1}^{\infty} = \{\langle f, \phi_j \rangle\}_{j=1}^{\infty}.$$

Thus, the operator $\mathcal{G}$ maps the coefficients to the inner product with the basis functions. If this operator is invertible, inverting it is then a way of obtaining the coefficients from $\{\langle f, \phi_k \rangle\}_{k=1}^{\infty}$. Note that for orthonormal bases $\mathcal{G}$ is the identity operator, leading immediately to (1.16). Two example non-orthogonal bases are shown in Figs. 1.1c and 1.1d.

1.3.1 B-splines

Given a set of points $x_j \in [a, b]$ a possible representation is that of a polynomial of degree n on each interval $[x_{j-1}, x_j]$. By carefully choosing the coefficients, continuity up to a certain degree k can be enforced at the knots so the resulting

expansion is in $C^k_{[a,b]}$. An example is the representation in B-splines

$$f_N(x) = \sum_{j=1}^{N} c_j B_{j,n}(x).$$

When the knots are distinct, this expansion is continuous up to the derivative of degree $n-1$. The total degrees of freedom N is equal to the number of knots plus the number of overlapping B-splines. The B-splines themselves are defined through the Cox-de Boor recursion formula [28]:

$$B_{i,0}(x) := \begin{cases} 1 & \text{if} \quad x_i \leq x < x_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{1.18}$$

$$B_{i,k}(x) := \frac{x - x_i}{x_{i+k} - x_i} B_{i,k-1}(x) + \frac{x_{i+k+1} - x}{x_{i+k+1} - x_{i+1}} B_{i,k-1}(x). \tag{1.19}$$

Figure 1.1c shows the B-splines of degree 1 in equispaced nodes, which are tent functions. The resulting approximation is a piecewise linear polynomial. Localised bases such as this one have the property that $\langle \phi_i, \phi_j \rangle$ is zero for most $i \neq j$. Thus the Gram operator (1.17) is an infinite but sparse matrix, a major advantage when solving linear systems involving (truncations of) this matrix.

1.3.2 Radial Basis Functions

Radial basis functions [18] are defined based on a (1-dimensional) shape function $\phi(r)$ and a collection of 'centers' $\boldsymbol{x}_i$:

$$\phi_i(\boldsymbol{x}) = \phi(\|\boldsymbol{x} - \boldsymbol{x}_i\|).$$

Well-known shape functions are the Gaussian $\phi(r) = \mathrm{e}^{(\epsilon r)^2}$ and the multiquadric $\phi(r) = \sqrt{1 + (\epsilon r)^2}$. These all have a shape parameter ϵ, that influences the width of the radial basis functions. There usually is a trade-off involving this shape parameter between accuracy and conditioning of the related system. For example, very narrow Gaussians are localised and lead to an approximately sparse Gram matrix, but poor approximations away from the centers. On the other hand, very wide Gaussians may have better approximations but are nearly linearly dependent [39].

As with the piecewise polynomials, there is a lot of freedom in the location of the centers. Figure 1.1d shows Gaussian radial basis functions on equispaced centers in one dimension. Because the function depends only on the distance metric, radial basis functions are easy to generalise to higher dimensions. Radial

basis functions are known to be particularly efficient for domains without boundaries, such as the sphere [41]. Therefore they are a popular choice in weather forecasting and related applications for the earth surface [38].

1.3.3 Riesz Bases

In §1.3 we defined the Gram operator $\mathcal{G} : \ell^2 \to \ell^2 : \{c_k\} \to \{\langle f, \phi_k \rangle\}_{k=1}^{\infty}$. Bases for which $\mathcal{G}$ is invertible are called *Riesz bases*.

Definition 1.14. *A function set $\{\phi_i\}_{i=1}^{\infty}$ is a Riesz basis for $\mathcal{H}$ if its span is dense in $\mathcal{H}$ and the relaxed Parseval identity*

$$A\|c\|^2 \leq \|\sum_{i=1}^{\infty} c_i \phi_i\|^2 \leq B\|c\|^2, \qquad \forall c \in \ell^2 \tag{1.20}$$

holds for positive constants A and B.

The norm of the Gram inverse is given by the optimal lower Riesz bound $\|\mathcal{G}^{-1}\| = A^{-1}$ [26, Proposition 3.6.8]. Note that the definition ensures the ϕ_i are not linearly dependent or ω-dependent, since $\sum_{i=1}^{\infty} c_i \phi_i = 0$ for some nonzero c would imply $A = 0$. Furthermore, (1.20) implies that a Riesz basis is a Schauder basis as in Definition 1.1, so the representation

$$f = \sum_i c_i \phi_i(x)$$

is unique. For general bases, the orthogonal projection on a subspace $\mathcal{H}_N$ results from

$$\langle f, \phi \rangle = \langle \mathcal{P}_N f, \phi \rangle, \qquad \forall \phi \in \Phi_N,$$

reformulated using the gram operator as

$$\mathcal{G}\{c_k\}_{k=1}^{N} = \{\langle f, \phi_k \rangle\}_{k=1}^{N}.$$

Finding the coefficients for the orthogonal projection thus corresponds to inverting the truncated Gram matrix

$$\mathcal{G}_N = \{\mathcal{G}_{i,j}\}, \qquad i, j = 1, \ldots, N.$$

The difficulty of inverting here depends on the choice of basis.

It should be noted that any finite subset of a linearly independent set is a Riesz basis for its span. In the examples of B-splines and radial basis functions above, take A_N as the optimal lower Riesz bound for $\{\phi_i\}_{I_N}$. Since the basis functions

are linearly independent, $A_N > 0$ and $\mathcal{G}_N$ is invertible. However, A_N may be arbitrarily small, unless $\{\phi_i\}_{i=1}^{\infty}$ forms a Riesz Basis with lower bound A, then $\inf A_n := A$ [26, Proposition 6.1.2]. Put differently, the Riesz basis property ensures some level of conditioning for the problem of finding the coefficients from the inner products with the basis functions.

Considerable interest has been given to *coherent* systems arising from a generating function ψ and a variable operator applied to it. Examples are the translating operator, dilation operator and modulation operator

$$T_a\psi(x) = \psi(x-a),$$

$$D_b\psi(x) = \psi(x/b)\sqrt{|b|}^{-1},$$

$$E_c\psi(x) = \psi(x)\mathrm{e}^{\mathrm{i}2\pi cx}.$$

Composing a generating function with modulation and translation leads to Gabor systems $\{E_cT_a\psi\}_{a\in S_a, c\in S_c}$; translation with dilation leads to wavelet systems $\{T_aD_b\psi\}_{a\in S_a, c\in S_c}$. Much effort has been put into determining conditions on ψ so that these systems are orthonormal. However, the conditions under which these systems are Riesz bases or frames are much more general, leading to more freedom in the choice of ψ and index sets. See [25, Chapters 9, 11] for more in-depth discussions.

1.4 Frames

Frames, the main topic of this thesis, generalise the concept of a basis. Introduced by Duffin and Schaeffer [33], they satisfy a generalised Parseval identity, called the *frame condition*.

Definition 1.15. *A sequence $\{\phi_i\}_{i=1}^{\infty}$ in $\mathcal{H}$ is a frame for $\mathcal{H}$ if there exists constants $A, B > 0$ such that*

$$A\|f\|^2 \leq \sum_{i=1}^{\infty} |\langle f, \phi_i\rangle|^2 \leq B\|f\|^2, \qquad \forall f \in \mathcal{H}. \tag{1.21}$$

The optimal constants A, B so that (1.21) holds are called the *frame bounds*. Note that every Riesz basis is a frame since (1.20) implies (1.21). However, the reverse is not true. In particular the frame definition allows the existence of infinite sequences $c \in \ell^2$ for which $\sum_{i=1}^{\infty} c_i\phi_i = 0$. The difference is that frames can be ω-dependent.

Subclassifications include tight frames, and exact frames. A frame is tight if for the optimal frame bounds $A = B$. A frame for $\mathcal{H}$ is exact if it is no longer a frame for this space when any one element is removed. A frame is exact if and only if it is a Riesz basis [26, Theorem 5.5.4], so the term frame is mostly used in the context of inexact frames.

The Gram operator is defined for frames as in (1.17), based on the analysis and synthesis operators. Note that, as opposed to Riesz bases, a frame does not guarantee an invertible Gram operator. In fact the spectrum of $\mathcal{G}$ consists of $\{0\} \cup [A, B]$. A related operator is the *frame operator*

$$\mathcal{S} = \mathcal{T}\mathcal{T}^* : \mathcal{H} \to \mathcal{H}, \quad f \to \sum_{k=1}^{\infty} \langle f, \phi_k \rangle \phi_k.$$

For an orthonormal basis $\mathcal{S}_N f$ converges to f, $\mathcal{S}_N = \mathcal{T}_N \mathcal{T}_N^*$, but for a Riesz basis or frame this is not necessarily true. However, it is always possible to find a so-called *dual frame*.

Definition 1.16. *A frame $\{\psi_i\}_{i=1}^{\infty}$ is a dual frame for $\{\phi_i\}_{i=1}^{\infty}$ if*

$$f = \sum_{k=1}^{\infty} \langle f, \psi_i \rangle \phi_i = \sum_{k=1}^{\infty} \langle f, \phi_i \rangle \psi_i, \qquad \forall f \in \mathcal{H}$$

By this definition, an orthonormal basis is its own dual. The dual frame is unique if and only if the frame is a Riesz basis. For a general frame a unique *canonical dual frame* $\{\psi_i\}_{i=1}^{\infty} = \mathcal{S}^{-1}\{\phi_i\}_{i=1}^{\infty}$ can be identified through inverting the frame operator. Then

$$f = \sum_{k=1}^{\infty} \langle f, \mathcal{S}^{-1}\phi_k \rangle \phi_k$$

always converges. Inverting the frame operator can be cumbersome since it is an infinite-dimensional operator. However, for a tight frame $\mathcal{S} = A\mathcal{I} = B\mathcal{I}$, a tight frame is self-dual up to a scaling. The canonical dual frame has the additional property that it leads to the smallest ℓ^2-norm of the coefficients, out of all possible dual frames.

So in fact, it is possible to get a converging sequence of approximations in a general frame

$$f_N = \sum_{k=1}^{N} \langle f, \mathcal{S}^{-1}\phi_k \rangle \phi_k$$

by truncating the canonical dual frame expansion. Much of the frame research has focused on either analytically identifying the canonical dual frame, or

numerically inverting it [27, 19, 24, 23]. However, in general $f_N \neq \mathcal{P}_N f$, so this truncated expansion is not the best approximation in the subspaces $\mathcal{H}_N$. It may even converge much slower, illustrated in §1.4.2 through the Fourier extension frame. At the same time this is an example of a linearly independent frame that is not a Riesz basis.

1.4.1 Regularizing orthogonal projections

The orthogonal projection on $\mathcal{H}_N$ can be computed as before through the truncated Gram matrix $\mathcal{G}_N$. In contrast to the frame operator $\mathcal{S}_N = \mathcal{T}_N \mathcal{T}_N^*$, this is equivalent to the identity operator only if the frame is an orthonormal system, so tightness is not sufficient.

The orthogonal projection coefficients c are determined by

$$\mathcal{G}_N c = \{\langle f, \phi_i \rangle\}_{I_N}. \tag{1.22}$$

The right hand side $\{\langle f, \phi_i \rangle\}_{I_N}$ is shortened to b in what follows for convenience. The truncated Gram matrices are bounded by the upper frame bound, $\|G_N\| = B_N < B$. However, the inverse $\|G_N^{-1}\|^{-1} = A_N$ only has lower bound greater than zero if the frame is a Riesz basis [1, Lemma 4.2], as in §1.3.3. This means the truncated Gram matrix can be very ill-conditioned, a phenomenon that will come up throughout the following chapters. Consequently, the exact orthogonal projection may be difficult to compute. Indeed, it is possible to construct functions $f, \|f\| = 1$ for which $\|c\|$ in (1.22) grows as $\mathcal{O}(A_N^{-1})$ [1, Proposition 5.1]. This growth can even be exponential, depending on the frame. A solution is to regularise the solution $\|c\|$ through the Singular Value Decomposition (SVD)

$$\mathcal{G}_N = U \Sigma V^*,$$

where U and V are unitary and $\Sigma = \operatorname{diag}(\sigma_1, \dots, \sigma_N)$ is a diagonal matrix containing the singular values in decreasing order of magnitude. The solution to $\mathcal{G}_N c = b$ is then given by $c = V \Sigma^{-1} U^* b$. Given a tolerance τ, define the Truncated Singular Value Decomposition (T-SVD) as

$$G_N^\tau = U \Sigma_\tau V^*,$$

where $\Sigma_\tau = \operatorname{diag}(\sigma_1, \dots, \sigma_{n_\tau}, 0, \dots)$ with n_τ defined by $\sigma_{n_\tau} \geq \tau > \sigma_{n_\tau + 1}$. Then define the regularised projection as

$$c^\tau = V \Sigma_\tau^\dagger U^* b, \tag{1.23}$$

$$\mathcal{P}_N^\tau f = \sum_{I_N} c_i^\tau \phi_i. \tag{1.24}$$

Here $\Sigma_\tau^\dagger = \operatorname{diag}(\sigma_1^{-1}, \dots, \sigma_{n_\tau}^{-1}, 0, \dots)$ is the *pseudo-inverse* of Σ_τ. For this truncated projection, the following convergence result holds

Theorem 1.17. *[1, Theorem 5.3]*

$$\|f - \mathcal{P}_N^\tau f\| \leq \|f - \mathcal{T}_N z\| + \sqrt{\tau}\|z\|, \quad \forall z \in \mathbb{C}^N, f \in \mathcal{H}. \tag{1.25}$$

For the orthogonal projection $\mathcal{P}f$ the bound $\|f - \mathcal{P}f\| \leq \|f - \mathcal{T}_N z\|$ necessarily holds for all $z \in \mathbb{C}^N$. The convergence of the truncated projection to the best approximation is thus dependent on the existence of good approximations with small coefficient norm $\|z\|$. However, the achievable precision guaranteed by the theorem is bounded. If $\tau = \varepsilon_{\text{mach}}$, only $\sqrt{\varepsilon_{\text{mach}}}$ precision can be expected.

1.4.2 Fourier extension frame

As in §1.2.1, consider the complex exponentials

$$\Phi_N = \left\{ \frac{1}{\sqrt{b-a}} \mathrm{e}^{\frac{in2\pi x}{b-a}} \right\}_{I_N}, \tag{1.26}$$

with I_N as in (1.12). The set Φ_∞ is an orthonormal basis for $\mathcal{L}^2_{[a,b]}$, and tensor products of these are orthonormal bases for $\mathcal{L}^2$ over a hypercube $R = (a,b)^d$. Now consider the space $\mathcal{L}^2_\Omega$, $\Omega \subset R$, with inner product

$$\langle f, g \rangle_\Omega = \int_\Omega f(\boldsymbol{x}) g(\boldsymbol{x}) d\boldsymbol{x}$$

and associated norm $\|\cdot\|_\Omega$. Then any function that is square integrable in R is square integrable in Ω.

Φ_∞ is not an orthogonal basis for $\mathcal{L}^2_\Omega$, as $\langle \cdot, \cdot \rangle_R \neq \langle \cdot, \cdot \rangle_\Omega$ in general. It is, however, a tight frame for $\mathcal{L}^2_\Omega$, with frame bounds $A = B = 1$ (see e. g. [25, Example 5.5.5]). It is also linearly independent. However, Φ_∞ is ω-dependent. Any nonzero function g that is supported only on $R \setminus \Omega$ and vanishes smoothly at the boundary has

$$\|g\|_R > 0, \qquad \|g\|_\Omega = \|f\|_\Omega.$$

Thus, for such g and any $f \in \mathcal{L}^2_\Omega$, $\|f - g\|_\Omega = 0$. An illustration can be found in Fig. 1.3, where several functions in $\mathcal{L}^2_{[-2,2]}$ are shown, that are equal on $(-1, 1)$.

This is the origin of the Fourier Extension (FE) name: the extension to R

$$f_{\text{ext}}(\boldsymbol{x}) = \sum_{i=1}^{\infty} c_i \phi_i(\boldsymbol{x}), \qquad \boldsymbol{x} \in R.$$

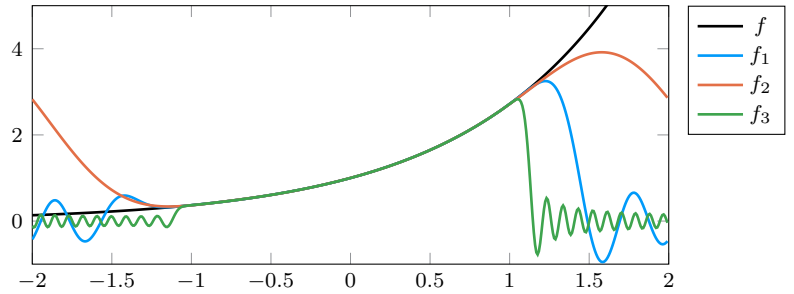


Figure 1.3: Different representations f_1, f_2, f_3 of $f(x) = \mathrm{e}^x$ in the Fourier basis on $(-2, 2)$, so that $\|f - f_i\|_{[-1,1]} = 0$. This illustrates the redundancy in the frame.

It is well defined everywhere in R and is a periodic extension of the function f. However, we are usually not interested in any particular extension. In fact, a better name would be *Fourier restriction frame*, emphasizing that the interest is in the approximation properties on the restriction to Ω. Unfortunately, Fourier Extension is more common than Fourier restriction, so FE will be used throughout the thesis to describe this frame.

In order to compute converging expansions in this frame, note that since Φ_N is tight with frame bounds 1, it is self dual. The dual frame expansion

$$f = \sum_{k=1}^{\infty} \langle f, \phi_k \rangle_\Omega \phi_k \tag{1.27}$$

is equivalent to the Fourier series on $\mathcal{L}^2_R$ of the extension of f by zero

$$\tilde{f}(\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}), & \boldsymbol{x} \in \Omega, \\ 0, & \text{elsewhere.} \end{cases}$$

As mentioned in §1.2.1, this Fourier series suffers from the Gibbs phenomenon if f or any of its derivatives are discontinuous at the boundary of Ω. This illustrates that even though the canonical dual frame expansion is guaranteed to converge, the convergence properties might not be very good.

In contrast, the orthogonal projection on $\mathcal{H}_N = \mathrm{span}(\Phi_N)$ can converge faster, under similar conditions as the regular Fourier series.

Theorem 1.18. *[1, Proposition 5.8] Let $\Omega \subset (-1, 1)^d$ be a sufficiently smooth (Lipschitz) domain. If $f \in \mathcal{H}^{k,d}$, the k-th standard Sobolev space[1] for Ω, then*

[1] For a definition, see (5.34)

for the exact projection

$$\|f - \mathcal{P}_N f\| \leq C_{k,d} N^{-k} \|f\|. \tag{1.28}$$

For the regularised projection, as an extension of Theorem 1.17

$$\|f - \mathcal{P}_N^\tau f\| \leq C_{k,d} \left(N^{-k} + \sqrt{\tau}\right) \|f\|. \tag{1.29}$$

In particular, this result guarantees superalgebraic convergence for infinitely differentiable f, up to $\sqrt{\tau}$ for the regularised projection.

1.4.3 Augmented Frames

Frames can be easily constructed by adding elements to a Riesz basis [26, Definition 6.1.2]. A common technique is to supplement a basis or frame with a small number of functions with special properties. For example, the convergence of Fourier series for nonperiodic functions on $\Omega = [a, b]$ can be sped up significantly by considering the truncated frame

$$\Phi_{N+r} = \{\mathrm{e}^{\frac{\mathrm{i}k\pi x}{b-a}}\}_{I_N} \cup \{P_k\}_{k=1}^r, \tag{1.30}$$

where P_k is a polynomial of degree k, and I_N is as in (1.12). A similar result to Theorem 1.18 holds, where the algebraic rate of convergence is limited by either the number of continuous derivatives of f, or the number of polynomials r [1, Proposition 5.9].

Another possibility is to supplement a basis or frame by functions with a known singularity. If a singularity is present in a function, this may significantly reduce the convergence of the approximations, but this can be avoided by including the right functions into the approximation space. We return to augmented frames in §5.1.

1.5 Sampling

The examples so far have all been approximations f_N that attempt to minimise the $\mathcal{L}_2$ norm error

$$\int_\Omega |f(\boldsymbol{x}) - f_N(\boldsymbol{x})|^2 d\boldsymbol{x}. \tag{1.31}$$

Computing the orthogonal projection through inversion of the Gram operator requires at least the inner products $b = \{\langle f, \phi_i \rangle\}, i \in I_N$. These might be

difficult to calculate efficiently, or accurately. We say the function f is *sampled* through inner products with $\{\phi_i\}_{i=1}^{\infty}$.

An alternative is to sample f through point evaluations. Replace (1.31) by the discrete norm

$$\sum_{\boldsymbol{x}\in P_\Omega} |f(\boldsymbol{x}) - f_N(\boldsymbol{x})|^2. \tag{1.32}$$

Here the function f is sampled in a countable set $P_\Omega \subset \Omega$. The approximation in $\{\phi_i\}_{i=1}^{\infty}$ that minimises this discrete norm is the solution of the system

$$Ac = b \tag{1.33}$$

where

$$A_{ij} = \phi_j(\boldsymbol{x}_i), \qquad b_i = f(\boldsymbol{x}_i).$$

We call A the *collocation* matrix. A can be seen as a discrete version of the synthesis operator $\mathcal{T}_N$, mapping coefficients not to functions but to function samples. If the number of collocation points equals the degrees of freedom this is an *interpolation* problem, and for suitable P_Ω the collocation matrix is non-singular and the solution to (1.33) is unique. Depending on the choice of interpolation points, the degree N interpolant may be very close to the orthogonal projection $\mathcal{P}_N f$, and have similar convergence results. For Chebyshev interpolation on $[-1, 1]$ the optimal sampling points are the Chebyshev nodes [100]

$$x_j = \cos\left(\frac{2j-1}{2N}\pi\right), \qquad j = 1, \ldots, N.$$

For Fourier series on $[0, 1]$ these are the equidistant points [51]

$$x_j = \frac{j}{N}, \qquad j = 0, \ldots, N-1.$$

Similar results are known for splines [96].

Remark 1.19. When formulating the normal equations $A^*A\boldsymbol{c} = A^*b$ the entries $(A^*A)_{ij} = \sum_k \phi_i(\boldsymbol{x}_k)\phi_j(\boldsymbol{x}_k)$ can be seen as approximations to the elements of the Gram matrix using a quadrature rule with points $\boldsymbol{x}_k$. If the sample points are equispaced and continually increased then $A^*A \to \mathcal{G}_N$. Similarly A^*b approximates the inner products of the basis functions with f.

For frame approximations based on point samples, a necessary condition for good convergence properties is *oversampling*: the number of sampling points should exceed the number of degrees of freedom. Unless mentioned otherwise the oversampling rate is a constant factor so that $M = \varrho N, \varrho > 1$. The system

(1.33) is then rectangular, and there is a unique least-squares solution vector c that minimises

$$\|Ac - b\|^2 = \sum_{\boldsymbol{x} \in P_\Omega} |f(\boldsymbol{x}) - (\mathcal{T}_N c)(\boldsymbol{x})|^2$$

When this system is solved through the T-SVD as in §1.4.1, Theorem 1.17 can be extended to the discrete sample case. Denote by $\mathcal{P}^\tau_{M,N}$ the T-SVD solution of (1.33) with M sample points and N degrees of freedom.

Theorem 1.20. *[3, Theorem 1.3]*

$$\|f - \mathcal{P}^\tau_{M,N} f\| \leq \|f - \sum_{n=1}^{N} z_n \phi_n\| + \kappa^\tau_{N,N} \|f - \sum_{n=1}^{N} z_n \phi_n\|_M + \tau \lambda^\tau_{M,N} \|z\|. \quad (1.34)$$

For well-behaved constants $\kappa^\tau_{N,N}$ and $\lambda^\tau_{M,N}$ the approximation thus converges up to $\tau\lambda^\tau_{M,N}$ if solutions with small norm exist. This markedly improves upon Theorem 1.17, signifying discretising and oversampling can actually be beneficial. However, these results are ongoing research and as of yet unpublished. For the specific case of Fourier extension, similar results have been established already [4], they will be reviewed in §2.3.2.

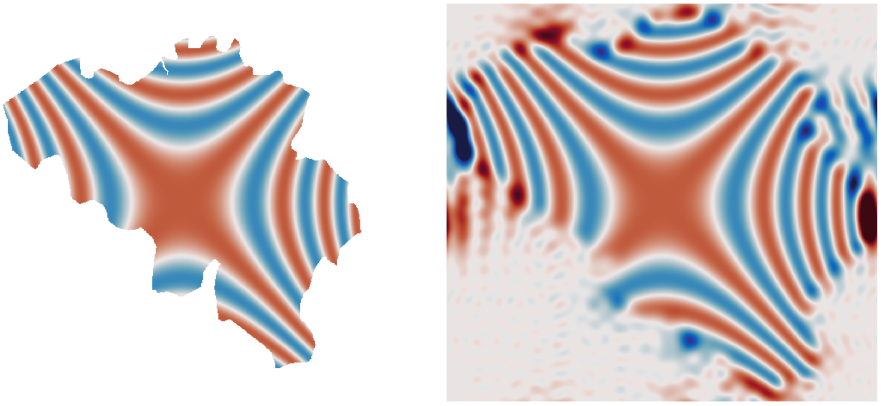
Note that the solution of (1.33) through the T-SVD is an explicit algorithm to compute regularised frame approximations from samples, at a cost of $\mathcal{O}(N^3)$ operations. Chapters 3 and 4 describe algorithms that improve upon this complexity.

An example of such an approximation is shown in Fig. 1.4. The function is given by $f(x, y) = \cos(20x^2 - 15y^2)$ where the domain takes on the shape of Belgium. The approximation $\mathcal{P}^\tau_{M,N} f$ where $N = 60^2$, $\tau = 10^{-14}$ and $\varrho = 2$ is shown in Fig. 1.4b. The maximum error satisfies $\|f - f_{60^2}\|_\infty < 10^{-12}$ on Ω, measured as the maximum in 10000 random points. Due to the nature of Φ_N the expansion is periodic when evaluated on the whole of R. Note that an orthonormal basis for $\mathcal{L}^2_\Omega$ in this case is hard to find, yet a frame for this space is trivially constructed.

An advantage that will come up repeatedly later on is that evaluation of a Fourier series in equidistant points can be calculated efficiently through the Fast Fourier transform (FFT). In particular b with

$$b_i = \sum_{j=1}^{N} c_j \mathrm{e}^{\mathrm{i}2\pi ji/N}, 0 \leq i \leq N-1 \quad (1.35)$$

can be computed from c in $\mathcal{O}(N \log N)$ operations. In §2.1, we will show that the collocation matrix for Fourier extension problems can be applied quickly to any vector using the FFT. The related discrete cosine transform (DCT) can be used to evaluate Chebyshev expansions in Chebyshev points at the same speed.



(a) Data points (b) Fourier Series on bounding box

Figure 1.4: Approximation of $f(x, y) = \cos(20x^2 - 15y^2)$ on a Belgium-shaped domain, using a Fourier series on a bounding box.

1.6 Overview

This introductory chapter has identified frames as useful generalisations of the well-known concept of (orthonormal) bases. A frame for a function space of interest may be much easier to construct than an orthonormal basis, with the example of Fourier series on irregular domains illustrated in Fig. 1.4. In order to keep the coefficients of the orthogonal projection to truncated frames bounded, the regularised projection $\mathcal{P}^\tau$ was introduced. These regularised projections have desirable convergence properties, up to a maximum accuracy of $\sqrt{\tau}$ or τ where τ is the regularisation parameter.

When approximating a function based on function samples, the regularised projection is necessarily oversampled. This sets the stage for Chapters 3 and 5: we will calculate approximations through solving an oversampled problem based on point samples using the T-SVD. The algorithms described in these chapters will provide efficient ways to calculate these approximations in practice.

Before the algorithms can be covered, in the following chapter we will study the Fourier extension frame in more detail. It has a long history of study as a self-contained subject, often without explicit links to frame theory. Some of these results will prove critical to the development of efficient algorithms.

In Chapter 3 we formulate two original algorithms to compute frame approximations using point samples for one dimensional Fourier extensions.

The execution time is loglinear in the number of degrees of freedom. Most of this chapter was previously published as [73], with the exception of the last section.

In Chapter 4 we adapt one of these algorithms to the higher dimensional case, and provide some necessary theoretical background. Most of this chapter is under review for publication as [75].

In Chapter 5 we cover some extensions of these algorithms to various applications. Examples include boundary value problems, and different types of frames. This chapter is being prepared for publication as [74].

In Chapter 6 we detail the implementation of these algorithms in a Juliapackage called *FrameFun*. Of particular interest is computing with arbitrary domains. These sections have appeared as [60].

In Chapter 7 we recap the most important contributions made in this thesis, and outline some possible future directions for this project.

Chapter 2

Fourier Extension

While the results of this thesis generalize to other frames and problem types, the Fourier Extension problem was the catalyst for all other developments. In this chapter we first formulate this central problem precisely, fixing the notation for the later chapters. Then, we go through the history of this problem and the relevant results from the literature. Of particular importance is the spectrum of the Gram and collocation matrices arising when computing the approximations.

2.1 Notation

In this section we formally state the Fourier extension problem as introduced in §1.4.2. Without loss of generality, assume that $\Omega \subset R = [0,1]^D$, so that we can use a tensor-product of the standard Fourier series on $[0,1]$. In order to avoid any periodicity requirements on f, we assume further that Ω lies fully in the interior of the box R. In the following, we will consistently use the symbols Ω for the time domain, and Λ for the frequency domain. For a set P_Λ with N_Λ frequencies, we denote the basis functions and the function space they span by

$$\phi_{\mathbf{l}}(\boldsymbol{x}) = \mathrm{e}^{\mathrm{i}(\boldsymbol{x}\cdot\mathbf{l})2\pi}, \tag{2.1}$$

$$\Phi_{N_\Lambda} = \operatorname{span}\{\phi_{\mathbf{l}}\}_{\mathbf{l}\in P_\Lambda}. \tag{2.2}$$

Here, $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_D)$ is a D-dimensional point and $\mathbf{l} = (\mathbf{l}_1, \ldots, \mathbf{l}_D)$ is a D-dimensional integer index. For simplicity, we assume an equal number of degrees of freedom n_Λ per dimension, hence $N_\Lambda = n_\Lambda^D$ and $\lfloor -n_\Lambda/2 \rfloor < \mathbf{l}_i < \lceil n_\Lambda/2 \rceil$.

This restriction could be lifted at the cost of minor complications further on, and it is not present in our implementation.

Disregarding the canonical dual frame expansion, we are interested in the orthogonal projection

$$\mathcal{P}_{N_\Lambda} f = \underset{g \in \Phi_{N_\Lambda}}{\arg\min} \|f - g\|_\Omega. \tag{2.3}$$

The problem of computing $\mathcal{P}_{N_\Lambda} f$ is sometimes referred to as the *Continuous Fourier Extension*, because the norm in (2.3) is the $\mathcal{L}^2_\Omega$ norm.

Any $g \in \Phi_{N_\Lambda}$ is uniquely described by a set of coefficients $\boldsymbol{c} \in \mathbb{C}^{n_\Lambda \times \cdots \times n_\Lambda}$. In what follows we will often assume an implicit linearization $\boldsymbol{c} \in \mathbb{C}^{N_\Lambda}$. These coefficients will be the result of the approximation algorithm, so we look for

$$\boldsymbol{c} = \underset{\boldsymbol{c} \in \mathbb{C}^{N_\Lambda}}{\arg\min} \|f - \sum_{\mathbf{l} \in P_\Lambda} \boldsymbol{c}_{\mathbf{l}} \phi_{\mathbf{l}}\|_\Omega. \tag{2.4}$$

The minimizer of (2.4) is found by constructing the Gram matrix

$$G_{\mathbf{k},\mathbf{l}} = \langle \phi_{\mathbf{k}}, \phi_{\mathbf{l}} \rangle_\Omega, \qquad \mathbf{k}, \mathbf{l} \in P_\Lambda \tag{2.5}$$

and solving the system

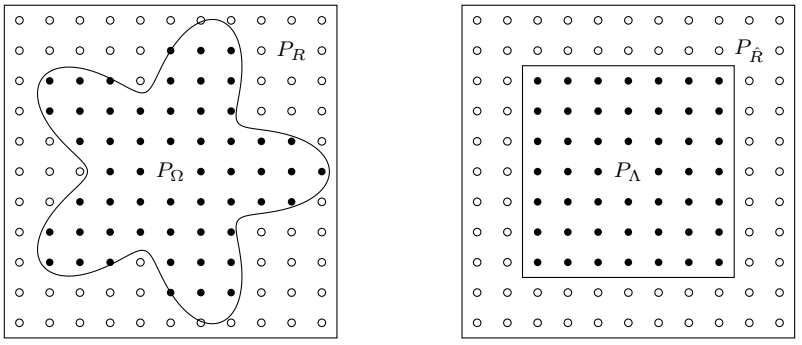
$$G\boldsymbol{c} = b, \qquad b_{\mathbf{l}} = \langle f, \phi_{\mathbf{l}} \rangle_\Omega. \tag{2.6}$$

However, the computational cost associated with evaluating the integrals in the right hand side of (2.6) is considerable, since the integrals are over Ω and not over the full box R. This precludes the use of the FFT and one would have to resort to some type of quadrature on Ω.

Instead, in this thesis we focus on the *Discrete Fourier approximation*. That is, the approximation found through oversampled collocation in equispaced points. We choose a point set P_R on R (recall that $R = [0,1]^D$) with n_R points per dimension, and restrict those to Ω. In summary

$$P_R = \left\{ \left(\frac{k_1}{n_R}, \ldots, \frac{k_D}{n_R} \right) \middle| \forall i : 0 \leq k_i < n_R \right\}, \qquad P_\Omega = P_R \cap \Omega. \tag{2.7}$$

There are efficient transformations using the FFT between the set P_R of $N_R = n_R^D$ points in the time domain and the set $P_{\hat{R}}$ in the frequency domain. If we choose $n_R \geq n_\Lambda$, then $P_{\hat{R}}$ encompasses P_Λ. The sampling sets thus defined are shown in Fig. 2.1. Though other choices can be made, this choice is such that we can efficiently evaluate a Fourier series using the index set P_Λ in all the points of P_Ω: we extend the coefficients with zeros from P_Λ to $P_{\hat{R}}$, followed



(a) time domain (b) frequency domain

Figure 2.1: The spatial domain Ω encompassing the sample set P_Ω, and the frequency domain Λ encompassing the discrete frequencies P_Λ. There is a fast FFT transform between the encompassing sets P_R and $P_{\hat{R}}$.

by an FFT transform from $P_{\hat{R}}$ to P_R, followed by a restriction of the values to those points in P_Ω.

We mostly assume a fixed oversampling rate, meaning $\varrho = N_\Omega/N_\Lambda$ is constant. We refer to [5, 3] for a study on the interplay of oversampling rate and choice of bounding box in one dimension, and more general results regarding the oversampling rate for collocation in frames. The one-dimensional results are recapped in §2.3.3.

As in §1.5, the minimization (2.4) is reformulated as a discrete least squares problem

$$F_{N_\Lambda}(f) = \operatorname*{arg\,min}_{g \in \Phi_{N_\Lambda}} \sum_{\boldsymbol{x} \in P_\Omega} (f(\boldsymbol{x}) - g(\boldsymbol{x}))^2. \tag{2.8}$$

Assuming a linear indexing $\boldsymbol{x}_k$ of P_Ω from 1 to N_Ω and ϕ_j of P_Λ from 1 to N_Λ, this is a least squares matrix problem

$$A\boldsymbol{c} = b, \quad A \in \mathbb{C}^{N_\Omega \times N_\Lambda}, \quad b \in \mathbb{C}^{N_\Omega} \tag{2.9}$$

where

$$A_{kj} = \frac{1}{\sqrt{N_R}} \phi_j(\boldsymbol{x}_k), \qquad b_k = f(\boldsymbol{x}_k). \tag{2.10}$$

The scaling of the basisfunctions is such that A is precisely a subblock of a multidimensional unitary DFT matrix, which will be of importance later on. This subblock property is a consequence of our choice of discrete grids, and it results in a fast matrix-vector product using the procedure described above: extension in frequency domain, DFT, and restriction in the time domain. Indeed,

note that in this discrete setting the action of the matrix A corresponds to evaluating a length N_Λ Fourier series in the points of P_Ω.

Note that there is also a fast matrix-vector product for A^* and it corresponds to the opposite sequence of operations: extension from P_Ω to P_R by zeros, fast transform to $P_{\hat{R}}$, followed by restriction in the frequency domain from $P_{\hat{R}}$ to P_Λ. Yet, it is clear that the solution to $Ax = b$ is not simply given by $x = A^*b$. Indeed, this is precisely the canonical dual frame approximation from §1.4, which is likely to exhibit the Gibbs phenomenon unless f goes to zero smoothly along the boundary $\delta\Omega$.

Remark 2.1. Though this chapter contains results for Fourier bases exclusively, a Chebyshev collocation matrix in Chebyshev points is entirely analogous. It consists of an appropriately scaled subblock of a multidimensional DCT matrix. The DCT is highly similar to the FFT, and in practice often computed through an optimised FFT routine. Therefore, most of the arguments, although not made explicit, apply to this case as well. This is of course due to the close connection between Chebyshev polynomials and trigonometric polynomials (see (1.14)).

Of particular importance is the SVD of the Gram (2.5) and collocation (2.10) matrices. As will become apparent throughout this chapter, the singular values for the matrices stemming from Fourier Extension problems always have a distinct profile, shown in Fig. 2.2. Three distinct regions are visible. For some $1 > \tau > 0$ we observe:

- A region $I_\alpha := \{\sigma : 1 > \sigma > 1 - \tau\}$ where all singular values are 1 up to a tolerance τ. This region contains approximately $N_\Omega N_\Lambda / N_R$ singular values.
- A region $I_\beta := \{\sigma : 1 - \tau \geq \sigma > \tau\}$, also referred to as the "plunge region" in a more general context regarding truncated frames. This name will be used often through the following chapters. The size of this region as the degrees of freedom N_Λ increases is always $o(N_{\hat{R}})$. This result can be traced back to [65, Theorem 1], but the first term in the asymptotic expansion can often be estimated, see Property 2.15 and Theorem 4.17.
- A region $I_\gamma := \{\sigma : \tau \geq \sigma > 0\}$ where the singular values further decay exponentially. Due to rounding errors, these singular values are difficult to compute past $\sigma \sim \varepsilon_{\text{mach}}$ without resorting to extended precision.

For later use we denote by $\eta(\tau, N_\Lambda)$ the size of the plunge region I_β,

$$\eta(\tau, N_\Lambda) = \min_{1 \leq j < k \leq N_\Lambda} (k - j - 1) \quad \text{s.t.} \quad \sigma_j > 1 - \tau, \quad \tau \geq \sigma_k. \tag{2.11}$$

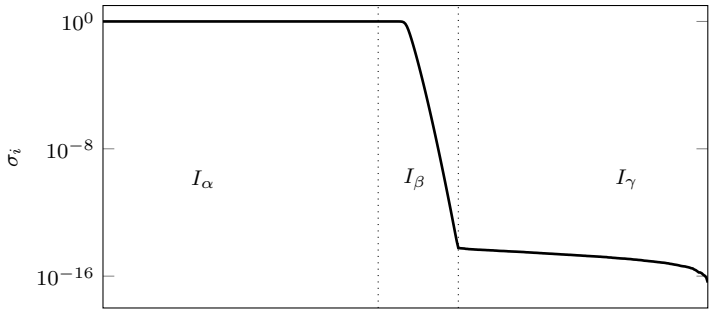


Figure 2.2: The subdivision of the spectrum of A into three distinct intervals, here indicated based on $\tau = 10^{-14}$. Due to rounding errors, the eigenvalues in region I_γ don't decay past machine precision.

The subdivision of singular values naturally leads to a subdivision of singular vectors. Let

$$U\Sigma V^* = \begin{bmatrix} U_\alpha & U_\beta & U_\gamma \end{bmatrix} \begin{bmatrix} \Sigma_\alpha & & \\ & \Sigma_\beta & \\ & & \Sigma_\gamma \end{bmatrix} \begin{bmatrix} V_\alpha & V_\beta & V_\gamma \end{bmatrix}^*,$$

where Σ_α is a diagonal matrix containing the singular values from I_α, and Σ_β and Σ_γ are defined analoguously. Note that for the collocation and Gram matrices each right singular vector v_i corresponds to a set of Fourier coefficients. Let

$$\mathcal{H}_\alpha = \operatorname{span}\{\mathcal{T}_N v\}, \qquad v \in V_\alpha$$

denote the space spanned by the right singular vectors V_α, with similar definitions $\mathcal{H}_\beta$ and $\mathcal{H}_\gamma$ for the other regions. Note that since the singular vectors are orthogonal and $\mathcal{T}$ is isometric, these spaces consist of orthonormal functions:

$$\langle \mathcal{T}_N v_i, \mathcal{T}_N v_j \rangle = \langle v_i, v_j \rangle = \delta_{ij}$$

Thus, the spaces $\mathcal{H}_\alpha, \mathcal{H}_\beta$ and $\mathcal{H}_\gamma$ are mutually orthogonal. Recall from §1.4.1 that the regularized projection $\mathcal{P}_N^\tau$ is defined in terms of the Truncated Singular Value Decomposition

$$A_N^\tau = \begin{bmatrix} U_\alpha & U_\beta \end{bmatrix} \begin{bmatrix} \Sigma_\alpha & \\ & \Sigma_\beta \end{bmatrix} \begin{bmatrix} V_\alpha & V_\beta \end{bmatrix}. \tag{2.12}$$

In this light the regularized projection $\mathcal{P}_N^\tau$ can be seen as an orthogonal projection on the space $\mathcal{H}_{\alpha+\beta} = \mathcal{H}_\alpha \cup \mathcal{H}_\beta$. These function spaces are spanned by Prolate Spheroidal Wave Functions and related functions, the topic of the next section.

2.2 FE as bandlimited Extrapolation

The Gram and collocation matrices for the Fourier Extension problem have appeared in a multitude of contexts, starting with signal processing research at Bell labs in the 1960s and 1970s. Slepian, Landau, Pollack and collaborators studied the problem of bandlimited extrapolation in a series of highly influential papers [91, 66, 84, 85, 89]. The problem they considered is the reconstruction of a signal from a time-limited observation, knowing the signal has limited support in the frequency domain. The following subsections describe the solutions to this extrapolation problem and its generalizations. As it turns out, the orthonormal bases for these problems are heavily connected to the FE problem. The first three subsections describe one-dimensional results, while §2.2.4 details higher-dimensional generalizations.

2.2.1 Prolate Spheroidal Wave Functions

Denote by $f(x)$ and $F(\xi)$ a function in $\mathcal{L}^2$ and its Fourier transform, so that

$$F(\xi) = \int_{-\infty}^{\infty} f(x)\mathrm{e}^{-\mathrm{i}2\pi x\xi}dx, \qquad f(x) = \int_{-\infty}^{\infty} F(\xi)\mathrm{e}^{\mathrm{i}2\pi x\xi}d\xi. \tag{2.13}$$

The time- and bandlimiting operators $\mathcal{D}$ and $\mathcal{B}$ are then defined as

$$\mathcal{D}f(x) = \breve{f}(x) = \begin{cases} f(x) & |x| \le W \\ 0 & |x| > W \end{cases} \qquad \mathcal{B}f(x) = \int_{-\omega}^{\omega} F(\xi)\mathrm{e}^{\mathrm{i}2\pi\xi x}d\xi, \tag{2.14}$$

which project onto $\mathcal{L}^2_{[-W,W]}$ and PW_ω, the Paley-Wiener space of bandlimited functions, respectively. In the context of the previous section this means $\Omega = [-W, W]$ and $\Lambda = [-\omega, \omega]$. Note that the bandlimiting operator can also be written as

$$\mathcal{B}f(x) = \int_{-\infty}^{\infty} f(s)\frac{\sin(2\pi\omega(x-s))}{\pi(x-s)}ds. \tag{2.15}$$

The Heisenberg-Gabor limit states that no nonzero function can be simultaneously concentrated in both time and frequency,

$$\|\mathcal{B}\mathcal{D}f\| < \|f\|.$$

However, one can look for nearly-invariant functions under this operator, functions for which $\|\mathcal{B}\mathcal{D}f\|/\|f\|$ is as close to 1 as possible.

Definition 2.2. *The* Prolate Spheroidal Wave Function*s* ϑ *are the eigenfunctions of the operator* $\mathcal{BD}$*, i.e. the solutions of*

$$\lambda\vartheta(x) = \int_{-W}^{W} \vartheta(s)\frac{\sin(2\pi\omega(x-s))}{\pi(x-s)}ds = \mathcal{BD}\vartheta, \tag{2.16}$$

normalised so that $\|\vartheta\| = 1$*.*

(2.16) is a Fredholm integral equation of the first kind. Slepian and collaborators showed that this equation has solutions only for select values of λ, a countably infinite set $1 > \lambda_0 > \lambda_1 > \cdots > 0$. Therefore there exists a unique Prolate Spheroidal wave function ϑ_0 that is maximally concentrated in both time and frequency, with $\|\mathcal{BD}f\|/\|f\| = \lambda_0$. This zeroth order Prolate is often used in filter designs [79, 97]. The naming stems from the curious observation that these functions are solutions to the spheroidal wave equation[1]

$$\left(1 - \frac{x^2}{W^2}\right)\frac{d^2\vartheta_i}{dx^2} - 2x\frac{d\vartheta_i}{dx} - (2\pi\omega W)^2 x^2\vartheta_i = \theta_i\vartheta_i. \tag{2.17}$$

This is a Sturm-Liouville equation with a set of unique eigenvalues $\cdots > \theta_{i-1} > \theta_i > \theta_{i+1} > \ldots$ corresponding to the functions ϑ_i, but different from the λ_i [91].

Since $\mathcal{B}$ and $\mathcal{D}$ are idempotent operators, it is convenient to consider the ϑ_i eigenfunctions of the Hermitian operator $\mathcal{BDB}$. Timelimiting both sides of (2.16), the timelimited functions $\breve{\vartheta}_i = \mathcal{D}\vartheta_i$ are the eigenfunctions of the Hermitian operator $\mathcal{DBD}$, with corresponding eigenvalues λ_i. The term Prolate Spheroidal Wave function is used for both the ϑ_i and the $\breve{\vartheta}_i$.

Property 2.3. *As eigenfunctions of a Hermitian operator, the* ϑ_i *and* $\breve{\vartheta}_i$ *are orthogonal*

$$\int_{-\infty}^{\infty} \vartheta_i(x)\vartheta_j(x)dx = \delta_{ij}, \qquad \int_{-W}^{W} \breve{\vartheta}_i(x)\breve{\vartheta}_j(x)dx = \lambda_i\delta_{ij},$$

and they are complete in PW_ω *and* $\mathcal{L}^2_{[-W,W]}$*, respectively.*

The Prolate Spheroidal Wave functions thus form an orthonormal basis for PW_ω, while the eigenvalue λ_i represents the fraction of energy of ϑ_i contained in $[-W, W]$. When normalised in $\mathcal{L}^2_{[-W,W]}$, $\{\vartheta_i/\sqrt{\lambda_i}\}$ becomes an orthonormal basis for this space as well.

[1]There are differing conventions regarding notation and normalisation for these functions, we follow [91] here.

This leads to a straightforward approach to continuous bandlimited extrapolation. Let f be a function segment in $\mathcal{L}^2_{[-W,W]}$. Then

$$g = \sum_{i=1}^{\infty} \frac{\langle \breve{\vartheta}_i, f \rangle}{\lambda_i} \vartheta_i \tag{2.18}$$

is a bandlimited function that agrees with f in the interval due to the completeness of the $\breve{\vartheta}_i$ in $\mathcal{L}^2_{[-W,W]}$. Furthermore, when truncating the sum, the first terms have the largest eigenvalues and capture the ϑ_i with relatively more of their energy inside the interval. The following relevant properties are from [91].

Property 2.4. *The ϑ_i are eigenfunctions of the finite Fourier transform,*

$$\int_{-W}^{W} \mathrm{e}^{\mathrm{i}2\pi t\xi} \vartheta_n(t) dt = \mathrm{i}^n \left(\frac{\lambda W}{\omega}\right)^{1/2} \vartheta_n\left(\frac{\xi W}{\omega}\right).$$

Property 2.5. *The eigenvalues λ_i cluster near 1 for low values of i, and decay exponentially after a set breakpoint*

$$\lambda_i \approx 1, i \ll 4\omega W \quad \textit{and} \quad \lambda_i \approx 0, i \gg 4\omega W.$$

The width of the region where $\lambda_i \in (\epsilon, 1-\epsilon)$ grows as $\log(\omega T)$.

Property 2.6. *Among functions in PW_ω, ϑ_0 is the most concentrated in $(-T, T)$ and its concentration is λ_0. Among functions in PW_ω orthogonal to ϑ_0, ϑ_1 is the most concentrated, with concentration λ_1, and so on.*

The suitability of PSWFs as an approximation scheme was investigated by Boyd [12], amongst others. The main difficulty associated with this technique is that the PSWFs can not be expressed in terms of classical functions. Thus, numerical schemes must always use approximations, which may be computationally expensive [79].

Remark 2.7. From the definition, it is clear the PSWF are only defined up to some complex unitary constant. This will hold for the generalisations in the following sections as well. Uniqueness is usually obtained through requiring either $\vartheta_i(0)$ or $\vartheta_i'(0)$ be real and positive. Since we are only interested in the spaces these functions span, we will not explicitly impose such restrictions.

2.2.2 Discrete Prolate Spheroidal Wave Functions and Sequences

There are several possible discretisations for PSWFs. The most well-known is the one proposed by Slepian[89], where the Fourier transform from (2.13) is

replaced with a Fourier series representation as in §1.2.1, on $R = [-1/2, 1/2]$

$$f(x) = \sum_{n=-\infty}^{\infty} c_n \mathrm{e}^{i2\pi xn}, \qquad c_n = \frac{1}{2\pi} \int_{-1/2}^{1/2} f(s) \mathrm{e}^{-\mathrm{i}2\pi sn} ds.$$

One could define the time and band-limiting operators here as

$$\mathcal{D}f(x) = \breve{f}(x) = \begin{cases} f(x) & |x| \le W \\ 0 & |x| > W \end{cases}$$

$$\mathcal{B}f(x) = \sum_{n \in I_N} \left(\frac{1}{2\pi} \int_{-1/2}^{1/2} f(s) \mathrm{e}^{-\mathrm{i}2\pi sn} dx \right) \mathrm{e}^{\mathrm{i}2\pi xn}.$$

Here $\Omega = [-W, W]$ is some subinterval of $[-1/2, 1/2]$, and I_N is as in (1.12). For ease of notation we assume N_Λ odd, since the expressions for the DPSWF and DPSS are slightly different based on the parity of N_Λ. Note that, similar to (2.15),

$$\mathcal{B}f(x) = \int_{-1/2}^{1/2} f(s) \frac{\sin N_\Lambda \pi(x-s)}{\sin \pi(x-s)} ds.$$

and

$$\mathcal{BD}f(x) = \int_{-W}^{W} f(s) \frac{\sin N_\Lambda \pi(x-s)}{\sin \pi(x-s)} ds.$$

Definition 2.8. *The* Discrete Prolate Spheroidal Wave Functions Φ_i *are defined as the eigenfunctions of* $\mathcal{BD}$

$$\lambda_i \Phi_i = \mathcal{BD}\Phi_i = \int_{-W}^{W} \Phi_i(s) \frac{\sin N_\Lambda \pi(x-s)}{\sin \pi(x-s)} ds, \qquad i = 1, \dots, N_\Lambda, \tag{2.19}$$

and the associated Discrete Prolate Spheroidal Sequences ψ *by*

$$\lambda_i \sum_{n=-\infty}^{\infty} \mathrm{e}^{\mathrm{i}2\pi xn} \psi_i[n] = \mathcal{B}\Phi_i.$$

The properties from section 2.2.1 largely carry over. The integral equation (2.19) has a degenerate kernel, meaning it has only N_Λ non-zero eigenvalues which are distinct. The DPSWF are the N_Λ eigenfunctions of this operator. As before, the DPSWF satisfy a Sturm-Liouville differential equation; the DPSS satisfy a second order difference equation. As a result, the double orthogonality holds:

Property 2.9.

$$\int_{-W}^{W} \Phi_i(x)\Phi_j(x)dx = \lambda_i \int_{-1/2}^{1/2} \Phi_i(x)\Phi_j(x)dx = \lambda_i \delta_{ij}$$

$$\sum_{n\in I_N} \psi_i[n]\psi_j[n] = \lambda_i \sum_{n=\infty}^{\infty} \psi_i[n]\psi_j[n] = \lambda_i \delta_{ij}.$$

Furthermore, $\{\breve{\Phi}_i/\sqrt{\lambda_i}\}$ forms an orthonormal basis for its span and can be used for approximations

$$g = \sum_{i=1}^{N_\Lambda} \frac{\langle \breve{\Phi}_i, f\rangle}{\lambda_i}\Phi_i$$

We note the following two properties concerning the index-limited sequences $\breve{\psi}_i$.

Property 2.10. *The $\breve{\psi}_i$ are the eigenvectors of the $N_\Lambda \times N_\Lambda$ matrix Q with entries*

$$Q_{ij} = \int_{-W}^{W} \mathrm{e}^{-\mathrm{i}2\pi s(i-j)}ds = \frac{\sin 2\pi W(i-j)}{\pi(i-j)}, \qquad i,j = 1,\ldots,N_\Lambda.$$

The matrix Q is known as the *prolate matrix* [102].

Property 2.11. *The $\breve{\psi}_i$ are the eigenvectors of the $N_\Lambda \times N_\Lambda$ tridiagonal matrix χ with entries*

$$\chi_{ij} = \begin{cases} \frac{1}{2}i(N_\Lambda - 1), & j = i-1 \\ \left(\frac{N_\Lambda - 1}{2} - i\right)^2 \cos 2\pi W, & j = i \\ \frac{1}{2}(i+1)(N_\Lambda - 1 - i), & j = i+1 \\ 0, & |j-i| > 1 \end{cases}, \qquad j,i = 0,\ldots,N_\Lambda - 1.$$

Property 2.11 is a direct result of the DPSS satisfying a second order difference equation. At this point the connection to the FE problem becomes clear: the matrix Q from Property 2.10 is exactly the Gram matrix (2.5) of the Continuous FE with $\Omega = [-W, W]$ and $P_\Lambda = I_N$. As such

$$\mathcal{P}^{\tau}_{N_\Lambda} f = \sum_{\lambda_k > \tau} \frac{\langle \breve{\Phi}_i, f\rangle}{\lambda_i}\Phi_i,$$

the (regularized) orthogonal projections on the FE frames are naturally expressed in terms of the Φ and ψ. This connection leads to a computational advantage. The tridiagonal matrix χ is well conditioned with separated eigenvalues [102, 50]. Therefore, individual functions $\breve{\psi}$ can be calculated efficiently as the eigenvectors of χ, in contrast to the PSWF [48].

2.2.3 Periodic Discrete Prolate Spheroidal Sequences

Seeing the close relation between continuous FE and the DPSWF in the previous section, one can wonder whether a further generalization of the Prolate Spheroidal Wave Functions exists that relates to the discrete FE, where the best approximation is formulated in terms of oversampled collocation. The answer is found in discrete bandlimited approximation, as defined in [61, 110, 20].

With the Discrete Fourier Transform for sequences of length N_R as

$$G[k] = \frac{1}{\sqrt{N_R}} \sum_{n=0}^{N_R-1} g[n] \mathrm{e}^{-\mathrm{i}2\pi kn/N_R} \qquad g[n] = \frac{1}{\sqrt{N_R}} \sum_{k=0}^{N_R-1} G[k] \mathrm{e}^{\mathrm{i}2\pi kn/N_R}.$$

Denote by F the $N_R \times N_R$ DFT matrix that maps g to G. Since F is unitary the inverse transform matrix is F^*. Then as before the (discrete) time- and bandlimiting operators are

$$D_\Omega v[k] = \breve{v}[k] = \begin{cases} v[k] & k \in P_\Omega \\ 0 & \text{otherwise} \end{cases}$$

$$B_\Lambda v = F D_\Lambda F^* v.$$

Here P_Ω and P_Λ denote the time and frequency sampling sets respectively. The operators are $N_R \times N_R$ real symmetric projection matrices, i.e. $D_\Omega^2 = D_\Omega$ and $B_\Lambda^2 = B_\Lambda$. If P_Λ is as in (1.12) and thus consists of contiguous frequency samples $(-(N_\Lambda - 1)/2, \ldots, (N_\Lambda - 1)/2) \mod N_R$, B_Λ is the *discrete prolate matrix*

$$(B_\Lambda)_{jk} = \frac{1}{\sqrt{N_R}} \sum_{l \in P_\Lambda} \mathrm{e}^{\mathrm{i}\frac{(j-k)2\pi l}{N_R}} = \frac{1}{\sqrt{N_R}} \frac{\sin \frac{(j-k)N_\Lambda \pi}{N_R}}{\sin \frac{(j-k)\pi}{N_R}}. \tag{2.20}$$

Note the similarity to the matrix Q from Property 2.10. The discrete bandlimited extrapolation problem formulated in terms of these operators is to find, given $D_\Omega f$, a bandlimited sequence $B_\Lambda g$ so that

$$D_\Omega B_\Lambda g = D_\Omega f \tag{2.21}$$

Jain and Ranganath [61] approached this problem by formulating the normal equations

$$B_\Lambda^* D_\Omega B_\Lambda g = B_\Lambda^* D_\Omega f.$$

They suggested the Levinson-Trench algorithm [112] to compute the inverse of $B_\Lambda^* D_\Omega B_\Lambda$ in $O(N_R^2)$ operations. Alternatively, they suggested using the SVD of $B_\Lambda^* D_\Omega B_\Lambda$ to solve the least-squares problem. This led to the definition of the Periodic Discrete Prolate Spheroidal Sequence[110].

Definition 2.12. *The Periodic Discrete Prolate Spheroidal Sequences φ_i are the eigenvectors of the Hermitian matrix $B_\Lambda^* D_\Omega B_\Lambda$ for which the eigenvalue μ_i is nonzero,*

$$B_\Lambda^* D_\Omega B_\Lambda \varphi_i = \mu_i \varphi_i, \qquad \mu_i > 0.$$

The P-DPSS share many properties with the PSWF and DPSWF. From the definition it follows that the time-limited versions satisfy

$$D_\Omega B_\Lambda D_\Omega \breve{\varphi}_i = \mu_i \breve{\varphi}_i, \qquad \mu_i > 0. \tag{2.22}$$

The φ_i are doubly orthogonal

$$\langle \varphi_i, \varphi_j \rangle = \delta_{ij}, \qquad \langle D_\Omega \varphi_i, D_\Omega \varphi_j \rangle = \mu_i \delta_{ij}.$$

The φ_i properties are similar to those of PSWFs:

Property 2.13. *The eigenvalues satisfy $1 \geq \mu_i \geq 0$.*

Proof. Since B_Λ is a projector it has eigenvalues 1 and 0. The property follows directly from the interlacing theorem for principal submatrices of Hermitian matrices (see Theorem 5.1). □

Property 2.14. *[110] There are exactly $\min(N_\Lambda, N_\Omega)$ P-DPSS. If $N_\Omega \geq N_\Lambda$, the φ_i are complete in the space of bandlimited sequences, spanned by the eigenvectors of B_Λ. If $N_\Omega \leq N_\Lambda$, the nonzero parts of the $\breve{\varphi}_i$ are complete in $\mathbb{R}^{N_\Omega}$.*

Property 2.15. *[106, 35, 111] Like the eigenvalues of the PSWFs, the eigenvalues μ_i are distinct and cluster exponentially near 1 and 0, in that*

$$\mu_i \approx 1, i \ll \frac{N_\Omega N_\Lambda}{N_R} \quad \textit{and} \quad \mu_i \approx 0, i \gg \frac{N_\Omega N_\Lambda}{N_R}.$$

The width of the plunge region where $\mu_i \in (\tau, 1-\tau)$ grows as $\mathcal{O}(\log N_\Omega N_\Lambda / N_R)$, for any $1 > \tau > 0$.

Remark 2.16. Throughout this thesis we assume the oversampling ratio is constant, i.e. $N_\Omega = \varrho N_\Lambda$. We further have a constant bounding box P_R with respect to Ω. In this case,

$$\mathcal{O}(\log N_\Omega N_\Lambda / N_R) = \mathcal{O}(\log N_\Lambda), \qquad N_\Lambda \to \infty,$$

and we use the latter for notational convenience.

Property 2.17. *The P-DPSSs satisfy a second order difference equation [110]*

$$b_k \breve{\varphi}_i[k-1] + c_k \breve{\varphi}_i[k] + b_{k+1} \breve{\varphi}_i[k+1] = \theta_i \breve{\varphi}_i[k], \qquad k = 1, \ldots, N_\Omega, \quad (2.23)$$

with coefficients

$$b_k = \sin\left(\frac{\pi k}{N_R}\right) \sin\left(\frac{\pi(N_\Omega - k)}{N_R}\right)$$

$$c_k = -\cos\left(\frac{\pi(2k - 1 - N_\Omega)}{N_R}\right) \cos\left(\frac{\pi N_\Lambda}{N_R}\right).$$

Here $\breve{\varphi}_i[0]$ and $\breve{\varphi}_i[N_\Omega + 1]$ are understood to be zero.

The other properties follow directly from the definitions:

Property 2.18. *Among sequences of length N_R, with frequency support in P_Λ, φ_0 is the most concentrated in P_Λ. Among sequences of equally limited frequency support orthogonal to φ_0, φ_1 is the most concentrated in Ω, and so on.*

Property 2.19. *Due to the duality of time and frequency, the P-DPSSs are eigenvectors of the index-limited DFT, but with the roles of Ω and Λ interchanged. Denote by $\varphi_{\Omega,\Lambda}$ the P-DPSS as in Definition 2.12, then*

$$D_\Lambda F \varphi_{\Omega,\Lambda,i} = D_\Lambda \varphi_{\Lambda,\Omega,i}.$$

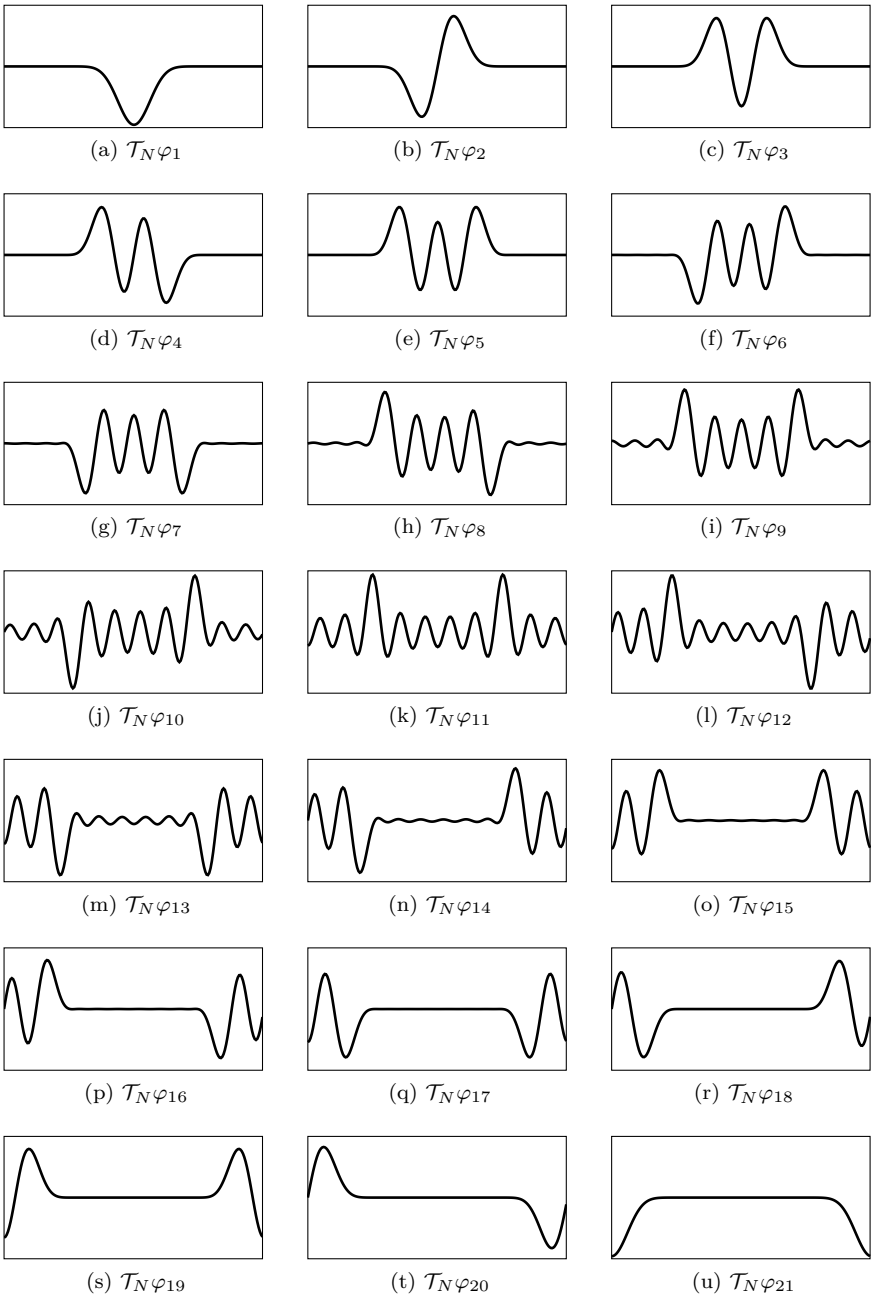
As with the DPSS, Property 2.17 leads to efficient calculations of a single P-DPSS. We will return to this procedure in the next chapter. Figure 2.3 shows examples of $\mathcal{T}_N \varphi_i$, the Fourier series associated with the P-DPSS, for $N_\Lambda = 21, N_\Omega = 41$, and $N_R = 81$. $R = [-1, 1]$ and $\Omega = [-1/2, 1/2]$. For the first few sequences, μ_i is very close to one, and $\|\psi_i\| \sim \|\breve{\psi}_i\|$, so the energy is almost completely concentrated inside Ω. For the last sequences, λ_i is close to zero, and $\|\psi_i\| \sim 0$, so the energy is almost completely concentrated in the complement of Ω. Property 2.15 is equivalent to saying the number of these functions, that are significant at the boundary of Ω, grow only as $\mathcal{O}(\log N_\Lambda)$.

A closer look at the collocation matrix A from the discrete Fourier extension in (2.10) reveals that

$$(AA^*)_{kl} = (D_\Omega B_\Lambda D_\Omega), \qquad k, l \in P_\Omega$$

$$(A^*A)_{kl} = (D_\Lambda B_\Omega D_\Lambda), \qquad k, l \in P_\Lambda.$$

From (2.22) it follows that the left singular vectors of A are thus exactly the $\breve{\varphi}_{\Omega,\Lambda}$, while the right singular vectors are the $\breve{\varphi}_{\Lambda,\Omega}$, the dual P-DPSS for which



(a) $\mathcal{T}_N\varphi_1$ (b) $\mathcal{T}_N\varphi_2$ (c) $\mathcal{T}_N\varphi_3$

(d) $\mathcal{T}_N\varphi_4$ (e) $\mathcal{T}_N\varphi_5$ (f) $\mathcal{T}_N\varphi_6$

(g) $\mathcal{T}_N\varphi_7$ (h) $\mathcal{T}_N\varphi_8$ (i) $\mathcal{T}_N\varphi_9$

(j) $\mathcal{T}_N\varphi_{10}$ (k) $\mathcal{T}_N\varphi_{11}$ (l) $\mathcal{T}_N\varphi_{12}$

(m) $\mathcal{T}_N\varphi_{13}$ (n) $\mathcal{T}_N\varphi_{14}$ (o) $\mathcal{T}_N\varphi_{15}$

(p) $\mathcal{T}_N\varphi_{16}$ (q) $\mathcal{T}_N\varphi_{17}$ (r) $\mathcal{T}_N\varphi_{18}$

(s) $\mathcal{T}_N\varphi_{19}$ (t) $\mathcal{T}_N\varphi_{20}$ (u) $\mathcal{T}_N\varphi_{21}$

Figure 2.3: $\mathcal{T}_N\varphi_i$ for $N_\Lambda = 21, N_\Omega = 41, N_R = 81$. In this case $R = [-1, 1]$ and $\Omega = [-1/2, 1/2]$. The ratio $\|\psi_i\|_\Omega / \|\psi_i\| = \mu_i$ decreases with i.

Ω is the frequency domain and Λ is the time domain. For the singular values σ_i of A this leads to $\sigma_i = \sqrt{\mu_i}$. The collocation problem is therefore just a reformulation of (2.21).

Where the normal equations method by Jain and Ranganath has the function values as unknowns

$$g = \sum_{i=0}^{N_R-1} \frac{1}{\mu_i} \langle \breve{\varphi}_i, B_\Lambda D_\Omega f \rangle_\Omega \varphi_i,$$

calculating the coefficients of the discrete FE by truncating the SVD as in (2.12) leads to

$$(A_N^\tau)^\dagger f = \sum_{\sqrt{\mu_i} > \tau} \frac{1}{\sqrt{\mu_i}} \langle \breve{\varphi}_{\Lambda,\Omega,i}, f \rangle_\Omega \breve{\varphi}_{\Omega,\Lambda,i}. \tag{2.24}$$

This formulation of the FE together with the P-DPSS properties listed above leads to the fast algorithms in the following chapters. Note here that the formulation of the normal equations leads to a lower number of P-DPSS present in the approximation compared to (2.24), since the lowest eigenvalue present is $\mu_i \sim \tau$, versus $\mu_i \sim \tau^2$ for the collocation system. This is reminiscent of the factors $\sqrt{\tau}$ and τ in Theorem 1.17 and Theorem 1.20.

Besides these SVD methods to solve (2.21), a well known method is the Papoulis-Gerschberg algorithm [44, 81]. It uses a two-step iteration process to alternate matching the given data and complying with the frequency constraints. Variants that use the conjugate gradients and related methods to speed up the iteration process tend to perform reasonably well numerically [95]. They operate at a cost of $O(N_\Lambda \log N_\Lambda)$ operations per iteration, where the number of iterations scales with the bandwidth of the signal. However, these methods only provide a very modest accuracy.

2.2.4 Multi-dimensional extensions

While the one dimensional Prolate Spheroidal Wave functions received considerable interest in signal processing and mathematics (for overviews, see [79, 58, 90, 49]), the generalization to multiple dimensions is not straightforward. Most generalizations are restricted to a setting where $\Omega = \Lambda$, or require at least some structure in both time and frequency domains. In contrast, the most general multidimensional equivalent would be for arbitrary 'frequency' and 'time' domains.

Multidimensional equivalents of PSWFs were first considered by Slepian and Pollack [85]. They demonstrated a double orthogonality property similar to the one dimensional case, and that the eigenvalues of the corresponding integral

equation are real and positive. Afterwards, they focused only on the most symmetric case, where both Ω and Λ are circular. In this case, the symmetry of the problem leads to PSWF generalizations as a combination of Bessel functions and one-dimensional PSWFs. Later results were described for rectangular time and frequency domains [9], or circular frequency regions [87]. For an overview, see [88].

Results on spectral properties for arbitrarily shaped regions appeared in 1982 [105], when H. Widom stated a conjecture on the traces of functions of Wiener-Hopf operators with discontinuous symbols in higher dimensions. The subject was the operator

$$\begin{aligned} T_\alpha f(\boldsymbol{x}) &= \left(\frac{\alpha}{2\pi}\right)^d \chi_\Omega \int_\Lambda \int_\Omega \mathrm{e}^{\mathrm{i}\alpha \boldsymbol{\xi}\cdot(\boldsymbol{x}-\boldsymbol{y})} f(\boldsymbol{y}) d\boldsymbol{y} d\boldsymbol{\xi}, \alpha > 0, \\ &= \mathcal{D}_\Omega \mathcal{B}_{\alpha\Lambda/2\pi} \mathcal{D}_\Omega, \end{aligned}$$

with higher-dimensional time and bandlimiting defined as in (2.14). Widom conjectured this operator would obey the trace relation

$$\lim_{\alpha\to\infty} \mathrm{tr}(T_\alpha - T_\alpha^2) = \alpha^{d-1} \log \beta \mathcal{W}_1(\delta\Lambda, \delta\Omega) + o(\alpha^{d-1} \log \alpha). \tag{2.25}$$

As shown in [67, 106], such trace relations are useful tools in determining the size of the plunge region. We will return to this in Lemma 4.16. Combined with

$$\mathrm{tr}(T_\alpha) = \left(\frac{\alpha}{2\pi}\right)^d \int_\Omega \int_\Lambda d\boldsymbol{\xi} d\boldsymbol{x}, \tag{2.26}$$

(2.25) yields a plunge region that grows at least one order slower in α than the region of ones (up to a log-factor). Moreover, the constant

$$\mathcal{W}_1(\Lambda, \Omega) = \frac{1}{2(2\pi)^{d+1}} \int_{\delta\Lambda} \int_{\delta\Omega} |\boldsymbol{n}_{\delta\Lambda}(\boldsymbol{x}) \cdot \boldsymbol{n}_{\delta\Omega}(\boldsymbol{\xi})| d\boldsymbol{\xi} d\boldsymbol{x}$$

is dependent only on the geometry of the domains Ω and Λ. This conjecture was proven roughly 30 years after its first appearance, by Sobolev [93] for arbitrary smooth domains, and the proof was later extended to piecewise continuous domains [92]. In §4.2.2 we prove a similar trace relation for the discretised higher dimensional version.

2.3 FE as approximation in a frame

Approximations on Ω involving a Fourier series on a bounding box R were studied under different names: FPIC-SU [11, 13], Fourier Continuation (FC)

[14, 15] and Fourier Extension [59]. As in §2.1, the least squares approximation is computed on an equispaced grid of collocation points.

The usefulness of approximation schemes hinges on three properties: the rate of convergence to the given function, the stability of the required computations, and the speed at which they can be computed. Considerable effort has been put into quantifying the first two properties both analytically and numerically for the one-dimensional FE. See [59] for an analysis where the FE problem is understood in terms of orthogonal polynomials, and [2, 4, 70] where it is seen in the context of frames. In some sense these results are special cases of Theorem 1.17 and Theorem 1.20, but they paint a more detailed picture that is specific for the FE. The following subsections recap the most important points.

The results distinguish between the exact discrete and continuous FE solutions $\mathcal{P}_{N_\Omega,N_\Lambda} f$ and $\mathcal{P}_{N_\Lambda} f$, and their computer-implemented counterparts. As in §1.4.1, computing the exact solutions is known to be unstable, as they can grow unbounded outside the domain of interest. Numerical algorithms however will never compute these exact solutions. Due to regularisation, the numerical FEs $\mathcal{P}^\tau_{N_\Omega,N_\Lambda} f$ and $\mathcal{P}^\tau_{N_\Lambda} f$ are more stable, while maintaining the desired convergence behaviour. As before, the regularized projections are obtained through the Truncated Singular Value Decomposition.

Remark 2.20. In [4], the analysis was carried without loss of generality for $\Omega = [-1, 1]$, $R = [-T, T]$. We will keep the shorthand $T = R/\Omega$.

2.3.1 Stability

Following [4], stability is defined in terms of the operator norm of the FE mapping

$$\kappa(\mathcal{P}_{N_\Lambda}) = \sup\{\|\mathcal{P}_{N_\Omega}(b)\| : b \in \mathbb{C}^N_\Omega, \|b\| = 1\},$$

where, with slight abuse of notation, $\mathcal{P}_{N_\Omega}(b)$ is the Fourier Extension obtained by solving the linear system (2.9) or (2.6) with right hand side b. Note the distinction from the condition number of the matrices G_N and A_N, that characterize the mapping from samples to coefficients, whereas this condition number characterizes the mapping from samples to approximations. These condition numbers can be computed for the continuous and discrete FEs, both the exact and numerical versions.

Theorem 2.21. *[4, Lemma 3.5, Theorem 5.4] The condition numbers for the exact projections $\mathcal{P}_{N_\Omega,N_\Lambda}$ and $\mathcal{P}_{N_\Lambda}$ are*

$$\kappa(\mathcal{P}_{N_\Lambda}) = E(T)^{2N_\Lambda}, \qquad \kappa(\mathcal{P}_{N_\Omega,N_\Lambda}) = D(N_\Lambda, N_\Omega),$$

where

$$E(T) = \cot^2\left(\frac{\pi}{4T}\right)$$

and

$$D(N_\Lambda, N_\Omega) = \sup\{\|\phi\|_\Omega : \phi \in \Phi_{N_\Lambda}, \sum_{x \in P_\Omega} |\phi(x)|^2 = N_\Omega/2\}.$$

Thus, it grows exponentially in N_Λ for the exact solution to the continuous problem, and is bounded for the exact solution to the discrete problem only when the quantity $D(N_\Lambda, N_\Omega)$ is bounded, which was shown to be only when $N_\Omega \sim N_\Lambda^2$. Roughly speaking, this means a Fourier series that is bounded at N_Ω equispaced points must have a degree that grows only as the square root of the number of samples. Moreover, $D(N_\Lambda, N_\Omega)$ grows exponentially for a fixed oversampling rate $N_\Omega = \varrho N_\Lambda$.

When looking at the numerical FEs the situation changes considerably.

Theorem 2.22. *[4, Theorem 4.7, Theorem 5.11]*

$$\kappa(\mathcal{P}^\tau_{N_\Lambda}) \lesssim E(T)^{-N_\Lambda}, \qquad \kappa(\mathcal{P}^\tau_{N_\Omega, N_\Lambda}) \lesssim \tau^{-a(\varrho,T))},$$

Regularizing $\mathcal{P}_{N_\Lambda}$ still leads to exponential ill-conditioning, but it is less severe than that in Theorem 2.21. The condition number of the regularized FE based on point samples is dependent on a constant $0 < a(\varrho; T) \leq 1$, independent of N_Λ, that satisfies $a(\varrho; T) \to 0$ as $\varrho \to \infty$ for fixed Ω and bounding box R. This means that for a sufficiently large oversampling factor ϱ, the condition number of the numerical FE mapping can be made reasonably close to 1; in particular, for $\Omega/R = 1/2$, oversampling by $\varrho = 2$ is sufficient.

Meanwhile, from the previous sections we know the condition number of the Gram and collocation matrices grows exponentially as $N_\Lambda \to \infty$. This is surprising, given the good condition of the FE mapping. It can be understood by noting that extensions with small coefficient norm and small residual are guaranteed to exist. The numerical algorithms will steer clear of the unstable exact solution, and instead return one of these alternatives. For a full exposition on the stability of FE calculations, see [4].

2.3.2 Convergence

Concerning convergence, Theorem 1.18 already established algebraic convergence depending on the smoothness of f. For analytic functions, convergence is limited by the region of analyticity of f in the complex plane, similar to the results for Fourier Series (Theorem 1.11) and Chebyshev Polynomials (Theorem 1.12).

This region $\Gamma(\rho)$ is a Bernstein ellipse $\Theta(\rho)$ under a transformation that allows Fourier extensions to be understood as polynomial approximations [59]. Again for $\Omega = [-1, 1]$, $R = [-T, T]$ this is

$$\Theta(\rho) = \left\{ \frac{1}{2}\left(\rho^{-1}\mathrm{e}^{\mathrm{i}\theta} + \rho\mathrm{e}^{-\mathrm{i}\theta}\right) : \theta \in [-\pi, \pi] \right\}, \tag{2.27}$$

$$\Gamma(\rho) = \left\{ \frac{\pi}{T} \arccos\left[c(T) + \frac{1 - c(T)}{2}(z + 1) \right] : z \in \Theta(\rho) \right\} \tag{2.28}$$

where $c(T) = \cos(\pi/T)$.

Theorem 2.23. *[2, Theorem 2.3, Theorem 5.4] For functions f that are analytic in $\Gamma(\rho^*)$ and bounded on the boundary, the exact FEs converge geometrically, with a speed*

$$\|f - \mathcal{P}_{N_\Lambda} f\| \leq c_f \rho^{-N_\Lambda}, \qquad \|f - \mathcal{P}_{N_\Omega, N_\Lambda} f\| \leq \sqrt{2}(1 + D(N_\Lambda, N_\Omega)) \inf_{\phi \in \Phi_{N_\Lambda}} \|f - \phi\|_\infty$$

Here $\rho = \min\{\rho^*, E(T)\}$ and c_f is proportional to $\max_{x \in \Gamma(\rho)} |f(x)|$.

For the FE from equispaced data, this theorem states that for exponential convergence of the exact solution, $D(N_\Omega, N_\Lambda)$ should be bounded. As with the condition number, constant oversampling is insufficient here.

The situation changes again when introducing the regularization.

Theorem 2.24. *[2, Section 5.3.2] If f is analytic in $\Gamma(\rho^*)$ and bounded on its boundary, than for the regularized FEs :*

1. *If $N_\Lambda < N_2$, where N_2 is a function-independent breakpoint, $\|f - \mathcal{P}^\tau_{N_\Omega, N_\Lambda} f\|$ converges or diverges exponentially fast at the same rate as the exact solution.*
2. *When $N_\Lambda \leq N_0$ (continuous) or $N_2 \leq N_\Lambda \leq N_1 := 2N_0$ (discrete), where N_0 is another function-independent breakpoint depending, both $\|f - \mathcal{P}^\tau_{N_\Lambda} f\|$ and $\|f - \mathcal{P}^\tau_{N_\Omega, N_\Lambda} f\|$ decay like ρ^{-N_Λ}.*
3. *When $N_\Lambda = N_0$ or $N_\Lambda = N_1$, the errors are approximately*

   $$\|f - \mathcal{P}^\tau_{N_\Lambda} f\| \approx c_f (\sqrt{\tau})^{d_f}, \|f - \mathcal{P}^\tau_{N_\Omega, N_\Lambda} f\| \approx c_f \tau^{d_f - a(\varrho;T)},$$

   where c_f is as before, and $d_f = \frac{\log \rho}{\log E(T)} \in (0, 1]$.
4. *When $N > N_0$ or $N > N_1$, the errors decay at least super algebraically fast down to* maximal achievable accuracies *of order $\sqrt{\tau}$ and $\tau^{1-a(\varrho;T)}$ respectively.*

This behaviour of the error offers insight into the usability of Fourier extensions. A first observation is that the continuous FE is limited to a maximal achievable accuracy of $\sqrt{\tau}$. Coupled with the need to compute Fourier integrals to compose the right hand side b in (2.6), this makes the continuous FE unfit for practical use. However, the algorithms for the discrete FE in the following chapters can be adapted to this context with little extra effort. This is documented in §3.4.1.

On the contrary, the numerical discrete FE guarantees convergence up to a certain power of τ. By varying this cutoff, the oversampling and the ratio R/Ω, the maximum achievable accuracy can be made very close to the machine precision.

Remark 2.25. The error behaviour in Theorem 2.24 seems to be contradicting a well-known result by Platte, Trefethen and Kuijlaars [83], that states that no stable procedure for approximating functions from equally spaced samples can converge exponentially for analytic functions. However, the Fourier extension from equispaced points circumvents this result by only guaranteeing convergence up to some prescribed tolerance, see [4] for a more thorough discussion.

Remark 2.26. In [4] a third type of FE was considered, the *discrete* FE. This approach consists of collocation in so-called *mapped-Chebyshev* nodes. The collocation solution using these nodes is an exact projection, under a weighted inner product, and has similar properties to the equispaced FE $\mathcal{P}_{N_\Omega,N_\Lambda}$. However, it has no known connection to Prolate Spheroidal wave theory, and in particular the collocation matrix has a spectrum that is markedly different from that in Fig. 2.2. Since the algorithms in the next chapter heavily depend on this singular value distribution, we further disregard this discrete FE.

2.3.3 Influence of the bounding box

So far we have assumed nothing about the bounding box R. The qualitative behaviour hold for any R, but it influences convergence and stability through the ratio $T = R/\Omega$ in the constants $E(T)$ and $a(\varrho, T)$, and implicitly in $D(N_\Lambda, N_\Omega)$. In this section we summarize the influence of this parameter on convergence, resolution power, and conditioning of the FE problem.

First note that the Fourier extension constant $E(T)$ grows with T. For functions analytic in a sufficiently large region, the convergence rate $\rho = \min(\rho^*, E(T))$ is limited by this constant. Thus $E(T)$ can be understood as a singularity that is introduced by the mapping in (2.28). For sufficiently analytic functions, increasing T thus increases the convergence rate, and vice versa. In particular, as $E(T) \to 1$, the FE approaches the regular Fourier series and suffers from the Gibbs phenomenon, limiting convergence speed.

The resolution power of a scheme, first studied by Gottlieb and Orszag [46] is a measure of the amount of point samples needed to resolve an oscillatory function to a certain precision.

Definition 2.27. *Let*

$$\mathcal{R}(\omega, \delta) = \min\{N_\Lambda \in \mathbb{N} : \|\mathrm{e}^{i\pi\omega x} - \mathcal{P}_{N_\Lambda}(\mathrm{e}^{i\pi\omega x})\|_\infty < \delta\}, \quad \omega > 0,$$

for some small δ. Then $\mathcal{P}_{N_\Lambda}$ has a resolution constant r if

$$\mathcal{R}(\omega, \delta) \sim r\omega, \quad \omega \to \infty.$$

For regular Fourier series, this constant has the optimal value 2. For Chebsyhev polynomials this value is π. A theoretical argument shows that for the continuous FE this resolution constant increases with T [2]. More specifically,

$$r(T) \leq 2T \sin\left(\frac{\pi}{2T}\right), \quad T \in (1, \infty).$$

Thus, for $T \approx 1$ the resolution constant $r(t) \approx 2T$ is close to optimal. When T tends to infinity, $r(T) \sim \pi$. It is even possible to optimally balance convergence speed with resolution power when aiming for a predetermined accuracy ϵ_{tol}. This is achieved by varying T with N_Λ, specifically

$$T(N_\Lambda, \epsilon_{tol}) = \frac{\pi}{4}\left(\arctan(\epsilon_{tol})^{\frac{1}{2N_\Lambda}}\right)^{-1}. \tag{2.29}$$

Although no equivalent analysis exists for the discrete FE, there have been several attempts to determine FE parameters that are in some sense optimal. In [15], Bruno et al. suggest the values $T = 2$ and $\varrho = 2$ as a general rule of thumb, but at the same time note that the optimal parameters are heavily function dependent. Note that increasing both the extension length T and the oversampling ϱ will likely increase the resolution constant r. Especially since it was shown in [2] that the limit $r(T) \sim \pi$ as $T \to \infty$ no longer holds for a discretised FE, instead the resolution constant grows as $r(T) \sim 2T$. Even though this was only observed for data points distributed as a variant of Chebyshev points, it is indicative that the resolution constant for $T = 2, \varrho = 2$ will be considerably above the optimal value.

The precise interplay between T and ϱ on the one hand, and resolution power and conditioning on the other hand was studied in detail by Adcock and Rua [5]. They found that the condition number $\kappa(\mathcal{P}_{N_\Omega, N_\Lambda})$ of the equispaced discrete FE depends only on the product $T\varrho$. Increasing either will lower the condition number. Thus as long as ϱ is increased or decreased accordingly when varying T, the conditioning of the FE mapping remains constant. This is cited as an

argument to limit T to 2, to profit from the at the time only available fast FE algorithm [69].

Furthermore, the resolution constant is also dependent on the product $T\varrho$, growing as $r(T) \sim T\varrho$. This illustrates the tradeoff between resolution power and conditioning. Numerical experiments in [5] showed that by allowing the condition number to grow from $\kappa \approx 10$ to $\kappa \approx 100$, the resolution constant was halved, while further increasing κ had very little additional value. However, it should be noted that these experiments were only carried out for $T = 2$. Lifting the restriction on T may thus offer more flexibility in finding a balance between resolution power and conditioning.

An interesting open problem raised in [5] is the possibility to vary T with M, to achieve optimal resolution power in a manner similar to (2.29). Due to the lack of a fast algorithm, any gains from varying T were considered of limited practical usability compared to the fast algorithm for $T = 2$. The fast algorithms presented in this thesis warrant a closer look at the possible benefits from this method.

2.4 FE as a method for differential equations

A lot of the interest in FEs stems from using it as a building block in ODE and PDE solvers. The use of the Fourier basis has obvious advantages, since taking derivatives is an elementwise operator on the coefficients. Taking the Fourier series on $\mathcal{L}^2_{[a,b]}$ as in §1.2.1,

$$f' = \sum_{k=-\infty}^{\infty} \hat{f}[k] \left(\mathrm{e}^{\frac{\mathrm{i}k\pi x}{b-a}} \right)' = \sum_{k=-\infty}^{\infty} \frac{\mathrm{i}k\pi}{b-a} \hat{f}[k] \mathrm{e}^{\frac{\mathrm{i}k\pi x}{b-a}}.$$

Fourier-based spectral methods are therefore popular, but mostly when the setting is smooth and periodic. The otherwise slow convergence due to the Gibbs phenomenon limits the usability. Using the Fourier Extension technique eliminates the periodicity requirements.

Bruno and Pohlmann used this to their advantage in higher dimensions to obtain smooth and periodic extensions around boundaries of complicated surfaces. This was the basis for the very efficient FC-Gram method for approximations and solving differential equations, using 1D Fourier extensions combined with an ADI approach [16, 72, 6]. In this method they do not compute $\mathcal{P}_{N_\Omega, N_\Lambda} f$, which in their naming scheme is the FC(SVD) method, but instead use precomputed extensions based on Gram polynomials, that are matched to the data at the ends of the interval. This leads to a scheme of high, but finite order.

The FE method is closely related to embedded or fictitious domain methods for solving certain partial differential equations using Fourier basis functions, the main difference being the approximation in the extension region. In embedded domain methods the function is explicitly extended outside the domain of interest, e. g. through convolution with Gaussian kernels [17] or using polynomial corrections [82]. In the Fourier extension technique, the approximation in the extension region is determined implicitly through solving a least squares problem.

Chapter 3

Fast algorithms for the one-dimensional Fourier Extension

In this chapter we formulate two distinct fast $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$ algorithms for the computation of a one-dimensional Fourier Extension from oversampled equispaced points, following §2.1. Most of this chapter was previously published as [73].

3.1 Existing approaches

The need for a dedicated algorithm becomes apparent when trying to solve a least squares system such as (2.9) and (2.6) using more general linear algebra methods. The most straightforward approach to the regularised projections is to calculate the SVD directly, and explicitly form the truncated pseudo-inverse (1.23). The cost of this Singular Value Decomposition is $\mathcal{O}(N_\Lambda^3)$ with constant oversampling. For larger N_Λ this quickly becomes unfeasible.

A possible solution is the use of iterative methods such as LSQR [80], since a single iteration consists of applying A and A^* and can be performed in $\mathcal{O}(N_\Lambda \log N_\Lambda)$ operations. However, the estimate for the residual after m iterations is

$$\|Ax_m - b\| \leq \left(\frac{\sigma_1 - \sigma_n}{\sigma_1 + \sigma_n}\right)^m \|Ax_0 - b\|, \tag{3.1}$$

with x_m the solution at step m and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{N_\Lambda}$ the singular values of A. Per the previous chapter, the matrix A from (2.9) is severely ill-conditioned even for moderate N_Λ. Although different alterations can be made to the iterative routines to deal with nearly singular matrices, see [8] for an example, all these methods have great difficulty dealing with the smallest singular values.

We further note that solving the ill-conditioned system using a general solver like MATLAB's backslash may return something closer to $\mathcal{P}^{\tau}_{N_\Omega,N_\Lambda}$ than $\mathcal{P}_{N_\Omega,N_\Lambda}$. The method MATLAB selects to solve this least squares system is the pivoted QR method [45], that has implicit regularisation of near singular systems. Even though this is a very straightforward approach, it has the same $\mathcal{O}(N_\Lambda^3)$ cost as the SVD.

A $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$ fast algorithm was presented in [69] by Lyon for the specific case where $T = R/\Omega = 2$. The algorithm is based on the observation that the singular values of A behave as in Fig. 2.2, clustering near 1 and then falling rapidly. Based on symmetries present when $T = 2$, the solution was decoupled into the sum of a sine series and a cosine series. Using several projectors, it was shown suitable sine series coefficients can be found by solving a low-rank system, with rank observed to be $\mathcal{O}(\log N_\Lambda)$. The coefficients of the cosine series then follow from a single application of the DCT.

The algorithms from the following sections, though motivated differently, have a similar structure to that in [69]. The singular value distribution lends itself to a decoupling of the problem into a subproblem with low rank, and a subproblem with good conditioning.

3.1.1 Randomised algorithms for low rank systems

The low-rank system in [69] was solved using a randomised approach, that is well suited for systems with low-rank that are sparse, or where the system matrix A can be applied efficiently [68, 108]. The algorithm consists of applying A to r random vectors, collected in W. They showed that if $r = \text{rank}(A) + k$, then the range of $\tilde{A} = AW$ is equal to the range of A with probability extremely close to 1, being $1 - 1\text{e}^{-17}$ when $k = 20$. They then proceed by calculating an SVD of the matrix $\tilde{A}$ and using that to construct an approximate pseudo-inverse of A. The solution is then obtained by multiplying the intermediate solution with W. The algorithm is outlined in Algorithm 1, with the computational cost for each step. It is assumed that A is an $N_\Lambda \times N_\Lambda$ matrix that can be applied in $\mathcal{O}(N_\Lambda \log N_\Lambda)$ operations, with $\text{rank}(A) \ll N$. The computational complexity is $\mathcal{O}(N_\Lambda r^2 + N_\Lambda r \log N_\Lambda)$. If r is constant, this algorithm operates in $\mathcal{O}(N_\Lambda \log N_\Lambda)$ operations. If $r = \mathcal{O}(\log N_\Lambda)$, as will be the case in §3.4, the cost is $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$.

Algorithm 1 Solution of $Ax = b$, where A has rank r.

$W = \text{rand}(N_\Lambda, r + 20)$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$\tilde{A} = AW$	$\triangleright\ \mathcal{O}(rN_\Lambda \log N_\Lambda)$
$USV^* = \tilde{A}W$	$\triangleright\ \mathcal{O}(N_\Lambda r^2)$
$y = V(S_\tau^\dagger(U^* b))$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$x = Wy$	$\triangleright\ \mathcal{O}(N_\Lambda r)$

3.2 Isolating the plunge region

In this chapter we focus on the one-dimensional case, and we assume $R = [-T, T]$ and $\Omega = [-1, 1]$. From §2.1, the FE matrix is

$$A_{kj} = \frac{1}{\sqrt{N_R}} \phi_j(x_k), \qquad \phi_j(x) = \mathrm{e}^{\mathrm{i}jx\pi/T}.$$

The SVD of this matrix can be expressed in terms of the P-DPSS

$$A = U\Sigma V^*, \qquad u_i = \varphi_{\Omega,\Lambda,i}, \quad v_i = \varphi_{\Lambda,\Omega,i}. \tag{3.2}$$

The cost of this full SVD is prohibitively large, so to improve upon this cost two subproblems are identified and solved in succession. The key to this division is in the distribution of the singular values μ, as shown in Fig. 2.2. Recall from §2.1 that for a specific τ there are three distinct regions:

- A region $I_\alpha := \{\mu : 1 > \mu > 1 - \tau\}$ where all singular values are 1 up to a tolerance τ. This region contains approximately $N_\Omega N_\Lambda / N_R$ singular values.
- A region $I_\beta := \{\mu : 1 - \tau \geq \mu > \tau\}$, also referred to as the "plunge region". This region grows as $\mathcal{O}(\log N_\Lambda)$, as $N_\Lambda \to \infty$.
- A region $I_\gamma := \{\mu : \tau \geq \mu > 0\}$ where the singular values further decay exponentially.

Recall the subdivision of the Singular Value Decomposition from §2.1

$$A = \begin{bmatrix} U_\alpha & U_\beta & U_\gamma \end{bmatrix} \begin{bmatrix} \Sigma_\alpha & & \\ & \Sigma_\beta & \\ & & \Sigma_\gamma \end{bmatrix} \begin{bmatrix} V_\alpha & V_\beta & V_\gamma \end{bmatrix}^*.$$

Since I_γ contains exactly those singular values below the cutoff τ, the T-SVD solution to the problem $Ax = b$ with truncation parameter τ is then

$$x = x_\alpha + x_\beta = V_\alpha \Sigma_\alpha^{-1} U_\alpha^* b_\alpha + V_\beta \Sigma_\beta^{-1} U_\beta^* b_\beta. \tag{3.3}$$

where the inverse operator applies to the diagonal elements. The right hand side is split along the orthogonal spans of U_α, U_β and U_γ, i.e.,

$$b_\alpha = U_\alpha U_\alpha^* b, \qquad b_\beta = U_\beta U_\beta^* b, \qquad b_\gamma = U_\gamma U_\gamma^* b.$$

Note that the T-SVD method implicitly requires

$$\|b_\gamma\|/\|b\| < \tau. \tag{3.4}$$

If this is not satisfied, b is not in the *numerical* range of A. In this case, we say the Fourier extension has not yet converged, and N_Λ should be increased. We further require b to satisfy the so-called *discrete Picard condition* in ill-posed problems [53].

Definition 3.1. *Let $\gamma_i = u_i^* b$ be the inner product of the singular vectors with the right hand side. Let τ be the truncation parameter. Then b is said to satisfy a Discrete Picard condition if for all $\mu_i > \tau$ the corresponding γ_i decay to zero faster than the μ_i, on the average.*

This together with (3.4) guarantee existence of a solution x with reasonable norm compared to b, such that $\|Ax - b\| < \tau$. This relates to Theorems 1.17 and 1.20, in that convergence is dependent on the existence of a solution with small norm. We will return to this condition in §3.6.

(3.3) splits the problem into two orthogonal subproblems. The solution corresponding to the plunge region singular vectors x_β can be found independently from x_α. Once it has, obtaining x_α is then straightforward, based on the following observation:

$$\begin{aligned} A^*(b - b_\beta) &= V_\alpha \Sigma_\alpha U_\alpha^* b_\alpha + V_\gamma \Sigma_\gamma U_\gamma^* b_\gamma \\ &= V_\alpha \Sigma_\alpha^{-1} U_\alpha^* b_\alpha + \mathcal{O}(\tau) \\ &= x_\alpha + \mathcal{O}(\tau). \end{aligned}$$

The b_γ term, which is already assumed to be negligible, is fully eliminated by the additional $\mathcal{O}(\tau)$ factor Σ_γ. Noting that due to the definition of I_α

$$\Sigma_\alpha = \Sigma_\alpha^{-1} + \mathcal{O}(\tau),$$

x_α can be found at the cost of a single (fast) multiplication with A^* ($\mathcal{O}(N_\Lambda \log N_\Lambda)$). The two approaches given in §§3.3 and 3.4 differ in how they isolate V_β, Σ_β and U_β from the plunge region and how they efficiently calculate x_β.

3.3 Explicit eigenvectors

Due to the intrinsic connection of the FE problem with DPSS, it is possible to explicitly compute U_β, Σ_β and V_β^*. The second order difference equation from (2.23) implies that

$$Z_{N_\Omega,N_\Lambda}\varphi_{N_\Omega,N_\Lambda,i} = \theta_i \varphi_{N_\Omega,N_\Lambda,i}, \qquad Z_{N_\Omega,N_\Lambda} = \begin{bmatrix} c_0 & b_1 & & \\ b_1 & c_1 & \ddots & \\ & \ddots & \ddots & b_{N_\Lambda-1} \\ & & b_{N_\Lambda-1} & c_{N_\Lambda-1} \end{bmatrix}$$

$$b_k = \sin\left(\frac{\pi k}{N_R}\right)\sin\left(\frac{\pi(N_\Omega - k)}{N_R}\right)$$

$$c_k = -\cos\left(\frac{\pi(2k-1-N_\Omega)}{N_R}\right)\cos\left(\frac{\pi N_\Lambda}{N_R}\right).$$

The P-DPSS can thus be found as eigenvectors of a tridiagonal matrix. The $\varphi_{N_\Omega,P_\Lambda,i}$ that make up U in (3.2) are eigenvectors of Z_{N_Ω,N_Λ}. The dual matrix Z_{N_Λ,N_Ω} yields the right singular vectors V. With the P-DPSS known, the original singular value μ_i is found in $\mathcal{O}(N_\Lambda \log N_\Lambda)$ operations as

$$\mu_i = \varphi^*_{\Omega,\Lambda,i} A \varphi_{\Lambda,\Omega,i}$$

This approach is already in use for the regular DPSS [48, 57].

To find x_β however, we are only interested in a subset of the singular values and vectors. Since the number of singular values in the plunge region grows only logarithmically, we require algorithms that calculate k specific eigenvalues and eigenvectors of a tridiagonal matrix Z in $\mathcal{O}(kN_\Lambda)$ operations [31, 30]. Algorithms for this computation require as input for the desired eigenvalues θ_i of Z either:

- A range $[C_1, C_2]$. The algorithm then finds the set
$$\{(v_i, \theta_i) : C_1 \leq \theta_i \leq C_2, Zv_i = \theta_i v_i\}$$
- An index set $\{i_{\min}, \ldots, i_{\max}\}$. The algorithm then finds
$$\{(\theta_i, v_i) : \theta_{i_{\min}} \leq \theta_i \leq \theta_{i_{\max}}, Zv_i = \theta_i v_i\},$$
from the ordered set $\theta_0 \geq \theta_1 \geq \ldots$.

The algorithms thus require knowing the θ_j, or alternatively the indices j, that correspond to $\mu \in I_\beta$. Denote the mappings between μ_i and θ_j, and between their indices i and j, by

$$\theta_j = \mathcal{M}_{N_\Lambda,N_\Omega}(\mu_i), \quad j = \mathcal{M}^*_{N_\Lambda,N_\Omega}(i) \Leftrightarrow \begin{cases} \varphi_i^* Z_{N_\Lambda,N_\Omega} \varphi_i = \theta_j \\ \varphi_i^* A_{N_\Lambda,N_\Omega} \varphi_i = \mu_i. \end{cases}$$

Getting a qualitative understanding of $\mathcal{M}_{N_\Lambda,N_\Omega}$ is difficult, as very little is known of this mapping. We will only show monotonicity, a trait shared with the PSWF and DPSWF equivalents (recall from (2.17) and Property 2.11 that these also satisfy second order diffferential or difference equations, with different eigenvalues). The monotonicity property is already very helpful, since it implies that $\mathcal{M}^*_{P_\Lambda,N_\Omega}(i) = i$.

Theorem 3.2. *If $N_\Omega \geq N_\Lambda$, the mapping $\mathcal{M}'_{N_\Lambda,N_\Omega}$ is monotone,*

$$\forall i_1, i_2 : i_1 > i_2 \Leftrightarrow \mathcal{M}^*_{N_\Lambda,N_\Omega}(i_1) > \mathcal{M}^*_{N_\Lambda,N_\Omega}(i_2).$$

Proof. The proof follows a mechanism used by Slepian in both [91, p. 61] and [89, §4.1]. The continuity of eigenvalues and eigenvectors as a function of a parameter, combined with a known ordering result for a specific value of this parameter, extends the known result to all parameter values.

First we recall a similar result from the continuous Fourier Extension. Let G_{N_Λ} be the Gram matrix from the continuous FE problem (2.6) with eigenvalues λ_i, and χ_{N_Λ} the tridiagonal matrix (3.5) with eigenvalues $\bar{\theta}_i$, $i = 1, \ldots, N_\Lambda$

$$\chi_{N_\Lambda} = \begin{bmatrix} \bar{c}_0 & \bar{b}_1 & & \\ \bar{b}_1 & \bar{c}_1 & \ddots & \\ & \ddots & \ddots & \bar{b}_{N_\Lambda-1} \\ & & \bar{b}_{N_\Lambda-1} & \bar{c}_{N_\Lambda-1} \end{bmatrix}, \tag{3.5}$$

$$\bar{c}_i = \left(\frac{N_\Lambda - 1}{2} - i\right)^2 \cos\frac{\pi}{T}, \qquad \bar{b}_i = \frac{i(N_\Lambda - i)}{2}. \tag{3.6}$$

Then the mapping $\bar{\mathcal{M}}'$

$$j = \bar{\mathcal{M}}^*(i) \Leftrightarrow \begin{cases} \psi_i^* \chi_{N_\Lambda} \psi_i = \bar{\theta}_j \\ \psi_i^* G_{N_\Lambda} \psi_i = \lambda_i \end{cases}$$

was proven to be monotone in [89, §4.1]. This result can be extended to the discrete FE case, since the limits of the entries of A^*A and Z_{N_Λ,N_Ω} for large

N_Ω can be written in terms of the corresponding matrices of the continuous problem:

$$\lim_{N_\Omega\to\infty}(A^*A)_{ij} = \frac{\sin\frac{(i-j)\pi}{T}}{\pi(i-j)} = (G_{N_\Lambda})_{ij}$$

$$\lim_{N_\Omega\to\infty} b_k = \frac{\pi^2 k(N_\Lambda - k)}{N_R^2} = \frac{2\pi^2}{N_R^2}\bar{b}_k$$

$$\lim_{N_\Omega\to\infty} c_k = \cos\frac{\pi}{T}\left(\frac{2\pi^2}{N_R^2}\left(\frac{N_\Lambda - 1}{2} - k\right)^2 - 1\right) = \frac{2\pi^2}{N_R^2}\bar{c}_k - \cos\frac{\pi}{T}$$

$$\lim_{N_\Omega\to\infty} Z_{N_\Omega,N_\Lambda} = \frac{2\pi^2}{N_R^2}\chi_{N_\Lambda} - \cos\frac{\pi}{T}I$$

The last line ensures that the eigenvalues of Z_{N_Ω,N_Λ} are those of χ_{N_Λ} under a linear mapping. Since this mapping preserves the ordering of eigenvalues, the theorem holds in the limit $N_\Omega \to \infty$. Further, when limited to the regime $T > 1, N_\Omega \geq N_\Lambda$, we have the following:

1. The tridiagonal matrix $Z_{N_\Omega,N_\Lambda}(N_\Omega)$ with diagonal elements c_k and subdiagonal elements b_k commutes with A^*A for integer values of N_Ω. However, by a substitution as in [20, Thm. 4.7], it is easy to see that this relationship holds for any real $N_\Omega > N_\Lambda$.

2. A classical result states that the eigenvalues of a matrix are continuous as the matrix entries vary continuously in the parameter. Thus all $\theta_i(N_\Omega)$ are continuous. In general they may coincide with each other. However, since all subdiagonal entries are always non-zero,

$$b_k^2 > 0, \qquad 1 \leq k \leq N_\Lambda - 1$$

   $Z_{N_\Omega,N_\Lambda}(N_\Omega)$ is a so-called normal Jacobi matrix and such matrices are known to have distinct eigenvalues [42, Ch. 2.1]. As a result of this distinctness, the eigenvectors can be chosen to be continuous in N_Ω as well [64, Ch. 2 §5.3].

3. To prove similar statements for the matrix A^*A, we note that the inductive proof in [110, Prop. 5] for the distinctness of eigenvalues of A^*A is dependent only on it being symmetric, and the commuting $Z_{N_\Omega,N_\Lambda}(m)$ being a normal Jacobi matrix. The eigenvalues $\mu_i(N_\Omega)$ are thus continuous and distinct $\forall N_\Omega \geq N_\Lambda$. Then the eigenvectors can also be chosen continuous in N_Ω.

Combining these statements, the distinctness preserves the relative ordering of the continuous eigenvalues. The continuity of the eigenvectors relates the eigenvalues of A and Z_{N_Λ,N_Ω} through the mapping $\mathcal{M}'$. Since this mapping is monotone in the limit $N_\Omega \to \infty$, the continuity ensures the mapping is monotone for all $N_\Omega \geq N_\Lambda$. □

The required index set for the eigenvalues of Z_{N_Ω,N_Λ} is exactly the index set corresponding to μ_i from the plunge region. From Property 2.15 these are known to be centered around $\frac{N_\Lambda N_\Omega}{N_R}$, and their number grows as $\mathcal{O}(\log N_\Lambda)$. All that remains is to determine the constants $C_{\min}$ and $C_{\max}$ so that

$$\mu_j \in I_\beta \Leftrightarrow j \in \left[\frac{N_\Lambda N_\Omega}{N_R} - C_{\min}\log N_\Lambda, \frac{N_\Lambda N_\Omega}{N_R} + C_{\max}\log N_\Lambda\right].$$

We denote the minimum and maximum indices as $j_{\min}$ and $j_{\max}$. The minimum required index set $[j_{\min}, j_{\max}]$ with cutoff $\tau = 1e-16$ for increasing N_Λ is shown in Figure 3.1, with the value $\frac{N_\Lambda N_\Omega}{N_R}$ as a dashed line. Experimentally, the choices $C_{\min} \geq 6$ and $C_{\max} \geq 3$ are deemed sufficient for all $\tau \geq \epsilon_{mach}$. This is slightly more optimistic than the best known nonasymptotic bounds in [111]. The V_β can be obtained by refining $A^*\varphi_{N_\Omega,N_\Lambda,i}$ as eigenvectors of Z_{N_Λ,N_Ω}.

Using a fast tridiagonal eigenvector algorithm, U_β, Σ_β and V_β can be computed in $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$ operations. The solution term

$$x_\beta = V_\beta \Sigma_\beta^{-1} U_\beta^* b$$

is then found in $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$ operations.

Remark 3.3. Monotonicity of the map $\mathcal{M}_{N_\Lambda,N_\Omega}$ is also observed to hold for integer N_Ω smaller than integer N_Λ. Specifically, a variant of Theorem 3.2 holds for the N_Ω nonzero singular values μ_i. The same index set can thus be used to compute both U_β and V_β with a fast tridiagonal eigenvalue algorithm. However, this mapping is unproven.

Combining x_β with the calculation of x_α described at the end of §3.3, the end result is a fast $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$ algorithm. A bare-bones version incorporating Remark 3.3 is given in Algorithm 2. The *trideig* function should return eigenvectors of a tridiagonal matrix given an index interval, e.g. the LAPACK routine *dstevx*, and are found at a cost of $\mathcal{O}(N_\Lambda \eta(\tau, N_\Lambda))$ operations for $\eta(\tau, N_\Lambda)$ eigenpairs [30].

Figure 3.2 shows the different stages of the algorithm, which easily relates the intermediate results to the projection on the P-DPSS shown in Fig. 2.3. The coefficients x_β are the orthogonal projection onto the plunge region singular

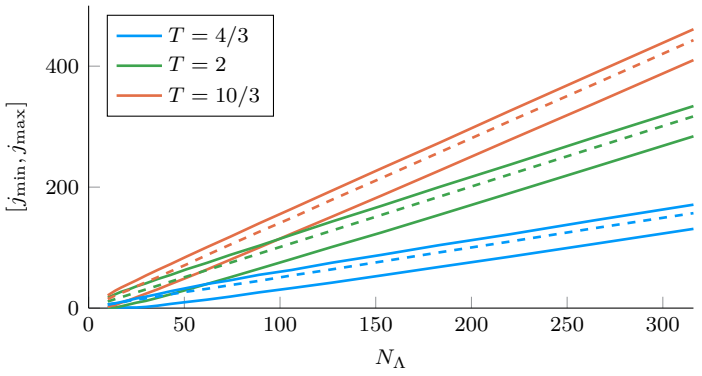


Figure 3.1: The behaviour of the index set of the plunge region. The minimal and maximal index of the plunge region are shown as solid lines, for different values of T. The point $N_\Lambda N_\Omega / N_R$, which is known to lie in the interval, is shown as a dashed line.

Algorithm 2 Explicit calculation of P-DPSS

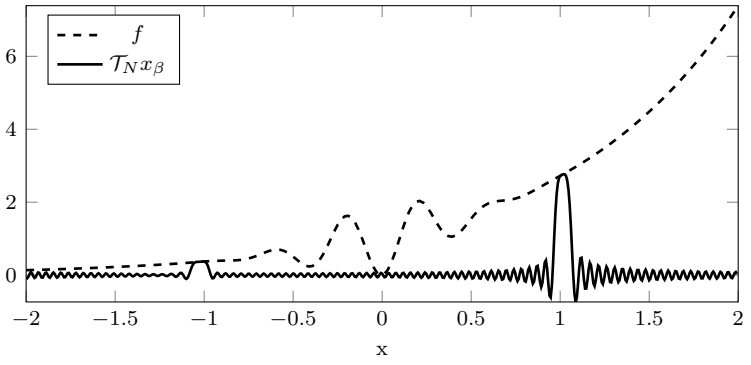
$U_\beta = \text{trideig}(Z_{N_\Omega, N_\Lambda}, \{j_{\min}, j_{\max}\})$	$\triangleright\ \mathcal{O}(N_\Lambda \log N_\Lambda)$
$V_\beta = \text{trideig}(Z_{N_\Lambda, N_\Omega}, \{j_{\min}, j_{\max}\})$	$\triangleright\ \mathcal{O}(N_\Lambda \log N_\Lambda)$
$\Sigma_\beta = U_\beta^* A V_\beta > \tau$	$\triangleright\ \mathcal{O}(N_\Lambda \log^2 N_\Lambda)$
$x_\beta = V_\beta \Sigma_{\beta,\tau}^\dagger U_\beta^* b$	$\triangleright\ \mathcal{O}(N_\Lambda \log N_\Lambda)$
$x_\alpha = A^*(b - A x_\beta)$	$\triangleright\ \mathcal{O}(N_\Lambda \log N_\Lambda)$
$x = x_\alpha + x_\beta$	$\triangleright\ \mathcal{O}(N_\Lambda)$

vectors, and approximates the function well at the boundary, since P-DPSS corresponding to both I_α and I_γ are small there. The residual $b - Ax_\beta$ therefore smoothly vanishes at the boundaries. Since A^* is equivalent to extending by zero and taking the Fourier transform on the interval $[-T, T]$, x_α is obtained immediately.

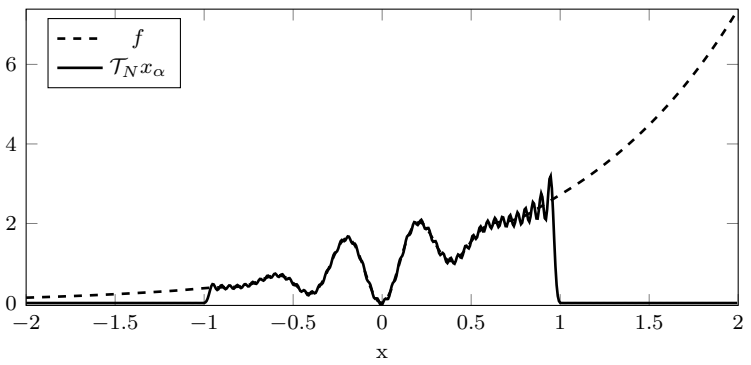
3.4 An implicit algorithm

The second approach to calculating x_β is more general, since it depends solely on the steep singular value profile associated with Prolate Spheroidal Wave functions and their generalisations, illustrated in Fig. 2.2. As such, it is extensible to any frame or ill-conditioned Riesz basis that exhibits a similar profile.

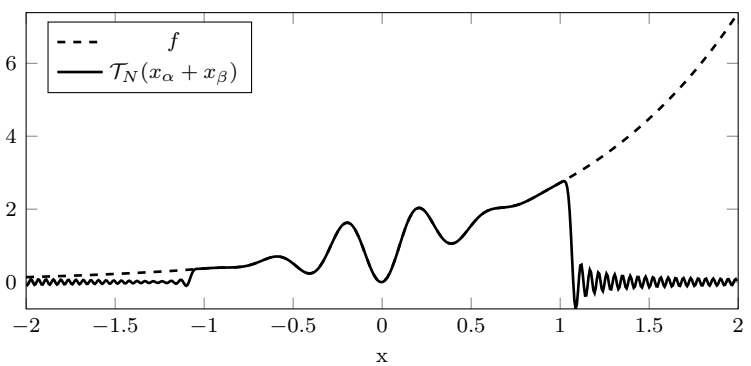
This algorithm finds a solution x_β as in (3.3), without explicitly computing the



(a) $\mathcal{T}_N x_\beta(x)$



(b) $\mathcal{T}_N x_\alpha(x)$



(c) $\mathcal{T}_N(x_\alpha + x_\beta)(x)$

Figure 3.2: Illustration of the different intermediate results in Algorithm 2. x_β represents the solution at the boundary. The residual then vanishes smoothly at the boundary, yielding x_α through extension by zero and the FFT.

V_β, hence the name. It is based on the observation that multiplying both the FE matrix A and right hand side by a factor

$$P = (AA^* - I) \tag{3.7}$$

isolates the problem to the plunge region. This is easily seen from the SVD. With $A = U\Sigma V^*$ we have

$$P = U(\Sigma^2 - I)U^*, \tag{3.8}$$

and

$$PA = U(\Sigma^3 - \Sigma)V^*.$$

Note here that the mapping

$$\mathcal{W}(\mu) = \mu^3 - \mu$$

isolates the singular values from the plunge region since $\forall \mu \in \{I_\alpha \cup I_\gamma\} : \mathcal{W}(\mu) = \mathcal{O}(\tau)$. This way, PA preserves the singular vectors of just the plunge region, but with mapped singular values

$$PA = U_\beta(\Sigma_\beta^3 - \Sigma_\beta)V_\beta^* + \mathcal{O}(\tau). \tag{3.9}$$

In theory, P is a square full rank matrix, and solving

$$PAx = Pb \tag{3.10}$$

is equivalent to solving $Ax = b$. In practice, PA has a large numerical nullspace. Hence, PA is approximately low rank, with the rank increasing with the size of the plunge region. The combination of this low rank with a fast matrix-vector product allows random matrix algorithms as in §3.1.1 to solve (3.10) very efficiently.

Assume W a uniform random matrix of dimensions $N_\Lambda \times r$, where $r = C \log N_\Lambda + D$ is a conservative estimate for the rank of PA. From the previous section, $C = C_{\min} + C_{\max} \geq 9$ is sufficient, with $D \sim 20$ ensuring a negligible probability of failure of the random matrix algorithm. Solving the following small linear system

$$PAWy = Pb$$

and letting

$$x_W = Wy$$

one obtains a solution to (3.10) a cost of $\mathcal{O}(N_\Lambda r^2)$. It follows from (3.9) that x_β is recovered to high accuracy, with high probability. On the other hand, this

solution process introduces additional solution terms from the nullspace of PA. Write x_W as $x_\beta + s_\alpha + s_\gamma$. Then as before we calculate

$$\begin{aligned} A^*(b - Ax_W) &= A^*(b_\alpha - As_\alpha - As_\gamma) + \mathcal{O}(\tau) \\ &= x_\alpha - s_\alpha - V_\gamma \Sigma_\gamma^2 V_\gamma^* s_\gamma + \mathcal{O}(\tau) \\ &= x_\alpha - s_\alpha + \mathcal{O}(\tau). \end{aligned}$$

Then $x = x_W + A^*(b - Ax_W)$ reduces to

$$x = x_\alpha + x_\beta + s_\gamma + \mathcal{O}(\tau).$$

so that $||Ax - b|| = \mathcal{O}(\tau)$. See §3.6 for a more precise characterisation of the residual. The total cost of this algorithm is again $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$ operations. A pseudocode version is given in Algorithm 3. Note the similarities to both Algorithm 1 and Algorithm 2.

Algorithm 3 Implicit projection on P-DPSS

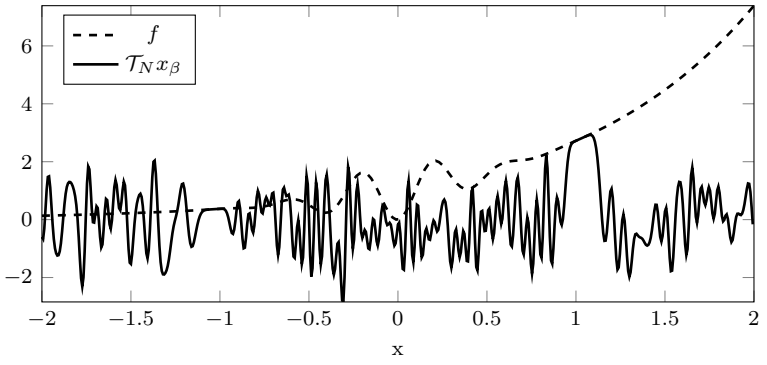
$W = \text{rand}(N_\Lambda, r + 20)$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$\tilde{A} = (AA^* - I)AW$	$\triangleright\ \mathcal{O}(rN_\Lambda \log N_\Lambda)$
$USV^* = \tilde{A}$	$\triangleright\ \mathcal{O}(N_\Lambda r^2)$
$y = V(S_\tau^\dagger(U^*((AA^* - I)b)))$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$x_W = Wy$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$x_\alpha = A^*(b - Ax_\beta)$	$\triangleright\ \mathcal{O}(N_\Lambda \log N_\Lambda)$
$x = x_\alpha + x_\beta$	$\triangleright\ \mathcal{O}(N_\Lambda)$

Figure 3.3 shows an interpretation of the intermediate results similar to Fig. 3.2. However, here x_W is seen to contain elements from the numerical nullspace of PA. The solution at the boundary is still recovered exactly, and the residual thus vanishes at the boundary. The nullspace elements corresponding to I_γ persist in the solution, but do not influence accuracy on $\Omega = [-1, 1]$.

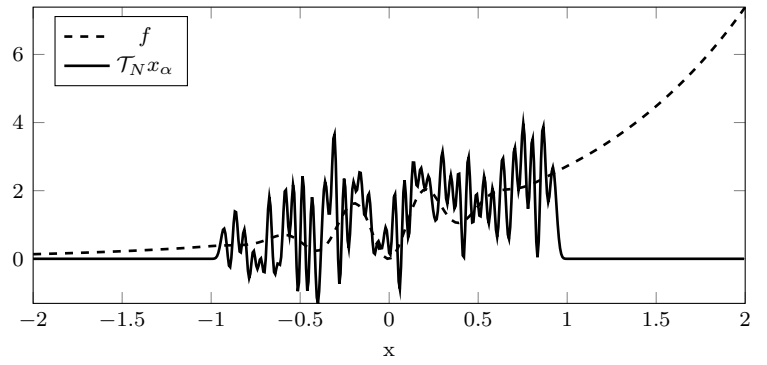
Remark 3.4. Algorithms 2 and 3 have a pre-computational step that approximates $A_\tau^\dagger$ in some sense, the first three steps in both cases. The right hand side is only introduced into the computation after this pre-computation. Notably, these last steps have a complexity of only $\mathcal{O}(N_\Lambda \log N_\Lambda)$. Storing an approximate decomposition of A_β in memory can thus provide a significant speedup when multiple right hand sides need to be approximated.

3.4.1 Adaptation for the continuous FE

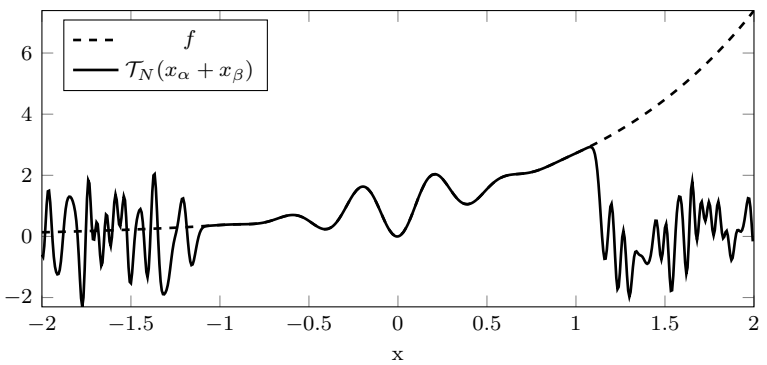
From section 2.2.2, the continuous FE has connections with Prolate Spheroidal Wave theory as well. In particular, the eigenvalues λ_i have a similar profile to



(a) $\mathcal{T}_N x_\beta$



(b) $\mathcal{T}_N x_\alpha$



(c) $\mathcal{T}_N(x_\alpha + x_\beta)$

Figure 3.3: Illustration of the different intermediate results in Algorithm 3. x_W represents the solution at the boundary, with added elements from the nullspace. As before, the residual vanishes smoothly at the boundary.

that of Fig. 2.2. Our second approach thus applies immediately, and Algorithm 3 is well suited to solve the continuous FE problem. Note however that this does not eliminate the theoretical $\mathcal{O}(\sqrt{\tau})$ error bound.

Furthermore, these DPSS also satisfy a second order difference equation, different from (2.23). In particular, the matrix G_{N_Λ} commutes with the tridiagonal matrix (3.5). It follows that, with minor modifications, Algorithm 2 can also be used to solve the continuous FE problem.

3.5 Numerical Results

In this section we apply the algorithms from the previous sections to a number of test problems. Most tests were performed in JULIA, as described in Chapter 6. The timing test was performed in MATLAB, single threaded, to compare with MATLAB implementations of other algorithms. The required fast matrix-vector products Ax and A^*y were implemented using ffts, and the LAPACK routine *dstevx* was used for the tridiagonal eigenvalue problem.

To show the validity of our algorithms for different values of T, experiments are carried out for

$$T_1 = 1.1, \quad T_2 = 2, \quad T_3 = 3.8.$$

Following [5], the product $N_R = 2T\gamma N_\Lambda$ is held constant when varying T in order to maintain a fixed condition number. This means the oversampling $\varrho = N_\Omega/N_\Lambda$ varies between experiments. The cutoff was $\tau = 10^{-14}$ unless mentioned otherwise.

3.5.1 Computational complexity

Figure 3.5.1 shows execution time for increasing degrees of freedom of the algorithm for different values of T. The figure confirms the $\mathcal{O}(N_\Lambda \log^2 N_\Lambda)$ asymptotic complexity of our algorithms. It also shows execution speed is on par with the previous fast algorithm [69].

3.5.2 Convergence

The accuracy of the solution obtained through our algorithms is shown in Figs. 3.5 to 3.8. Algorithm 2 is shown in dashed green lines, Algorithm 3 in solid blue lines. The accuracy is measured as both the maximum pointwise error over an equispaced grid, sampled ten times denser than the one used for

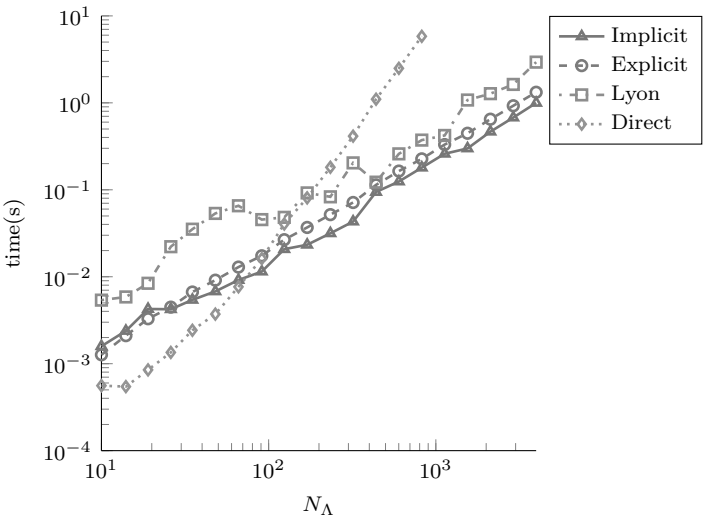


Figure 3.4: Execution time for increasing degrees of freedom N_Λ, for the explicit and implicit algorithms (Algorithms 2 and 3), the algorithm by Lyon [69] and a direct solver.

construction, and the relative residual error when solving the system. This is measured for increasing number of degrees of freedom N_Λ for four test functions:

- A well-behaved, smooth function to show convergence in near optimal conditions,
$$f_1(x) = x^2.$$
- An oscillatory function, to show the resolution power of Fourier extensions for oscillatory functions,
$$f_2(x) = \mathrm{Ai}(67x).$$
- The Runge example, a function with two poles in the complex plane at $\pm \mathrm{i}/5$. This function is notoriously difficult to interpolate in equispaced points using polynomials,
$$f_3(x) = \frac{1}{1+25x^2}.$$
- A function with a discontinuity, to test convergence in this case,
$$f_4(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}.$$

The convergence behaviour seen is in accordance with the results from [59, 4, 5], as summarised in §2.3.2. For $T = 2$, it also very closely agrees with the Lyon algorithm (not shown). Convergence for functions analytic in $[-1, 1]$ is at least geometric (Fig. 3.5), even when singularities are present near the real interval (Fig. 3.7). Following the earlier arguments about resolution power from 2.3.3, Fourier extensions of oscillatory functions start to converge sooner for lower values of T (Fig. 3.6). Note that this is in terms of degrees of freedom N_Λ, and that we increased the oversampling for lower T to maintain conditioning.

When the function has a discontinuity the Fourier coefficients decay as $O(N_\Lambda)$. Following §1.2 we expect no pointwise convergence, as is seen in Fig. 3.8b. The residual is an approximation to the $\mathcal{L}^2$ norm, and shows $\mathcal{O}(N_\Lambda^{-1/2})$ convergence as expected (Fig. 3.8a).

3.5.3 Robustness

To ensure the algorithms are robust for large N_Λ, Figs. 3.9a and 3.9b show successive approximations of

$$f(x) = \sin(10x)$$

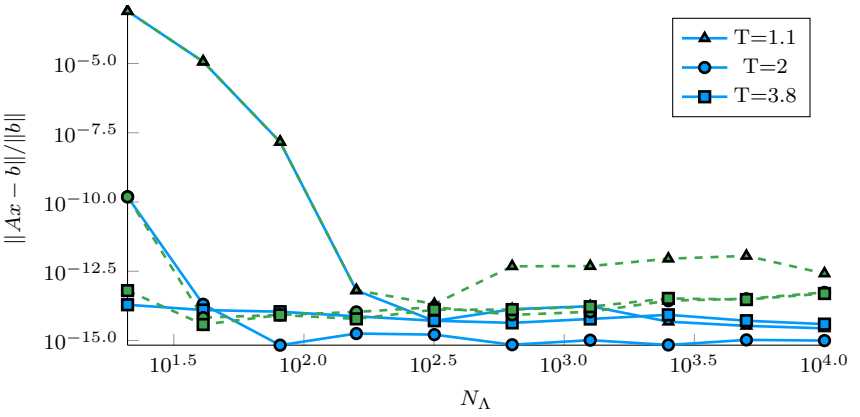
and

$$f(x) = \sin(N_\Lambda x/2), \tag{3.11}$$

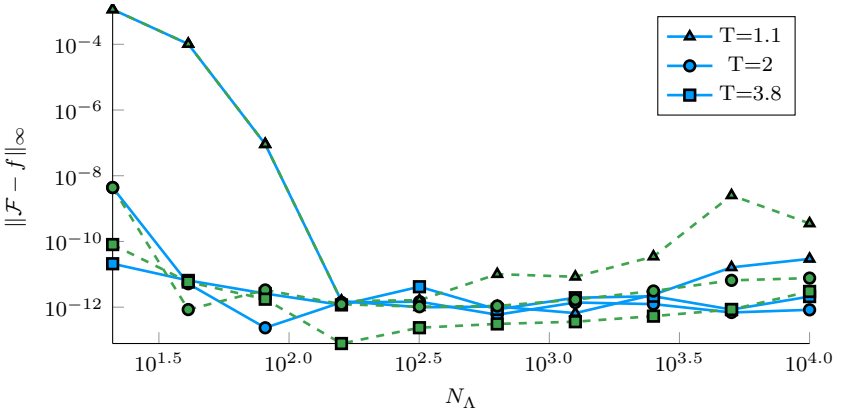
with the error measured as the relative residual. Figure 3.9a shows that at least for Algorithm 3, the residual stays near τ when increasing the frequency. Figure 3.9b shows the approximation of an increasingly oscillatory function that is right at the limit of the approximation power. For $T = 3.8$, the maximum frequency in the Fourier basis is lower than the frequency of the signal for every N_Λ, so there cannot be any convergence. For the other values of T the error stays close to τ, but increases slightly with N_Λ. One possible explanation lies in the constant c_f in Theorem 2.24 that determines the accuracy at the point where geometric convergence breaks down and superalgebraic convergence sets in. This constant is function-dependent, and grows with N_Λ for (3.11). See also [1, Proposition 5.8], where the Sobolev norm of the approximant appears in the error estimates directly. However, further research into highly oscillatory functions is needed.

3.6 Influence of noise

In this last subsection we perform a slightly more thorough error analysis on Algorithm 3 than that of §3.4, that was published in [73]. In exact arithmetic,

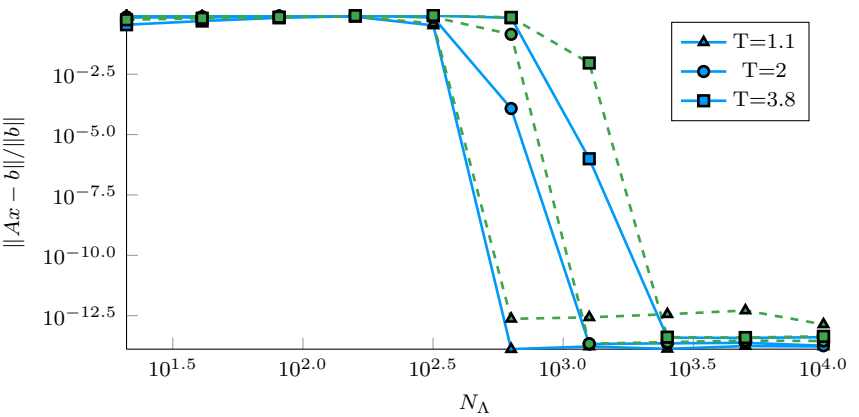


(a)

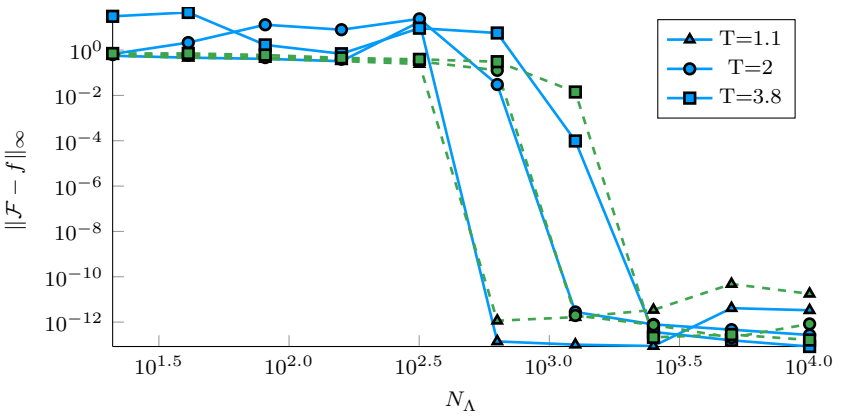


(b)

Figure 3.5: The residual norm of the system, and $\mathcal{L}_\infty$ norm of the error, computed by oversampling the solution by a factor 10, for test function $f_1(x) = x^2$.

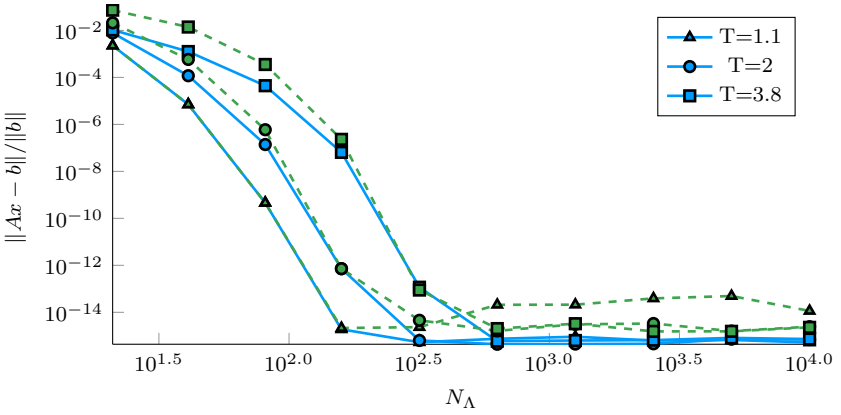


(a)

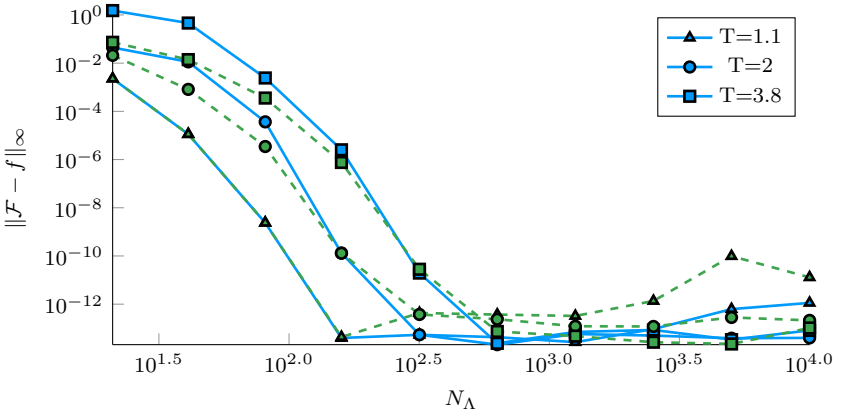


(b)

Figure 3.6: The residual norm of the system, and $\mathcal{L}_\infty$ norm of the error, computed by oversampling the solution by a factor 10, for test function $f_2(x) = \mathrm{Ai}(67x)$.

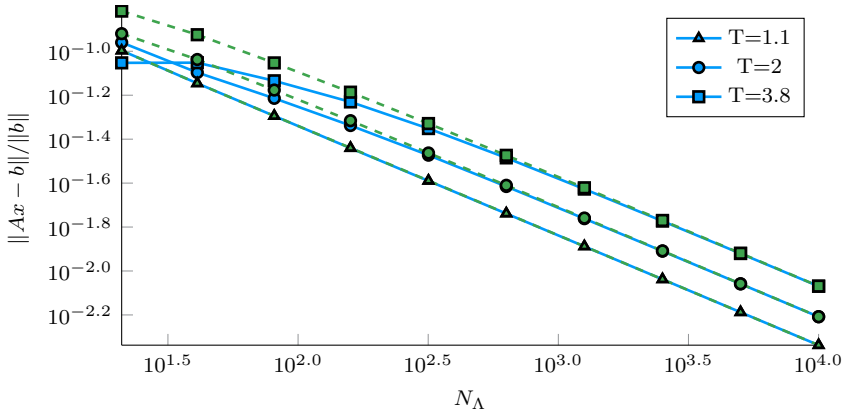


(a)

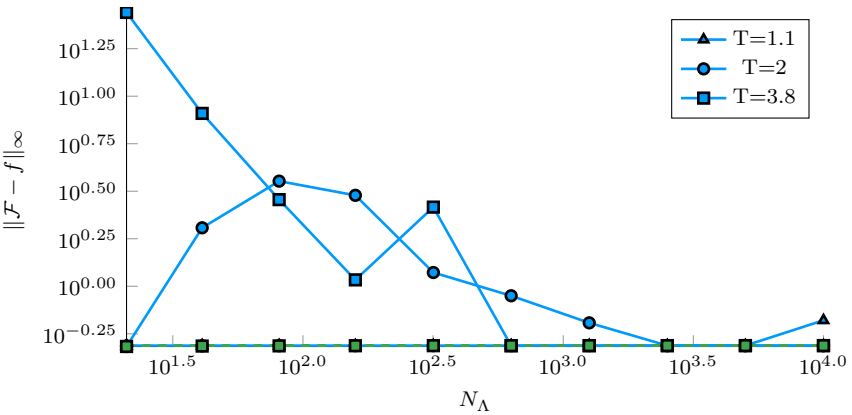


(b)

Figure 3.7: The residual norm of the system, and $\mathcal{L}_\infty$ norm of the error, computed by oversampling the solution by a factor 10, for test function $f_3(x) = \frac{1}{1+25x^2}$.

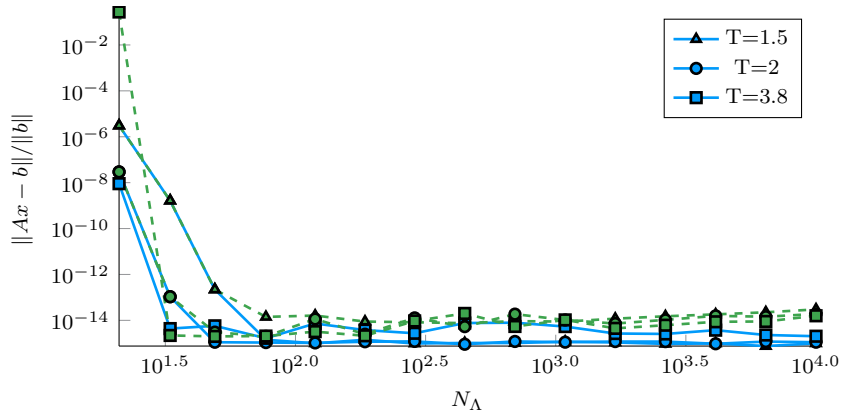


(a)

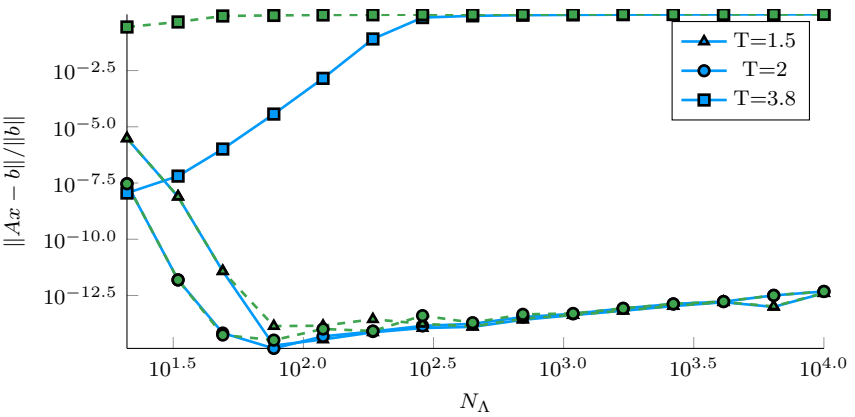


(b)

Figure 3.8: The residual norm of the system, and $\mathcal{L}_\infty$ norm of the error, computed by oversampling the solution by a factor 10, for the Heaviside test function f_4.



(a) $f(x) = \sin 10x$



(b) $f(x) = \sin N_\Lambda x/2$

Figure 3.9: Illustration of the robustness of FE approximations for large N_Λ.

the error is

$$\|A(x_\alpha + x_\beta) - b\| = \|P(Ax_\beta - b)\|.$$

The error in Algorithm 3 is then the same as the error made in solving the first system. Using the T-SVD of PA with cutoff τ, $x_\beta = (PA)^\dagger_\tau Pb$, the error is

$$\|P(Ax_\beta - b)\| = \sum_{\mu_i^3 - \mu_i < \tau} |\mu_i^2 - 1||u_i^* b|.$$

Now $\mu_i - \mu_i^3 < \tau$ implies either $\mu_i > 1 - \tau/2 + \mathcal{O}(\tau^2)$ or $\mu_i < \tau + \mathcal{O}(\tau^3)$. Then since we assumed $\|b_\gamma\| = \sum_{\mu_i<\tau} |u_i^* b| \leq \tau\|b\|$ under (3.4), we have for small τ

$$\begin{aligned}\|P(Ax_\beta - b)\| &= \sum_{\mu_i<\tau} |\mu_i^2 - 1||u_i^* b| + \sum_{\mu_i > 1-\tau/2} |\mu_i^2 - 1||u_i^* b| \\ &\leq 2\tau\|b\| + \mathcal{O}(\tau^2). \end{aligned} \tag{3.12}$$

However, this is for exact arithmetic, and there are several possible sources of error. There could be noise in the function samples, present on the right hand side b. This could either be signal noise, or a sign that the scheme has not yet converged. There is also the error in solving a system involving PA, using a randomised singular value decomposition. And there is the possibility of round-off error.

3.6.1 Approximate SVD

We first take a closer look at the randomised SVD from §3.1.1. An error bound for this approximation was first derived in [52] and proved sharp in [107].

Theorem 3.5. *[107, Theorem 1.4] Let A be a matrix with singular values $\mu_1 \geq \mu_2 \geq \ldots$. Let W be a $n \times l$ sampled test matrix with independent, mean-zero, unit-variance Gaussian entries and $l = k + p$ for integers k and p. If*

$$\tilde{U}\tilde{\Sigma}\tilde{V}^* = AW$$

then

$$\mathbb{E}\|A - \tilde{U}\tilde{U}^* A\| \leq \left(1 + \sqrt{\frac{k}{p-1}}\right)\mu_{k+1} + \mathrm{e}\frac{\sqrt{k+p}}{p}\left(\sum_{i>k}\mu_k\right)^{1/2}. \tag{3.13}$$

Here the norm $\|A\|$ is the matrix 2-norm, given by the largest singular value. $\mathbb{E}\|\cdot\|$ is the expected value of this 2-norm. This theorem relates the expected

error of in approximating the range of A to the quality of our rank estimate k, i. e. the singular values smaller than μ_k. Furthermore, following [52, Corollary 10.9], the probability that $||A - \tilde{U}\tilde{U}^*A||$ does not satisfy a slightly scaled (3.13) is at most $3e^{-p}$. This motivates our choices of k and p from §3.4. If the singular values μ_i decay sufficiently fast, and p is proportional to k, the error bound in Theorem 3.5 can be expressed as a constant times μ_{k+1}.

In the following we denote by $\tilde{U}\tilde{\Sigma}\tilde{V}^*$ the calculated svd. Denote by δA the additive error made in this computation, either through round-off or because of Theorem 3.5:

$$\tilde{A} = A + \delta A, \qquad \tilde{U}\tilde{\Sigma}\tilde{V} = \tilde{A}.$$

Then the following theorems due to Weyl and Mirsky relate the error E to the perturbation of the singular values σ.

Theorem 3.6. *[104] Let $\mu_1 \geq \mu_2 \geq \dots$ denote the singular values of A, and $\tilde{\mu}_1 \geq \tilde{\mu}_2 \geq \dots$ the singular values of $\tilde{A} = A + \delta A$. Then for any δA*

$$|\tilde{\mu}_i - \mu_i| \leq \|\delta A\|_2.$$

Theorem 3.7. *[76] Within the setting of Theorem 3.6,*

$$\sqrt{\sum_i (\tilde{\mu}_i - \mu_i)^2} \leq \|\delta A\|_F.$$

In this last theorem the norm $\|\cdot\|_F$ is the Frobenius norm

$$\|A\|_F = \sqrt{\sum_{i,j} |a_{ij}|^2}.$$

Theorems 3.6 and 3.7 show that finding the singular values of a matrix is a well-conditioned problem, since the singular values never deviate more from their exact value than by the magnitude of the error. However, this does not mean that every singular value can be accurately computed. For small μ_i, the relative error might be high. Note that for some matrix classes algorithms exist that allow (more) accurate SVDs, see e. g. [29].

With $PA = U\Sigma V^*$ the exact, and $\tilde{U}\tilde{\Sigma}\tilde{V}^*$ the calculated SVD, denote by δ_k the error for the k-th singular value:

$$\forall k : \tilde{\sigma}_k = \mu_k^3 - \mu_k + \delta_k. \tag{3.14}$$

We now want to quantify the impact of replacing $V\Sigma_{\tilde{\tau}}^{\dagger}U^*$ by $\tilde{V}\tilde{\Sigma}_{\tilde{\tau}}^{\dagger}\tilde{U}^*$ in Algorithm 3. If we assume the singular vectors are computed exactly, we

have

$$\tilde{x}_\beta = V\tilde{\Sigma}^\dagger U^* P b = \sum_{\tilde{\sigma}_k > \tilde{\tau}} v_k \frac{\mu_k^2 - 1}{\tilde{\sigma}_k} u_k^* b$$

and

$$\tilde{x}_\alpha = A^*(b - A\tilde{x}_\beta) = \sum_{\tilde{\sigma}_k > \tilde{\tau}} v_k \left(\mu_k - \frac{\mu_k^4 - \mu_k^2}{\tilde{\sigma}_k}\right) u_k^* b.$$

With $\tilde{x} = \tilde{x}_\alpha + \tilde{x}_\beta$, this leads to

$$\begin{aligned}\|P(A\tilde{x} - b)\| &= \sum_{\tilde{\sigma}_k > \tilde{\tau}} \left|\frac{\delta_k}{\tilde{\sigma_k}}\right| \left|\mu_k^2 - 1\right| |u_k^* b_k| + \sum_{\tilde{\sigma}_k < \tilde{\tau}} |\mu_k^2 - 1||u_k^* b|, \\ &= \sum_{\tilde{\sigma}_k > \tilde{\tau}} \left|\frac{\sigma_k \delta_k}{\sigma_k + \delta_k}\right| + \mathcal{O}(\tilde{\tau}), \end{aligned} \tag{3.15}$$

where the last line follows from (3.12), and our assumption $|u_k^* b| < C\mu_k$ as before. Now note that

$$\max_{\tilde{\sigma}_k > \tilde{\tau}} \left|\frac{\sigma_k \delta_k}{\sigma_k + \delta_k}\right| \approx \delta_k \left(1 + \frac{\delta_k}{\tilde{\tau}}\right),$$

obtained when $\tilde{\sigma}_k \approx \tilde{\tau} - \delta_k$ with δ_k negative. In other words, the worst case is a singular value that is perturbed downwards to right above the cutoff τ. This shows the importance of choosing $\tilde{\tau}$: it should be at the expected noise level of the singular values, or larger. Otherwise the noise level gets amplified by a factor $\delta_k/\tilde{\tau}$.

This does not cover the complete algorithm: for that we would need to characterise the singular vectors $\tilde{U}$ and $\tilde{V}$ as well, more specifically how well $V_\beta^* \tilde{V}_\beta$ and $U_\beta^* U_\beta$ approximate the identity. We only mention a possible approach based on the most common perturbation result for singular vectors by Wedin (see e. g. [94]). The angle between subspaces $\tilde{U}_1$ and U_2 is bounded by the inverse of the minimum separation between any diagonal element of $\tilde{\Sigma}_1$ and Σ_2. In practice, this means singular vectors can only be accurately retrieved if the corresponding singular values are well-separated. For $\tilde{\sigma}_k$ large the singular vector will mostly be precise, with relative accuracy diminishing with $\tilde{\sigma}_k$, and generally being on the order $\varepsilon_{\text{mach}}/\tilde{\sigma}_k$. The main difficulty in the analysis is that sometimes $\mu_k^3 - \mu_k \approx \mu_j^3 - \mu_j$, for μ_k close to 1 and μ_j close to zero. This complicates the analysis considerably.

3.6.2 Noise in the right hand side

The T-SVD solution is known to be robust to perturbations of b up to a noise level σ_0, under a few conditions.

Theorem 3.8. *[55, Characterisation 2] Let x_τ be the Truncated Singular Value Decomposition solution of a system $Ax = b$. Let $b = b^{exact} + e$, where e has zero mean and covariance matrix $\sigma_0^2 I$, and b satisfies the discrete Picard condition. Furthermore, let δ_0 be the part of b that is outside the range of A, $\delta_0 = \|(I - UU^T)b\|$. Then there exists a C so that for $\tau < C$*

$$\|Ax_\tau - b\| \approx \sqrt{\sigma_0^2(m) + \delta_0^2}$$

A more detailed formulation can be found in [54].

If no a priori noise information is available, the best choice of cutoff paramter is not obvious. A common approach in discrete ill-posed problems is utilizing the so-called L-curve. That is, plotting the solution norm $\|x\|$ versus the residual $\|Ax - b\|$. When decreasing the cutoff, the residual decreases and the solution norm increases. This curve, with both axes on a logarithmic scale, often has a distinctive corner, and the sharpness of the corner depends on the decay rate of the singular values [56]. For Fourier extensions the singular values are known to decay rapidly, leading to a sharp corner.

In Fig. 3.10 a discrete FE approximation is computed of $f(x) = \cos(20x - 1.3)$ with 500 degrees of freedom, and a cutoff ranging from $\tilde{\tau} = 10^{-15}$ to $\tilde{\tau} = 10^{-1}$, where $\tilde{\tau}$ is the cutoff for the T-SVD of PAW in Algorithm 3. The matrix W is chosen large enough for the minimal singular value of PAW to be below $\tilde{\tau}$. The results are shown for f perturbed at different (white) noise levels σ_0 as in Theorem 3.8. There is indeed an optimal cutoff parameter, where adding more singular values to the solution only approximates the noise. Since the noise does not satisfy the Discrete Picard condition, this increases the solution norm but not the accuracy.

By varying the cutoff parameter and detecting this corner, the algorithm can be made robust to noise.

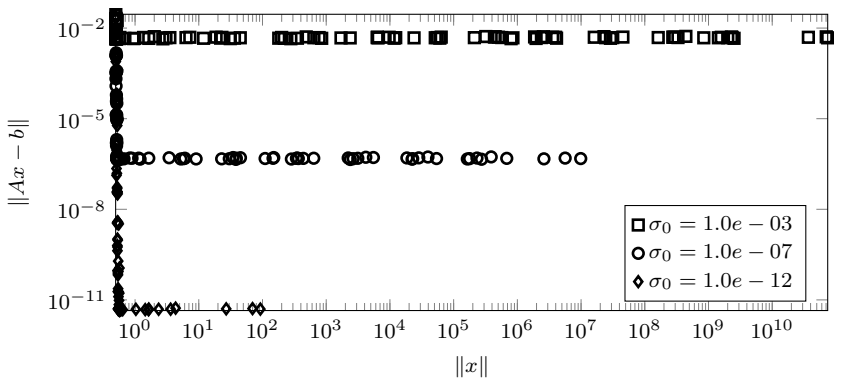


Figure 3.10: Residual versus solution norm for different noise levels. The data points correspond to different values of the cutoff parameter $\tilde{\tau}$. The curve has a distinct L shape, with the optimal solution found at the corner.

Chapter 4

Higher dimensional problems

This chapter details the extension of the algorithm described in §3.4 to the higher dimensional setting. As before, we focus on the oversampled collocation problem, using a Fourier extension frame.

The aim of the chapter is twofold. After defining equivalents to the P-DPSS from Chapter 2, we prove an extension of Property 2.15 under mild conditions for a two-dimensional domain Ω. This extension relates the growth of the plunge region to a measure of the boundary of Ω. The second part of this chapter details numerical experiments on the algorithm efficacy in two dimensions, in terms of complexity, accuracy and robustness. Most of the contents of this chapter are from [75].

4.1 Generalizing discrete Prolate Spheroidal wave sequences

We recall the FE problem matrix in a multi-dimensional setting from Chapter 2. The collocation matrix A is $N_\Omega \times N_\Lambda$, with an implicit ordering of the points $\boldsymbol{x}_k \in P_\Omega$, and the frequencies $\mathbf{l}_k \in P_\Lambda$ are in an $n_\Lambda \times \cdots \times n_\Lambda$ grid.

As before, the matrix A can be seen as the composition of three operations in time and frequency: extending Λ to $\hat{R}$ by zeros in the frequency domain, applying a discrete Fourier transform and restricting the result to Ω in the time domain. We shall develop this notion more formally.

The operators operate on sequences of length N_R on an $n_R \times n_R \times \dots$ grid. For indexing purposes we convert

$$P_R = \left\{ \left(\frac{k_1}{n_R}, \dots, \frac{k_D}{n_R} \right) \middle| \forall i : 0 \le k_i < n_R \right\}, \qquad P_\Omega = P_R \cap \Omega.$$

to the integer sets

$$I_R = n_R P_R \qquad \text{and} \qquad I_\Omega = n_R P_\Omega.$$

In accordance with §2.2, we denote by T_Ω the discrete space-limiting operator that sets all values outside Ω to zero,

$$(T_\Omega)_{\mathbf{k},\mathbf{l}} = \begin{cases} 1, & \mathbf{k} = \mathbf{l} \in I_\Omega, \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

Similarly, the discrete operator B_Λ is an $N_{\hat{R}} \times N_{\hat{R}}$ bandlimiting operator that eliminates all frequency content outside Λ. With F the D-dimensional Fourier transform, $B_\Lambda = F T_\Lambda F^*$.

With these definitions, the matrix AA' is the nonzero subblock of the operator $T_\Omega B_\Lambda T_\Omega$. Similar to the univariate case, the entries of $T_\Omega B_\Lambda T_\Omega$ are given in terms of a convolution kernel

$$(T_\Omega B_\Lambda T_\Omega)_{\mathbf{k},\mathbf{l}} = B(\mathbf{k} - \mathbf{l}), \qquad \forall \mathbf{k}, \mathbf{l} \in I_\Omega$$

where in the multivariate case B is a product of univariate Dirichlet kernels,

$$B(\mathbf{k}) = \prod_{d=1}^{D} b(k_d) \tag{4.2}$$

$$b(k) = \frac{\sin(\pi n_\Lambda k / n_R)}{n_R \sin(\pi k / n_R)}.$$

Here, $\mathbf{k} = (k_1, k_2, \dots)$ is a multidimensional point, and D is the number of dimensions.

Using these definitions for $B_\Lambda T_\Omega B_\Lambda$ and $T_\Omega B_\Lambda T_\Omega$, Definition 2.12 can be extended to generalised, multidimensional P-DPSS. As before, the eigenvectors of the Hermitian matrix $B_\Lambda T_\Omega B_\Lambda$ are denoted by φ_i, and $\breve{\varphi}_i = T_\Omega \varphi_i$ are eigenvectors of $T_\Omega B_\Lambda T_\Omega$. The corresponding eigenvalues of both matrices are the same and denoted by μ_i.

Similar to Properties 2.13 to 2.19, the following properties can be shown for the generalised P-DPSS:

Property 4.1. *The eigenvalues are bounded above by* 1 *and below by* 0.

Property 4.2. *Define the discrete inner products* $\langle \varphi_i, \varphi_j \rangle_R = \varphi_i \cdot \varphi_j$ *and* $\langle \varphi_i, \varphi_j \rangle_\Omega = (T_\Omega \varphi_i) \cdot (T_\Omega \varphi_j)$. *The* φ_i *are doubly orthogonal with respect to these inner products:*

$$\langle \varphi_i, \varphi_j \rangle_R = \delta_{ij}, \qquad \langle \varphi_i, \varphi_j \rangle_\Omega = \mu_i \delta_{ij}.$$

Property 4.3. *The* $\varphi_i(\Omega, \Lambda)$ *are eigenvectors of the* D*-dimensional DFT, with* Ω *and* Λ *interchanged.*

$$T_\Lambda F \varphi_i(\Omega, \Lambda) = \breve{\varphi}_i(\Lambda, \Omega),$$

where F *is the* D*-dimensional DFT matrix.*

Property 4.4. *Consider the discrete (semi-)norms corresponding to Property 4.2* $\|\cdot\|_R$ *and* $\|\cdot\|_\Omega$. *Then among all multidimensional sequences of size* N_R *with frequency support in* P_Λ, φ_1 *is the one most concentrated in* P_Ω *with concentration* $\langle \varphi_1, \varphi_1 \rangle_\Omega / \langle \varphi_1, \varphi_1 \rangle_R = \mu_1$. *Similarly, among the sequences of equal frequency support orthogonal to* φ_1, φ_2 *is the most concentrated in* Ω.

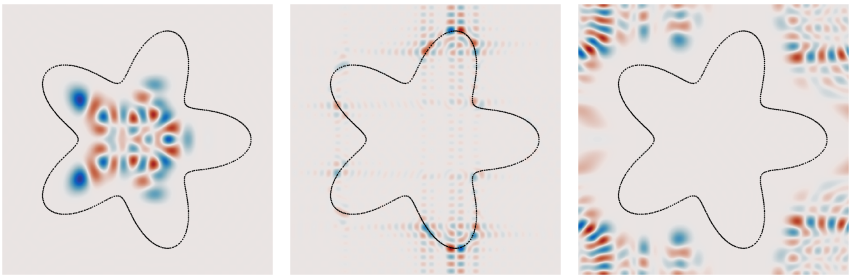
Figure 4.1 shows the Fourier series corresponding to these generalised P-DPSS, to be compared to Fig. 2.3. Property 4.4 is clearly illustrated: The maximal ratio $\mu_1 = \|\varphi_1\|_\Omega / \|\varphi_1\|_R$ with $\mu_1 \approx 1$ means that φ_1 is almost entirely supported on Ω – this in spite of being compactly supported in the (discrete) frequency domain. They are, after all, a finite Fourier series. Such a function is shown in the left panel of Fig. 4.1. In contrast, the functions corresponding to small eigenvalues are almost entirely supported on the exterior domain $R - \Omega$, as shown in the right panel of the figure. Finally, the middle functions with eigenvalues in the plunge region are supported everywhere. This is illustrated in the middle panel. In particular, these functions are the only ones that are non-neglible in a neighbourhood of the boundary. This is a clear indication that the plunge region is a phenomenon that relates to the boundary of the domain at hand.

The solution to $Ax = b$ using a truncated Singular Value Decomposition can be expressed in terms of these generalised discrete Prolate Spheroidal sequences,

$$x = \sum_{\mu_i > \tau} \frac{1}{\sqrt{\mu_i}} \breve{\varphi}_i(\Lambda, \Omega) \langle f, \breve{\varphi}_i(\Omega, \Lambda) \rangle_\Omega,$$

where the inner products are those from Property 4.2. This means Algorithm 3 is applicable to the multi-dimensional problem as well. The different stages of the algorithm are illustrated in Fig. 4.2, for function samples b, to be compared with Fig. 3.3.

The vector $x_\beta \approx V_\beta \Sigma_\beta^{-1} U_\beta' b$ found after the first step is based on the middle singular values, which correspond to functions that are supported along the



(a) $\mu_{60} \sim 1 - 10^{-10}$ (b) $\mu_{557} \sim 0.54$ (c) $\mu_{1300} \sim 10^{-10}$

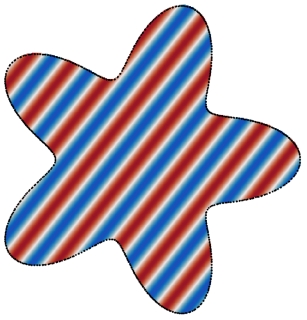
Figure 4.1: Fourier series $\mathcal{T}_N \varphi_i$ corresponding to φ_i for different values of the eigenvalue μ_i.

boundary of the domain. The Fourier series with x_β as its coefficients is shown in Fig. 4.2b: it approximates the data well in a neighbourhood of the boundary. Subtracting this approximation from the original function yields a function that vanishes smoothly towards the boundary of Ω. Hence, this function can be extended by zero and approximated efficiently with a Fourier transform, and that is expressed by the step $x_\alpha = A'(b - Ax_\beta)$. The vector x_α is a linear combination of the generalised P-DPSS that are concentrated in the interior of the domain. The numerical null space of A corresponds to the generalised P-DPSS concentrated in the exterior of the domain. Any such prolate can be added to our solution but it will only affect the extension, not the approximation on Ω itself, unless it is multiplied with a very large coefficient.

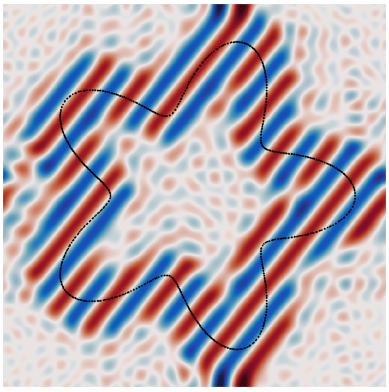
4.2 Singular value profile for generalised discrete Prolate Spheroidal Sequences

Algorithm 3 is very general. In fact, it will provide an approximate T-SVD solution for any linear system $Ax = b$. However, it only provides a speedup if the matrix $PA = (AA' - I)A$ has low rank, or equivalently, if the plunge region $\eta(\tau, N_\Lambda)$ of A grows slower than N_Λ. In the higher dimensional case, the cost of applying the matrix A is still $\mathcal{O}(N_\Lambda \log N_\Lambda)$. As we will show in this section, the cost of Algorithm 3 is now dominated by the SVD of the matrix PAW, with a cost $\mathcal{O}(N_\Lambda \eta(\tau, N_\Lambda)^2)$, that is lower than the full SVD cost $\mathcal{O}(N_\Lambda^3)$.

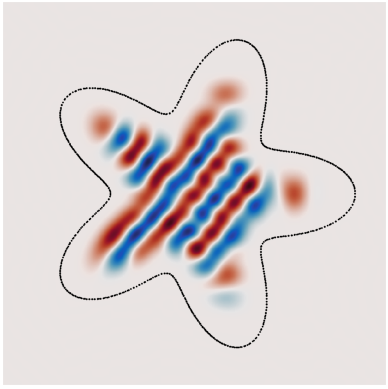
As in [65, 106], a bound on $\eta(\tau, N_\Lambda)$ can be inferred from trace iterates of the operator $T_\Omega B_\Lambda T_\Omega$. After bounding the difference between $tr(T_\Omega B_\Lambda T_\Omega)$ and



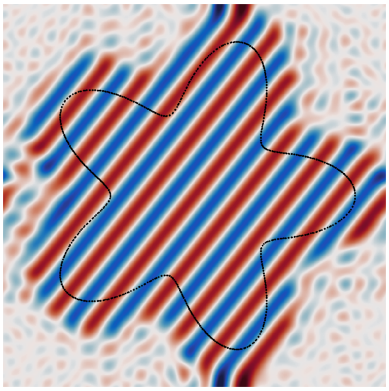
(a) b



(b) $\mathcal{T}_N x_\beta$



(c) $\mathcal{T}_N x_\alpha$



(d) $\mathcal{T}_N(x_\alpha + x_\beta)$

Figure 4.2: Steps in algorithm 3: Data is given on Ω (Fig. 4.2a), approximated using the eigenvalues $1 - \epsilon > \mu_i > \epsilon$, and yields a good approximation on the boundary (Fig. 4.2b). This solution subtracted from the data (Fig. 4.2c) is easily approximated by a regular Fourier series on the bounding box (Fig. 4.2d).

$tr((T_\Omega B_\Lambda T_\Omega)^2)$, this bound is shown to be of the same order as $\eta(\tau, N_\Lambda)$. We formulate our final result in Theorem 4.17.

The bound hinges on two observations:

- The contribution of a single point in P_Ω to $tr(T_\Omega B_\Lambda T_\Omega) - tr((T_\Omega B_\Lambda T_\Omega)^2)$ is inversely proportional to the distance between that point and the domain boundary.
- The number of points at a certain distance from the boundary is bounded by the number of boundary points and some terms depending only on the discrete domain topology.

The next section contains a concise illustrated proof of the second observation in the two-dimensional case. The first observation is proven in §4.2.2. Due to the discrete nature of the problem, we use some concepts borrowed from digital topology, see [21] for an overview.

4.2.1 Distance away from the boundary for general 2D domains

For reasons that will become clear later on, the metric of choice is the ℓ_∞ distance,

$$d(\mathbf{k}, \mathbf{l}) = \max_{i=1,\dots,D} |k_i - l_i|.$$

A point $\mathbf{k}$ on a regular grid in two dimensions can have up to 8 neighbors at a distance 1. We also define for each point set P a distance to its complement.

Definition 4.5. *The distance of a point* $\mathbf{k}$ *to the complement of a set* P *is given by*

$$d(\mathbf{k}; P) = \min_{\mathbf{l} \notin P} \|\mathbf{l} - \mathbf{k}\|_\infty.$$

Evidently it is true that $\forall \mathbf{k} \notin P : d(\mathbf{k}; P) = 0$, and

$$\forall \mathbf{k} \in P : d(\mathbf{k}; P) = 1 + \min_{\mathbf{l}: d(\mathbf{l}, \mathbf{k}) = 1} d(\mathbf{l}; P). \tag{4.3}$$

Next, let S_i denote the points in set S that are at a distance i away from the complement of S,

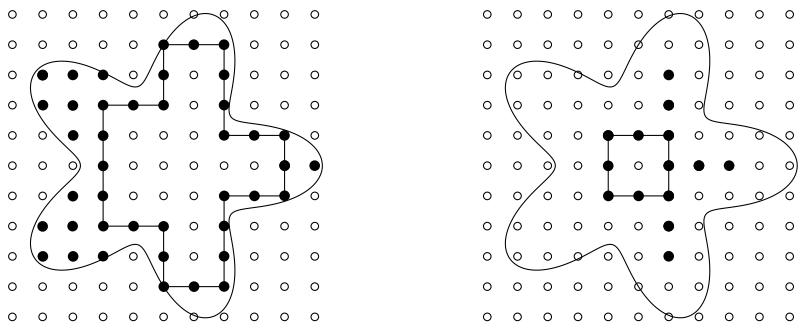
$$S_i = \{\mathbf{k} \in S : d(\mathbf{k}; S) = i\}. \tag{4.4}$$

The main result of this section is a bound on the size of these sets, in particular of $|S_{i+1}|$ in terms of $|S_i|$, which can be obtained using results from digital topology.

Let $\bar{S}_i$ denote the points in set S_i that have no neighbour in S_{i+1}

$$\bar{S}_i = \{\mathbf{k} \in S_i : \max_{\mathbf{l}:d(\mathbf{l},\mathbf{k})=1} d(\mathbf{l}; S) \leq i\}. \tag{4.5}$$

The sets S_i and $\bar{S}_i$ are illustrated in Fig. 4.3. For example, in the left panel the solid black dots not connected by a line belong to $\bar{S}_1$: their neighbours are either also in S_1 or in the exterior of the domain. The black dots connected by a line make up $S_1 \setminus \bar{S}_1$.



(a) S_1, $S_1 \setminus \bar{S}_1$(regular curve) (b) S_2, $S_2 \setminus \bar{S}_2$(regular curve)

Figure 4.3: An illustration of the sets S_1, $\bar{S}_1$ and their difference $S_1 \setminus \bar{S}_1$ in a component without holes (left panel), and similarly for S_2. It is clear that $|S_1| \geq |S_1 \setminus \bar{S}_1| > |S_2| \geq |S_2 \setminus \bar{S}_2|$. The set S_3 in this example consists of a single point.

Following the terminology of [21], we define a line cell as an adjacent pair of points $(\mathbf{k}, \mathbf{l}) : d(\mathbf{k}, \mathbf{l}) = 1$ and a surface cell as a set of four points where all pairwise distances are 1. We say that a pair of surface cells is point connected if they share a point, and line-connected if they share two points.

This allows us to state the definition of a regular digital manifold.

Definition 4.6. *[21, Definition 5.14] A point set S on a rectangular grid is a regular digital manifold if*

- *all points belong to a surface cell;*
- *for any pair of surface cells there is a line-connected path between them.*

We also define a slightly broader class of digital manifolds:

Definition 4.7. *A set S is a pseudoregular digital manifold if*

- *all points belong to a surface cell;*
- *for any pair of surface cells there is a point-connected path between them.*

We have the following lemma's.

Lemma 4.8. *For any finite 2-dimensional set S, $S - \bar{S}_1$ is a finite union of pseudoregular digital manifolds.*

Proof. If $S - \bar{S}_1$ is empty, then the result is true. Henceforth we assume it is not empty. To prove the first requirement of a pseudoregular manifold, note that because of (4.3) and (4.5), every $\mathbf{k} \in S_2$ is surrounded by points in $S - \bar{S}_1$ and is therefore part of 4 surface cells. Furthermore, because of (4.5) every point in $S_1 - \bar{S}_1$ has at least one neighbor in S_2, and is therefore part of a surface cell. Grouping surface cells by point-connectedness, the result is a union of pseudoregular manifolds. □

Lemma 4.9. *The distance to the boundary is preserved for all points after the removal of $\bar{S}_1$,*

$$\forall \mathbf{k} \in S \setminus \bar{S}_1 : d(\mathbf{k}; S - \bar{S}_1) = d(\mathbf{k}; S). \tag{4.6}$$

Proof. First note that the distance of a point is the minimum over all the 8 connected neighbours plus one. Therefore if S_i stays the same, S_{i+1} stays the same. Then note that all neighbors of points in S_2 are retained in $S - \bar{S}_1$. □

Theorem 4.10. *[21, Theorem 5.4] The boundary δS of a regular 2-dimensional manifold S is itself regular, and the union of closed regular curves.*

Lemma 4.11. *[21, Lemma 9.1] A closed 2-dimensional digital curve has 4 more convex corners than non-convex corners.*

Note that for now we assume the regular manifolds to be simply connected, i. e. the line connected paths from Definition 4.6 can be incrementally varied. This implies the boundary is a single closed regular curve.

The combination of these two theorems leads to

Lemma 4.12. *For a 2-dimensional pseudoregular manifold*

$$|S_{i+1}| \leq |S_i| - 8, \qquad i \geq 1. \tag{4.7}$$

Proof. Consider a regular manifold S. All points in S_1 form a closed digital curve with 4 more convex corners than non-convex corners. As illustrated in Fig. 4.4, a point on a straight segment maps to one element of S_2, a convex

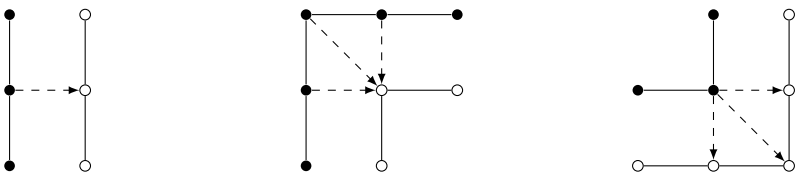
corner maps three points of S_1 to one of S_2, and a non-convex corner maps one points of S_1 to three of S_2. Since these points may coincide, the bound given by (4.7) is obtained.

For a pseudoregular manifold, note that a pair of point connected components can be regarded as 2 regular manifolds, with one point in common. Combining the bounds for both regular manifolds and subtracting the one point in common we again end up with (4.7).

This reasoning can be applied recursively by removing $\bar{S}_{i+1}$, obtaining another pseudoregular set.

□

For an illustration of this reasoning, see Fig. 4.4. Note here that this bound is



(a) straight (b) convex (c) non-convex

Figure 4.4: Illustration accompanying Lemma 4.12. Since there are four more convex corners than non-convex corners, and the target points can coincide, $|S_{i+1}| \leq |S_i| - 8$

sharp when S is a square set, i. e. each S_i has just four – convex – corners.

We conclude with a generalisation that allows for isolated connected components with a finite number of holes. The set S_{i+1} may be larger in this case than S_i, but the small growth does not invalidate the asymptotic complexity in the next section.

Lemma 4.13. *For a 2-dimensional set containing c pseudo-regular manifolds with h holes, the number of points a distance i away from the boundary is bounded by*

$$|S_{i+1}| \leq |S_i| - 8(c - h), \qquad i \geq 1. \tag{4.8}$$

Proof. For c pseudoregular manifolds, Lemma 4.12 holds individually for each S_{ij}. Thus the bound for the combined sets S_i is

$$|S_{i+1}| \leq |S_i| - 8c. \tag{4.9}$$

A hole in this context is a connected component not in P but entirely surrounded by it. Denote by S_{iB} the points whose closest neighbor not in P is in the hole. Then a similar reasoning to Lemma 4.12 shows that

$$|S_{i,B}| \leq |S_i \setminus \bar{S}_{i,B}| < |S_{i+1,B}| + 8. \tag{4.10}$$

Summing the bounds completes the proof. □

Remark 4.14. Theorem 4.12 does not hold in three dimensions and higher. In fact, the set S_{i+1} can be larger than S_i even for domains without a hole. A domain with an intrusion can have interior non-convex angles, at which a single point in S_i maps to many points in S_{i+1}.

4.2.2 Bounding $\eta(\tau, N_\Lambda)$

Theorem 4.15. *Let T_Ω and B_Λ be as in* (4.1) *and* (4.2)*. We are interested in the behavior for large N_Λ, with constant oversampling $\varrho = N_\Omega/N_\Omega$. Furthermore, let $N_{\delta\Omega}(n_\Lambda)$ denote the number of points in $P_{\delta\Omega}$ neighbouring the boundary, i. e. S_1 from the previous section:*

$$P_{\delta\Omega} = \{\mathbf{k} \in \Omega \quad | \quad \exists \mathbf{l}, \|\mathbf{l}\|_\infty = 1 : \mathbf{k} + \mathbf{l} \notin \Omega\}. \tag{4.11}$$

We further assume that the limit

$$\lim_{N_\Lambda \to \infty} (h(P_\Omega) - c(P_\Omega)) = C$$

exists with $C < \infty$, where $h(P_\Omega)$ and $c(P_\Omega)$ are as before the number of holes and distinct connected components of P_Ω. Then for the operator $T_\Omega B_\Lambda T_\Omega$

$$\lim_{N_\Lambda \to \infty} \operatorname{tr}(T_\Omega B_\Lambda T_\Omega) - \operatorname{tr}((T_\Omega B_\Lambda T_\Omega)^2) = \mathcal{O}\left(N_{\delta\Omega} \log N_\Lambda\right). \tag{4.12}$$

Proof. The trace of $T_\Omega B_\Lambda T_\Omega$ is, using (4.2),

$$\operatorname{tr}(T_\Omega B_\Lambda T_\Omega) = \sum_{\mathbf{k} \in I_\Omega} B(\mathbf{k} - \mathbf{k}) \tag{4.13}$$

$$= N_\Omega B(\mathbf{0}) = \frac{N_\Omega N_\Lambda}{N_R}. \tag{4.14}$$

For the squared operator trace, note that

$$\operatorname{tr}((T_\Omega B_\Lambda T_\Omega)^2) = \|T_\Omega B_\Lambda T_\Omega\|_F = \sum_{\mathbf{k} \in I_\Omega} \sum_{\mathbf{l} \in I_\Omega} |(T_\Omega B_\Lambda T_\Omega)_{\mathbf{k},\mathbf{l}}|^2.$$

Now, define an intermediate function

$$f(\mathbf{k}) = \sum_{\mathbf{l}\in I_\Omega} |(T_\Omega B_\Lambda T_\Omega)_{\mathbf{k},\mathbf{l}}|^2, \qquad \operatorname{tr}((T_\Omega B_\Lambda T_\Omega)^2) = \sum_{\mathbf{k}\in I_\Omega} f(\mathbf{k}).$$

This f can be rewritten as

$$\begin{aligned} f(\mathbf{k}) &= \sum_{\mathbf{k}\in I_\Omega} |B(\mathbf{k}-\mathbf{l})|^2 \\ &= \sum_{\mathbf{l}\in I_R} |B(\mathbf{k}-\mathbf{l})|^2 - \sum_{\mathbf{l}\in (I_R\setminus I_\Omega)} |B(\mathbf{k}-\mathbf{l})|^2 \end{aligned}$$

The first sum is equal to N_Λ/N_R through Parseval's equation. The second term is the sum over the index set $I_R \setminus I_\Omega$. As a shorthand notation, use

$$q_{\mathbf{k}} = d(\mathbf{k}; I_\Omega).$$

The largest inscribed square around $\mathbf{k}$ is then given by $\mathbf{k} + Q_{\mathbf{k}} \times Q_{\mathbf{k}}$, see Fig. 4.5. with $Q_{\mathbf{k}} = \{-q_{\mathbf{k}}+1, \ldots, q_{\mathbf{k}}-1\}$. Restricting I_Ω to this square and using that due to periodicity $\sum_{\mathbf{l}\in I_R} |B(\mathbf{l})|^2 = \sum_{\mathbf{l}\in I_R-\mathbf{k}} |B(\mathbf{l})|^2$, the last sum can be bounded by

$$\begin{aligned} \sum_{\mathbf{l}\in (I_R\setminus I_\Omega)} |B(\mathbf{k}-\mathbf{l})|^2 &< \sum_{\mathbf{l}\in (I_R\setminus (\mathbf{k}+Q_{\mathbf{k}}\times Q_{\mathbf{k}})} |B(\mathbf{k}-\mathbf{l})|^2 \\ &= \sum_{\mathbf{l}\in (I_R\setminus (Q_{\mathbf{k}}\times Q_{\mathbf{k}})} |B(\mathbf{l})|^2 \\ &= \left(\sum_{k\in R_d\setminus Q_{\mathbf{k}}} |B(k)|_d^2 \right)^2 + 2 \left(\sum_{k\in R_d\setminus Q_{\mathbf{k}}} |B(k)|^2 \sum_{k\in Q_{\mathbf{k}}} |B(k)|^2 \right). \end{aligned} \tag{4.15}$$

Here $R_d = \{0, \ldots, n_R - 1\}$, and $B(k)$ is the one-dimensional kernel.

From [106], the first sum can be bounded by

$$\sum_{k\in R_d\setminus Q_{\mathbf{k}}} |B(k)|_d^2 = \sum_{k=q_{\mathbf{k}}}^{n_R-q_{\mathbf{k}}} \left(\frac{\sin(\pi k/\varrho)}{n_R \sin(\pi k/n_R)} \right)^2 < \frac{1}{4q_{\mathbf{k}}} + \frac{\varrho}{16 q_{\mathbf{k}}^2}.$$

Further, $\sum_{k\in Q_{\mathbf{k}}} |B(k)|_d^2 < \varrho$. Then (4.15) can be bounded by a rational polynomial in $q_{\mathbf{k}}$.

$$\sum_{\mathbf{l}\in (I_R\setminus I_\Omega)} |B(\mathbf{k}-\mathbf{l})|^2 < \frac{\varrho}{2} q_{\mathbf{k}}^{-1} + \left(\frac{\varrho^2}{2^3} + \frac{1}{2^4} \right) q_{\mathbf{k}}^{-2} + \frac{\varrho}{2^5} q_{\mathbf{k}}^{-3} + \frac{\varrho^2}{2^8} q_{\mathbf{k}}^{-4},$$

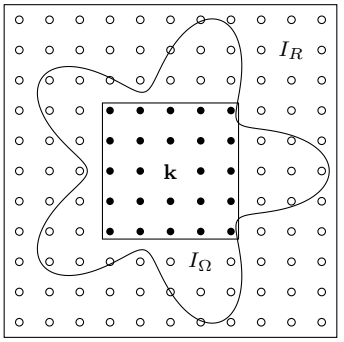


Figure 4.5: The largest inscribed square in I_Ω around any point $\mathbf{k}$ is $\mathbf{k}+Q_\mathbf{k}\times Q_\mathbf{k}$. In this figure $q_\mathbf{k}=3$, leading to a 5×5 square.

with all coefficients independent of N_R. Then

$$\begin{aligned}\operatorname{tr}((TBT)^2) &= \sum_{\mathbf{k}\in I_\Omega} f(\mathbf{k}) \\ &> \frac{N_\Omega N_\Lambda}{N_R} - \sum_{\mathbf{k}\in I_\Omega} \left(\varrho q_\mathbf{k}^{-1} + O(q_\mathbf{k}^{-2})\right).\end{aligned}$$

Now recall from §4.2.1 that I_Ω can be divided into sets $S_i = \{\mathbf{k} : q_\mathbf{k} = i\}$. Lemma 4.13 states that $|S_{i+1}| < |S_i| - 8(c-h)$. Furthermore, the size of the bounding box dictates that $q_\mathbf{k}$ can never exceed $\frac{n_R}{2}$. With this in mind it is easier to sum over the regions S_i than over all points at once. This leads to a bound

$$\frac{N_\Omega N_\Lambda}{N_R} - \operatorname{tr}((TBT)^2) < \sum_{i=1}^{n_R/2} \sum_{S_i} \left(\varrho i^{-1} + O(i^{-2})\right) \tag{4.16}$$

$$< \sum_{i=1}^{n_R/2} \left(N_{\delta\Omega} + 8i(h-c)\right)\left(\varrho i^{-1} + O(i^{-2})\right) \tag{4.17}$$

$$< C_1 N_{\delta\Omega} \log n_R + C_2 N_{\delta\Omega} \tag{4.18}$$

(4.14) and (4.18) combined give the desired result. □

Next, we want to relate the difference between iterated traces to the plunge region. This relies on a fairly general counting argument, as in [106, 67]. Recall that the trace of a matrix equals the sum of its eigenvalues, and the trace of a matrix squared equals the sum of the squares of its eigenvalues.

Lemma 4.16. *Let $1 > \lambda_1(N) > \lambda_2(N) > \cdots > \lambda_N(N) > 0$ be a given ordered series where*

$$\sum_{i=1}^{N} \lambda_i(N) = CN \tag{4.19}$$

$$\sum_{i=1}^{N} \lambda_i(N)^2 = CN - g(N) \tag{4.20}$$

where $g(N) = o(N)$ is a positive function. Then $|\{\lambda_k : \epsilon < \lambda_k < 1 - \epsilon\}| = O(g(N))$.

Proof. Define $k_{\min}$ and $k_{\max}$ as the limits of the intermediate region

$$k_{\min} = \arg\min_k \lambda_k : \lambda_k < 1 - \epsilon, \qquad k_{\max} = \arg\max_k \lambda_k : \lambda_k > \epsilon. \tag{4.21}$$

Then $\forall k > k_{\min} : \lambda_k^2 < (1 - \epsilon)\lambda_k$ and $\forall k \leq k_{\min} : \lambda_k^2 < (1 - \epsilon)\lambda_k + \epsilon$, so that

$$\sum_k \lambda_k^2 < (1 - \epsilon) \sum_k \lambda_k + \epsilon k_{\min}. \tag{4.22}$$

Substituting (4.19) and (4.20) leads to

$$k_{\min} > CN - \frac{g(N)}{\epsilon}. \tag{4.23}$$

Similarly, $\forall k < k_{\max} : \lambda_k^2 < (1 + \epsilon)\lambda_k - \epsilon$ and $\forall k \geq k_{\max} : \lambda_k^2 < (1 + \epsilon)\lambda_k$, so that

$$\sum_k \lambda_k^2 < (1 + \epsilon) \sum_k \lambda_k - \epsilon k_{\max}. \tag{4.24}$$

Combined with (4.19) and (4.20) this yields the upper bound

$$k_{\max} < CN + \frac{g(N)}{\epsilon} \tag{4.25}$$

□

Theorem 4.17. *Let $\Omega, T_\Omega, B_\Lambda$ and $P_{\delta\Omega}$ be as in Theorem 4.15. Then for the operator $T_{P_\Omega} B_\Lambda T_{P_\Omega}$*

$$\eta(\tau, N_\Lambda) = O\left(N_{\delta\Omega} \log n_R\right) \tag{4.26}$$

where $\eta(\tau, N_\Lambda)$ is as in (2.11).

Proof. The proof follows directly from Theorem 4.15 and Lemma 4.16, and noting that for square matrices $\operatorname{tr}(A^k) = \sum \lambda^k$. □

Remark 4.18. Theorem 4.17 gives a bound in terms of $N_{\delta\Omega}(n_\Lambda)$. For any 2-dimensional non-fractal domain, $N_{\delta\Omega} = \mathcal{O}(n_\Lambda)$. To see this, note that $d = \lim_{n_\Lambda \to \infty} \frac{\log N_{\delta\Omega}}{\log n_\Lambda}$ is equal to the box-counting or Minkowski-Bouligand definition of the boundary dimension [36]. For a non-fractal domain this is equal to the topological dimension of the boundary which is 1.

Remark 4.19. Theorem 4.17 can easily be seen to hold in the one-dimensional case whenever Ω is a finite union of k disjoint intervals. The plunge region would then grow as $\mathcal{O}(k \log N_\Lambda)$. This can be seen as a discrete version of [67], where time- and bandlimiting operators for such Ω where proven to have this singular value profile in the continuous case.

Remark 4.20. Theorem 4.17 leads to an $O(N_\Lambda^2 \log(N_\Lambda)^2)$ complexity for Algorithm 3 on 2D domains. It is however difficult to extend the results from §4.2.1 to higher dimensions, as there is no straightforward equivalent of Lemma 4.12. Even if it could be extended to those domains, Theorem 4.17 would still yield asymptotic reductions in higher dimensions, though the savings have diminishing returns, generally of the order $O(n_\Lambda^{3d-2})$ versus $O(n_\Lambda^{3d})$ for a full SVD.

4.3 Numerical results

This section contains examples and numerical results for various two-dimensional domains and method parameters. The aim is to demonstrate the asymptotic complexity, convergence properties and robustness of the algorithm. The different geometries analyzed in this section are shown in Fig. 4.6. All domains are normalised to have equal area. These domains where chosen for their contrasting properties:

- The square and diamond show the method is not rotation invariant.
- The square and disk show the effect of corners
- The disk and ring show the effect of a simply connected domain versus a not simply connected domain.
- A double asteroid is included to study boundaries that are not smooth.

The effect of the domain on complexity and accuracy is discussed in detail in §4.3.3.

Throughout these experiments, the basis of choice is a Fourier basis on the rectangle $[-T, T] \times [-T, T]$ with $N_\Lambda = n_\Lambda^2$ degrees of freedom. Unless specified

otherwise, the value for T is 2 and the oversampling factor N_Ω/N_Λ is taken to be 4. The cutoff τ is consistently 10^{-14}.

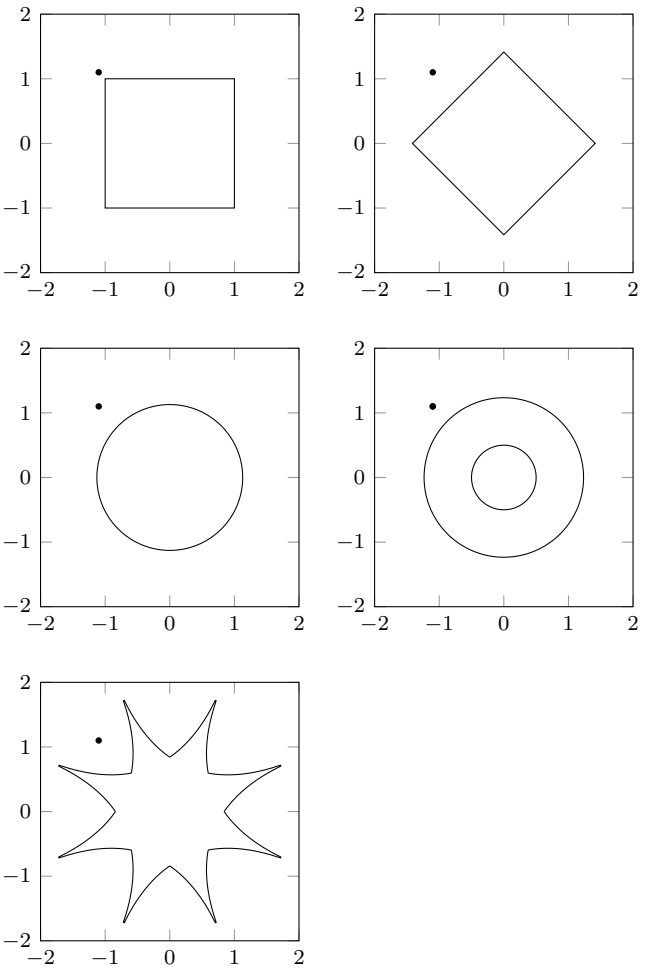


Figure 4.6: Test domains used throughout this section. The dot marks the location of the singularity in the second test function.

Remark 4.21. When the domain is rectangular, applying Algorithm 3 directly is less efficient than decoupling the problem and solving it through tensor product 1D Fourier extensions. In general there are always opportunities to exploit structure in the domain when it is present. However, to keep the comparisons uniform, we will not do so in this section.

4.3.1 Complexity

Figure 4.7 shows execution time for Fourier frame approximations with increasing degrees of freedom. The approximant is irrelevant here since complexity of Algorithm 3 is independent of the right hand side. Our algorithm computes the equivalent of a truncated SVD and is applied to the sampled function in a single step. The domain is also largely irrelevant since the plunge region size is similar for the chosen domains, see §4.3.3. Therefore the timings are shown for just the one example: approximating

$$f(x, y) = \mathrm{e}^{(x+y)} \cos(20xy)$$

on a disk of area 4.

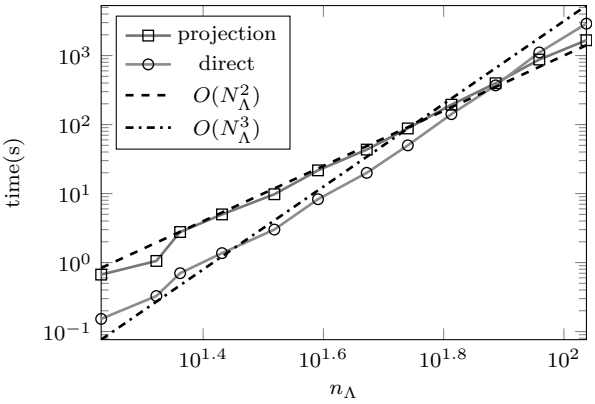


Figure 4.7: Execution time for a 2D frame approximation, using both a direct solver and the projection algorithm. $O(N_\Lambda^2)$ and $O(N_\Lambda^3)$ shown dashed in black.

The results confirm the $O(N_\Lambda^2 \log^2 N_\Lambda)$ complexity of the projection algorithm, with the dominant cost being the SVD used in the randomised SVD solver. Unfortunately, the direct method is only overtaken for $n_\Lambda > 90$, making the projection method mostly suited for problems requiring a large number of degrees of freedom, such as oscillatory functions. However, the algorithm complexity directly depends on the plunge region size, which heavily depends on τ. In the presence of noise on the order of δ, one may want to choose $\tau > \delta$, see §3.6.2, lowering the rank-estimates considerably.

4.3.2 Accuracy

To show convergence, the frame approximation method was applied to a set of test functions, for increasing degrees of freedom. The test functions are

- A well-behaved, smooth function

$$f(x, y) = \mathrm{e}^{x+y}.$$

- A function with a singularity inside the bounding box

$$f(x, y) = \frac{1}{((x - 1.1)^2 + (y - 1.1)^2)^2}.$$

- An oscillatory function

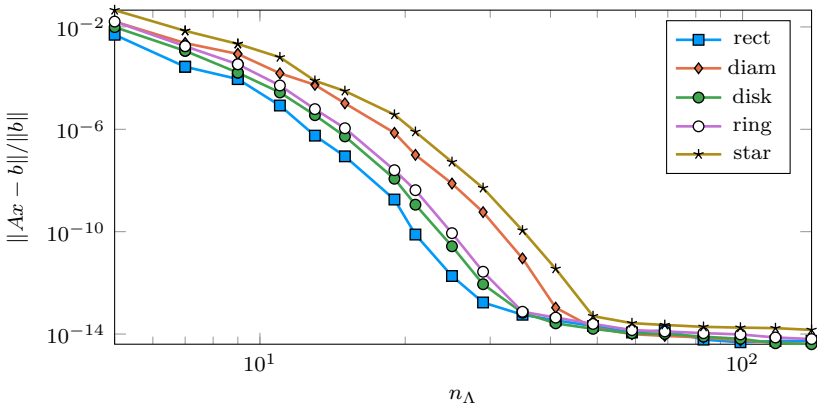
$$f(x, y) = \cos(24x - 32y)\sin(21x - 28y).$$

- A function with a discontinuity in the first order partial derivatives

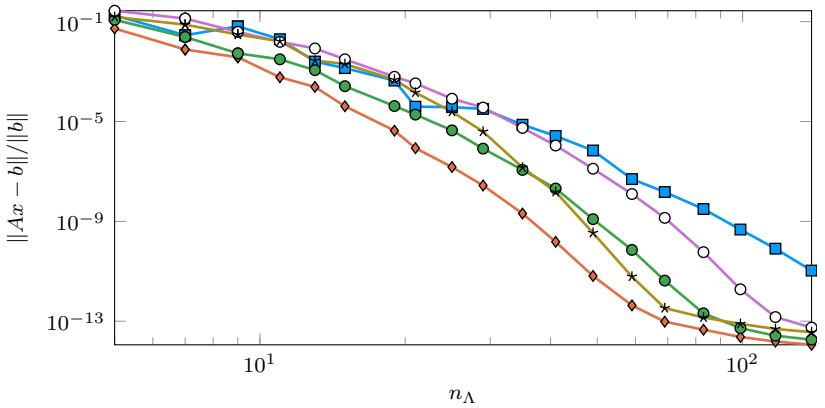
$$f(x, y) = |xy|.$$

The results are shown in Figs. 4.8 to 4.11, for the residual norm $\|Ax - b\|_2/\|b\|_2$ on the one hand and for the largest point error $\|\mathcal{F} - f\|_\infty$ on the other hand, sampled randomly in the domain (10000 samples). There are a few interesting observations to be made regarding the convergence behavior for different target functions.

- The approximation error for the smooth function shows superalgebraic convergence on all domains, strengthening claims in this regard [1, 3]. The only exception is the star-shaped point error. We return to this in §4.3.3.
- The approximation error for the oscillatory function behaves exactly as expected, decreasing rapidly once the highest oscillatory mode can be resolved by the basis functions.
- Figure 4.10b shows the results for a function with a singularity right outside the domain of interest. Similar to the 1D case, this results in a slower, yet still superalgebraic rate of convergence.
- A function that has s continuous derivatives will exhibit order $s + 1$ convergence, as seen in Fig. 4.11b, for the residual error. The largest point error shows very little, if any, convergence.

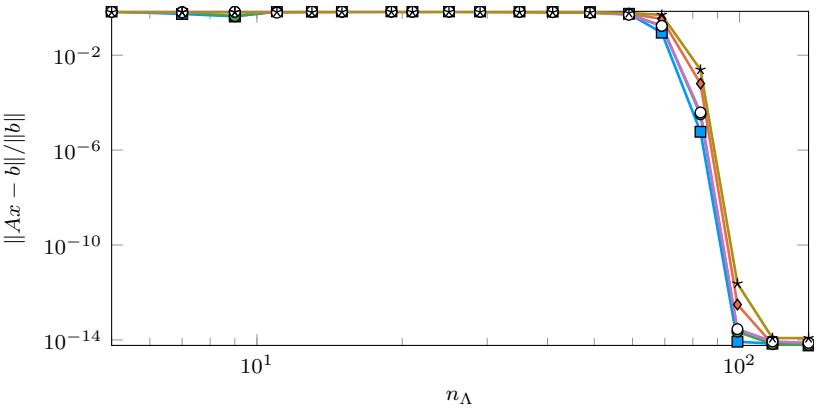


(a) $f(x,y) = \mathrm{e}^{x+y}$

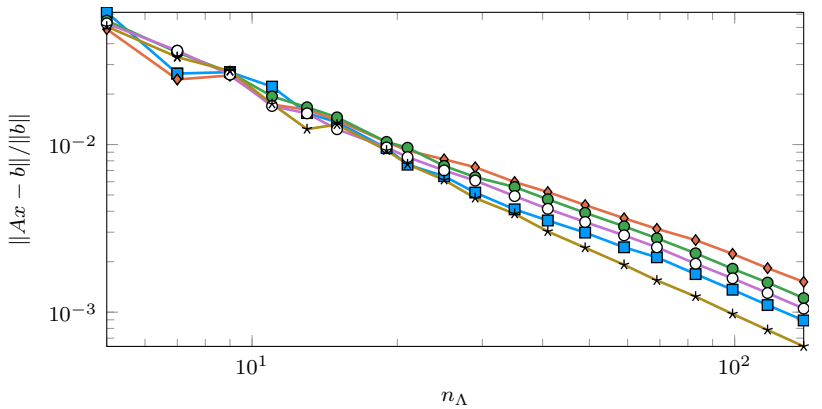


(b) $f(x,y) = \frac{1}{((x-1.1)^2+(y-1.1)^2)^2}$

Figure 4.8: Residuals for a 2D frame approximation, for different domains and approximants.

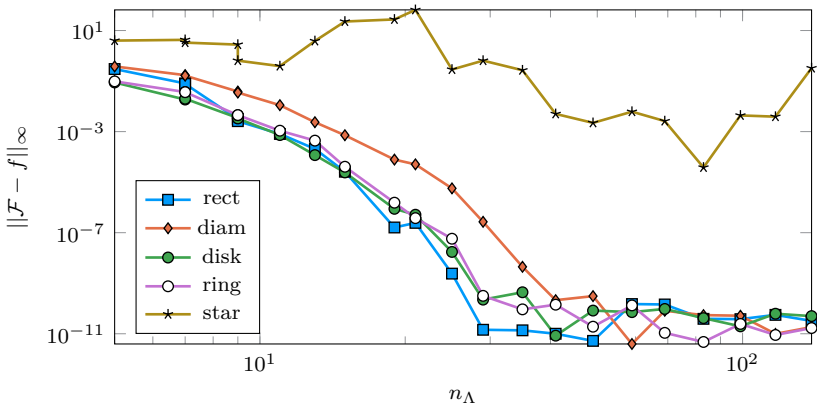


(a) $f(x,y) = \cos(24x - 32y)\sin(21x - 28y)$

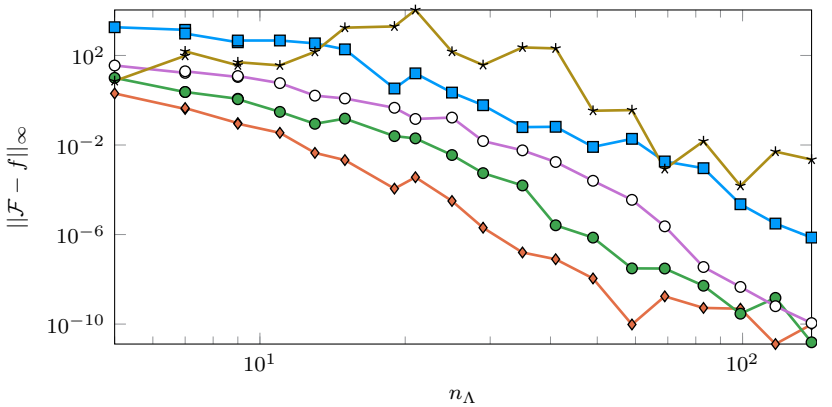


(b) $f(x,y) = |xy|$

Figure 4.9: Residuals for the approximations from Fig. 4.11.

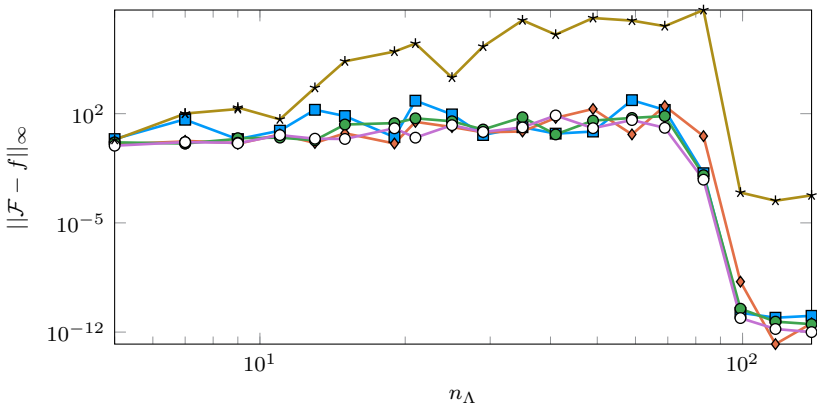


(a) $f(x, y) = \mathrm{e}^{x+y}$

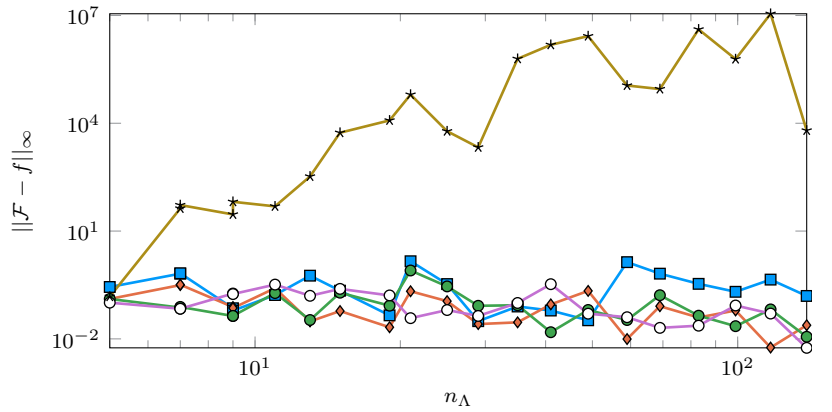


(b) $f(x, y) = \frac{1}{((x-1.1)^2+(y-1.1)^2)^2}$

Figure 4.10: Maximum pointwise error for a 2D frame approximation, for different domains and approximants.



(a) $f(x, y) = \cos(24x - 32y)\sin(21x - 28y)$



(b) $f(x, y) = |xy|$

Figure 4.11: Maximum pointwise error for a 2D frame approximation, for different domains and approximants.

4.3.3 Influence of domain shape

Plunge region estimates

Theorem 4.17 leads to an estimate of the plunge region of the form

$$\eta(\tau, N_\Lambda) = C_1 N_{\delta\Omega} \log n_R + O(N_{\delta\Omega}) \tag{4.27}$$

where $C_1 = \left(\frac{n_\Lambda}{n_R}\right)^{D-1} \frac{1}{\tau^2}$. This is because the eigenvalues λ_i of $T_\Omega B_\Lambda T_\Omega$ are the squares of the singular values σ_i of the collocation matrix A, so that

$$\tau < \sigma_i < 1 - \tau \Leftrightarrow \tau^2 < \lambda_i < 1 - 2\tau + \tau^2.$$

The constant C_1 is a gross overestimate, as shown in Fig. 4.12, which plots the ratio $\eta(\tau, N_\Lambda)/(N_{\delta\Omega} \log n_R)$ as a function of N_Λ. The ellipse, square

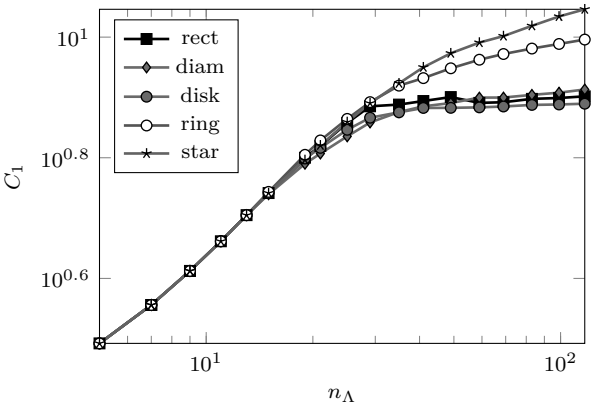


Figure 4.12: Estimate of plunge region size with respect to $N_{\delta\Omega} \log n_R$.

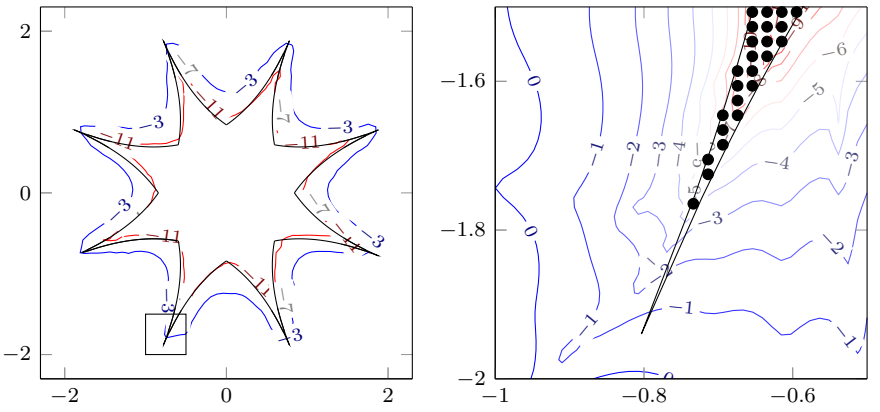
and diamond seem to reach the asymptotic behavior (4.27) with a constant $C_1 \sim 10$. The ring and star domain have not yet reached their plateau, but it is reasonable to assume this plateau, like the bound from Theorem 4.17, is proportional in some way to the domain boundary length. Using the Euclidean length, the plateau for the ring would be at $10^{1.08}$ and for the star at $10^{1.39}$, both plausible from Fig. 4.12.

Remark 4.22. An alternative to using an estimate for the plunge region rank is to use an adaptive form of the random matrix algorithm for unknown ranks. The idea is to iteratively increase the number of random vectors until the smallest singular value of PAW is below the chosen threshold τ. As per [68], this eliminates the need for a difficult estimation of C_1, at a maximum factor 2 increase in cost.

Influence on convergence

The influence of domain shape on convergence is readily apparent from Figs. 4.10a to 4.11b. There are a number of factors that combined lead to the differences seen between the domains.

The maximum pointwise error In Fig. 4.10 the error was taken as the infinity norm over Ω for $\mathcal{F} - f$, calculated over 10000 random samples of Ω. However, the actual approximation $\mathcal{F} - f$ in all these experiments was computed from an equispaced grid of collocation points. Some points in Ω, e.g. at the spikes of the star shape may be far away from the equispaced grid, see Fig. 4.13. In this figure the maximum pointwise error is found at the tip of the spikes, far removed from the actual sample points. Since no information about the tips was taken into account, convergence in these areas cannot be expected until they are sufficiently covered by the grid. This problem is unique to the higher-dimensional case, as in the one-dimensional problem the endpoints can be guaranteed to be included.



(a) Error Contour ($\log_{10} \|\mathcal{F} - f\|$) (b) Detail of sharp feature ($\log_{10} \|\mathcal{F} - f\|$)

Figure 4.13: Error contour for the star-shaped domain with test function $f(x, y) = \mathrm{e}^{x+y}$ and $n_\Lambda = 50$. Right figure shows detail of a sharp feature together with the location of actual sample points.

Figure 4.8 contains the experiments from Fig. 4.10 but now shows the relative error in the grid points only. The results show that all approximations do converge as expected in the collocation points. The difference is, as was expected most apparent in the star-shaped region. In §5.1.3 we discuss a

possible approach to improve accuracy precisely at the tips of the star-shaped region.

Proximity to the singularity In the 1D case, the effect of the presence of a singularity on the convergence rate was covered in §2.3.2. In particular, the convergence rate is $\|\mathcal{P}^{\tau}_{N_\Omega,N_\Omega} f - f\| \sim \rho^{-N_\Lambda}$ after a certain breakpoint, with

$$\rho = \min\{\rho^*, E(T)\}$$

and ρ^* determined by the largest singularity-free region $\Gamma(\rho^*)$ (see (2.28)). As illustrated in Fig. 3.7, we expect a singularity close to the domain to be the limiting factor in the convergence rate.

This is confirmed in Figs. 4.8b and 4.10b. With the singularity of the test function

$$f(x,y) = \frac{1}{((x-1.1)^2 + (y-1.1)^2)^2}$$

located at $(1.1, 1.1)$, the rectangle is closest to the singularity. In rough order of proximity to the singularity, the other domains are star, ring, circle and diamond. The effect is most apparent in Fig. 4.8b, where the convergence rate of the error for the diamond shape is significantly faster than the rate for the rectangle, where for test functions without singularities they differ much less.

4.3.4 Robustness

To ensure the results remain stable for large N_Λ, Fig. 4.14 shows the approximation of a function

$$f(x,y) = \sin\left(\frac{n_\Lambda}{2}(x+y)\right)$$

for increasing degrees of freedom N_Λ on a unit circle. This showcases the close relationship between the number of degrees of freedom needed per wavelength and the size of the extension region. For $T = 1.2$ and $T = 2$, the extension region is narrow enough for the approximation to resolve the oscillation. The main difference here is the convergence rate, which is slower for smaller T as per the 1D case. For $T = 3$, the highest frequency mode present in the Fourier basis of degree N_Λ is not enough to resolve the function, and it is impossible for convergence to occur. This behavior is identical to that observed in Fig. 3.9.

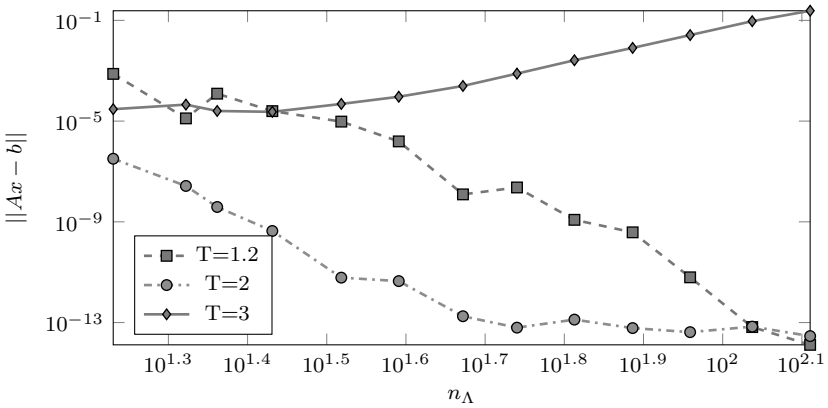


Figure 4.14: Accuracy for a 2D frame approximation for an increasingly oscillatory function, and different extension regions $R = [-T, T] \times [-T, T]$.

Chapter 5

Algorithm modifications

In Chapter 3 an implicit algorithm was introduced, that efficiently solves the least squares collocation problem, given the singular value profile from Fig. 2.2 and a fast matrix-vector product.

In Chapter 4 we showed that this singular value profile is present in the higher-dimensional case, and that the algorithm has applications in that setting as well. This chapter takes Algorithm 3 a step further. First we show that when the collocation matrix A is extended with some extra rows or columns, the singular value profile still holds. This makes Algorithm 3 applicable to a variety of problems that depend in some way on function approximation, illustrated in Sections 5.1.2 to 5.1.4. This includes the solution of elliptic boundary value problems with constant coefficient differential operators. Second, the redundancy in the solution is used to obtain a smoother solution in §5.2.

5.1 More general minimisation problems

The previous chapters have covered the case of finding

$$f_{N_\Lambda} = \arg\min_{g \in \Phi_{N_\Lambda}} \|g - f\|_2 \tag{5.1}$$

with the norm either the $\mathcal{L}^2$ norm on Ω, or a discrete norm on P_Ω, and f_{N_Λ} in a truncated Fourier frame Φ_{N_Λ}. A suitable f_{N_Λ} is obtained through the T-SVD, and the algorithms from Chapters 3 and 4 calculate the regularised projection

$\mathcal{P}^{\tau}_{N_\Omega,N_\Lambda} f$. In this chapter we extend (5.1) to minimizing

$$f_{N_\Lambda} = \underset{g\in\Phi_{N_\Lambda}}{\arg\min} \sum_{i=1}^{S} c_i \|L_i g - h_i\|_{\Omega_i}. \tag{5.2}$$

$$\Phi_{N_\Lambda} = \{\phi_i\}_{i=1}^{N_1} \cup \{\psi_i\}_{i=1}^{N_2} \cup \dots \tag{5.3}$$

where Φ_{N_Λ} can be a collection of frames and/or bases, and L_i operators.

Following the previous chapters, the minimiser is found through oversampled collocation. As an example, consider the Poisson equation on Ω, with homogenous Dirichlet boundary conditions:

$$\Delta f(\boldsymbol{x}) = h_1(\boldsymbol{x}), \qquad \boldsymbol{x} \in \Omega, \tag{5.4}$$

$$f(\boldsymbol{x}) = 0, \qquad \boldsymbol{x} \in \delta\Omega. \tag{5.5}$$

A possible approach to solving this problem is based on collocation on both domain and boundary, as proposed by Kansa in [62, 63] for Radial Basis Functions (see §1.3.2). Following [22], we formulate the problem as an (oversampled) weighted least squares problem

$$f_{N_\Lambda} = \underset{g\in\Phi_{N_\Lambda}}{\arg\min} \left(\sum_{\boldsymbol{x}\in P_\Omega} |\Delta g(\boldsymbol{x}) - h_1(\boldsymbol{x})|^2 + c^2 \sum_{\boldsymbol{x}\in P_{\delta\Omega}} |g(\boldsymbol{x})|^2 \right), \tag{5.6}$$

where $N_\Omega + N_{\delta\Omega} > N_\Lambda$, $N_\Omega = |P_\Omega|$, $N_{\delta\Omega} = |P_{\delta\Omega}|$. The coefficients x of f_{N_Λ} are found through solving the least squares system

$$\begin{bmatrix} A_\Omega \hat{L}_1 \\ c A_{\delta\Omega} \end{bmatrix} x = \begin{bmatrix} h_1 \\ \mathbf{0} \end{bmatrix}. \tag{5.7}$$

Here, $\hat{L}_1$ is a matrix that represents Δ acting on the Fourier coefficients, A_Ω and $A_{\delta\Omega}$ are the collocation matrices in P_Ω and $P_{\delta\Omega}$ respectively, and h_1 contains the right hand side of (5.4) sampled in P_Ω. When Φ_{N_Λ} consists of certain RBFs, this approach converges to the solution of (5.2) under certain conditions [22]. Among these are ellipctic L_1, smoothness of the exact solution, and Ω Lipschitz. In §5.1.4 we will show that such weighted least squares problem can be solved efficiently using Algorithm 3, although no convergence results are known.

For Algorithm 3 to be applicable, there are two requirements, that will be made more precise in §5.1.1. In particular:

- At least one of the operators L_i in (5.2) should be invertible in coefficient space, i.e. $\hat{L}_i^{-1}$ exists. Furthermore, when discretising as in (5.7), the

matrix subblock corresponding to L_i should be dominant in size. Looking at the example in (5.6) and (5.7), the matrix block $A_\Omega \hat{L}_1$ should be largest, or equivalently $N_\Omega \gg N_{\delta\Omega}$.

- Likewise, one of the function sets in (5.3) should have many elements compared to all others. This should be a frame or basis for which fast function approximation exists, like orthonormal bases or Fourier extension frames.

This section will cover three examples of these more general minisation problems: the aforementioned boundary value problem, a problem where the sampling set is no longer uniform, and a frame consisting of an orthogonal basis augmented by polynomials or singular functions. These last two examples illustrate the second limitation: the number of polynomials or singular functions added should be small compared to the size of the other basis or frame. Augmenting bases and frames resembles Eckhoff's method [34], where an approximation of the form

$$f(x) = f_{N_\Lambda}(x) + \sum_{i=1}^{T} \sum_{j=1}^{k} Q_{i,j} V_{i,j}(x) \tag{5.8}$$

is computed, where $f_{N_\Lambda}(x)$ is a Fourier series, and $V_{i,j}(x)$ are known features of f that are difficult to approximate with Fourier series. The key difference is that Eckhoff's method explicitly derives formulae for the $Q_{i,j}$, while in our approach the Fourier coefficients and the $Q_{i,j}$ are found simultaneously, through a coupled least squares system.

5.1.1 Singular values of the extended matrix

In §3.4 we formulated two conditions for Algorithm 3 to be significantly faster than $\mathcal{O}(N_\Lambda^3)$ operations when solving

$$Ax = b, \qquad A \in \mathbb{C}^{N_\Omega \times N_\Lambda}. \tag{5.9}$$

First, the collocation matrix A needs to be applicable fast, preferably in $\mathcal{O}(N_\Lambda \log N_\Lambda)$ operations. Second, the size of the plunge region

$$\eta(\tau, N_\Lambda) = \min_{1 \leq k,j \leq N_\Lambda} (k - j) \quad \text{s.t.} \quad \sigma_j \geq 1 - \tau, \quad \tau > \sigma_k \tag{5.10}$$

needs to grow as $o(N_\Lambda)$. The complexity of Algorithm 3 is then $\mathcal{O}(N_\Lambda \eta(\tau, N_\Lambda)^2)$ operations.

A requirement for our approach is that the collocated minimisation problem can always be written as a block matrix equation,

$$\tilde{A} = \begin{bmatrix} A_{11} & A_{12} & \dots \\ A_{21} & \ddots & \\ \vdots & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \end{bmatrix}, \tag{5.11}$$

where A_{11} has small $\eta(\tau, N_\Lambda)$. The other blocks should correspond to a relatively small number of rows n_r and columns n_c:

$$\tilde{A} \in \mathbb{C}^{(N_\Omega+n_r)\times(N_\Lambda+n_c)}, \qquad n_r + n_c \ll N_\Lambda.$$

The cost of applying $\tilde{A}$ is equal to the cost of applying A_{11}, with an additional $n_r N_\Lambda + n_c N_\Omega + n_r n_c$ multiplications. For the application to be $\mathcal{O}(N_\Lambda \log N_\Lambda)$ we therefore need $n_r + n_c = \mathcal{O}(\log N_\Lambda)$.

For the singular value profile of $\tilde{A}$, we show the relationship between $\eta(\tau, N_\Lambda)$ for A, and $n_r + n_c$ in Theorem 5.2. This follows from the well known Cauchy Interpolation theorem for eigenvalues of Hermitian matrices, and its equivalent for singular values of rectangular matrices.

Theorem 5.1. *[86] Let $N_\Omega, N_\Lambda, n_c, n_r$ be natural numbers. Given nonnegative real numbers $\tilde{\sigma}_1 \geq \tilde{\sigma}_2 \geq \cdots \geq \tilde{\sigma}_{\min\{N_\Lambda+n_c, N_\Omega+n_r\}}$, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min\{N_\Omega, N_\Lambda\}}$, there exists an $(N_\Lambda + n_c) \times (N_\Omega + n_r)$ complex matrix with $\tilde{\sigma}_i$ as singular values and having a $N_\Omega \times N_\Lambda$ submatrix with σ_i as singular values if and only if*

$$\tilde{\sigma}_i \geq \sigma_i, \qquad i = 1, 2, \dots, \min\{N_\Omega, N_\Lambda\}, \tag{5.12}$$

$$\sigma_i \geq \tilde{\sigma}_{i+n_c+n_r}, \qquad i = 1, 2, \dots, \min\{N_\Omega - n_c, N_\Lambda - n_r\}. \tag{5.13}$$

This leads immediately to

Theorem 5.2. *Let n_c and n_r be the number of columns and rows added to the matrix A to get $\tilde{A}$. Furthermore let $\eta(\epsilon, N_\Omega, A)$ and $\eta(\epsilon, N_\Omega, \tilde{A})$ be defined as in* (5.10) *for A and $\tilde{A}$ respectively. Then*

$$\eta(\epsilon, N_\Omega, \tilde{A}) \leq \eta(\epsilon, N_\Omega, A) + n_c + n_r. \tag{5.14}$$

Proof. Let $\tilde{\sigma}_i$ denote the singular values of $\tilde{A}$. Then from Theorem 5.1, with k and j as in (5.10):

$$\epsilon > \sigma_k \geq \tilde{\sigma}_{k+n_c+n_r},$$

and

$$\tilde{\sigma}_j \geq \sigma_j \geq 1 - \epsilon.$$

Then

$$\eta(\epsilon, N_\Omega, \tilde{A}) \leq k + n_c + n_r - j \tag{5.15}$$

$$\leq \eta(\epsilon, N_\Omega, A) + n_c + n_r. \tag{5.16}$$

□

Furthermore, from Theorem 5.1 we have

$$\forall k \geq n_c + n_r : \tilde{\sigma}_k \leq 1. \tag{5.17}$$

Together with Theorem 5.2 this leads to

$$|\{|\tilde{\sigma} - \tilde{\sigma}^3| > \varepsilon\}| \leq \eta(\varepsilon, N_\Omega, A) + 2(n_c + n_r), \tag{5.18}$$

a rank estimate for $(\tilde{A}\tilde{A}^* - I)\tilde{A}$ in Algorithm 3. This means the asymptotic complexity for Algorithm 3 is unchanged as long as

$$n_c + n_r = \mathcal{O}(\eta(\tau, N_\Lambda)). \tag{5.19}$$

5.1.2 Augmented bases and frames

Fourier basis plus polynomials

An example borrowed from [1] is the augmentation of the Fourier basis by a set of polynomials, in this example the normalised Legendre polynomials $L_k(x) = \sqrt{k + 1/2}P_k(x)$, both on the interval $[-1, 1]$. The truncated frame is

$$\Phi_{N_\Lambda,k} = \left\{ \frac{1}{\sqrt{2}} \mathrm{e}^{\mathrm{i}\pi n x} \right\}_{n \in I_N} \cup \{L_j(x)\}_{j=1,\ldots,k}. \tag{5.20}$$

We exclude $L_0(x) = 2^{-1/2}$ since it is already present in the Fourier basis functions. The least squares system becomes

$$\begin{bmatrix} A_F & A_L \end{bmatrix} \begin{bmatrix} x_F \\ x_L \end{bmatrix} = b, \qquad \Omega = [-1, 1] \tag{5.21}$$

where $A_F \in \mathbb{C}^{N_\Omega \times N_\Lambda}, A_L \in \mathbb{C}^{N_\Omega \times k}$ are the (oversampled) collocation matrices and $x_F \in \mathbb{C}^{N_\Lambda}, x_L \in \mathbb{C}^k$ are the coefficients for the Fourier and Legendre terms respectively.

We illustrate this approach by approximating

$$f(x) = \mathrm{e}^x + \cos 5(x - 0.1)^2 \tag{5.22}$$

for a range of values of N_Λ and k. Solving (5.21) immediately results in an expansion in the frame (5.20). Figures 5.1a and 5.1b show both timings and the approximate $\mathcal{L}_\infty$ error (calculated in a random 10000 point grid) for the approximation as a function of $N_\Lambda + k$.

Since f is not periodic on $[-1, 1]$, the regular Fourier Series, i. e. $\Phi_{N_\Lambda,0}$ does not converge in the $\mathcal{L}_\infty$ norm. On the other hand, since f is entire, the best Chebyshev approximation in Φ_{0,N_Λ} converges superexponentially. For the frame $\Phi_{N_\Lambda,k}$, [1] has an equivalent of Theorem 1.18 for this frame when calculating the continuous FE $\mathcal{P}^\tau_{N_\Lambda} f$.

Theorem 5.3. *[1, Proposition 5.9] Let $k \in \mathbb{N}$ be fixed and consider the frame (5.20). If $f \in H^k(-1, 1)$ for $0 \le k \le K$ then*

$$\|f - \mathcal{P}_{N_\Lambda} f\| \le C_k N_\Lambda^{-k} \|f\|_{H^k(-1,1)} \tag{5.23}$$

and

$$\|f - \mathcal{P}^\tau_{N_\Lambda} f\| \le C_{k,d} \left(N_\Lambda^{-k} + \sqrt{\tau}\right) \|f\|_{H^k(-1,1)}. \tag{5.24}$$

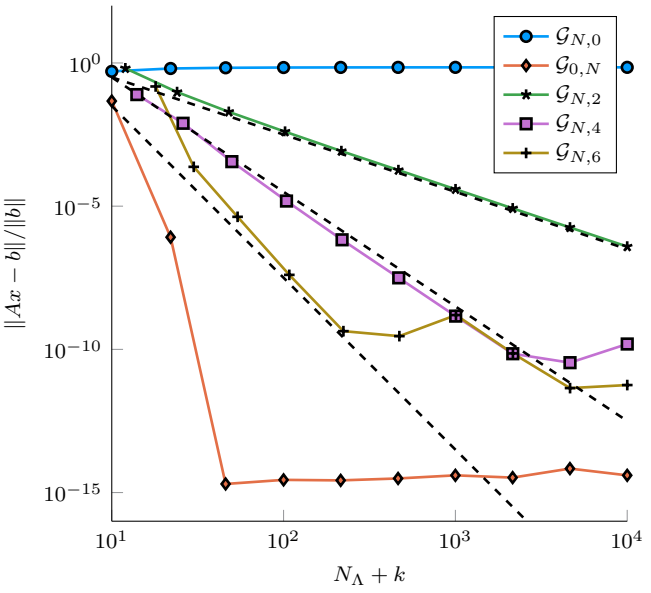
Intuitively, the nonperiodic polynomials allow some derivatives to be interpolated at the boundary, smoothing out the periodic extension so the Fourier series converges faster. The convergence rates in Fig. 5.1a for the discrete regularised projections agree with those for the continuous projections, as evidenced by the black lines showing $\mathcal{O}(N_\Lambda^{-k})$ complexity.

The complexity for moderate N_Λ is dominated by the cost of the SVD, which is $\mathcal{O}(N_\Lambda(20 + k)^2)$ operations. Figure 5.1b shows the linear cost in N_Λ. The cost difference seen between $\Phi_{N_\Lambda,0}$ and $\Phi_{N_\Lambda,2}$ is explained by the minimum of 20 random vectors, needed as a safety buffer for the randomised SVD algorithm, as in §3.1.1. For large N_Λ the FFTs dominate the cost, and the algorithm has an asymptotic complexity of $\mathcal{O}((20 + k)N_\Lambda \log N_\Lambda)$ operations. Note that the frame approximation in Φ_{0,N_Λ} was computed using collocation in Chebyshev points to achieve the $O(N_\Lambda \log N_\Lambda)$ complexity.

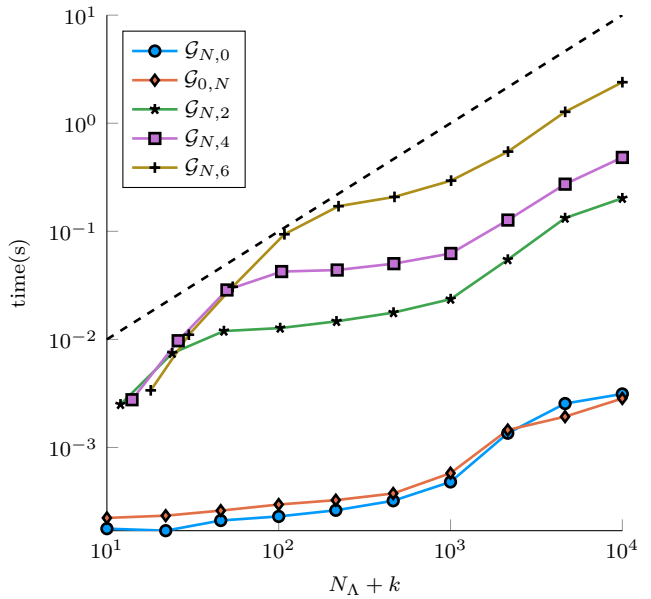
Fourier Frame plus Weighted polynomials

Taking the previous example further we approximate the two-dimensional function

$$f(\boldsymbol{x}) = \sin\left(7\sqrt{\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2}\right) + 0.2(\cos(21\boldsymbol{x}_1 - 22\boldsymbol{x}_2)^2 + \sin(23\boldsymbol{x}_1 + 24\boldsymbol{x}_2)) \tag{5.25}$$



(a) Convergence



(b) Timings

Figure 5.1: $\mathcal{L}_\infty$-error and time complexity when using Algorithm 3 to approximate (5.22) using a N_Λ term Fourier series augmented with k polynomials. In Fig. 5.1a the black dotted lines show $\mathcal{O}(N_\Lambda^{-k})$ convergence. In Fig. 5.1b the black dotted line shows $\mathcal{O}(N_\Lambda)$ complexity.

The first term has a square root type singularity at $\boldsymbol{x} = (0, 0)$. The second term in (5.25) is oscillatory. The domain Ω is a circle with radius 0.7 centered at the origin. We can tailor a frame on the bounding box $[-1, 1] \times [-1, 1]$ to this type of singularity:

$$\Phi_{N_\Lambda, k^2} = \left\{ \mathrm{e}^{in\pi \boldsymbol{x} \cdot \mathbf{l}/2} \right\}_{\mathbf{l}_1, \mathbf{l}_2 = -n_\Lambda, \ldots, n_\Lambda} \cup \{\omega(\boldsymbol{x}) T_i(\boldsymbol{x}_1) T_j(\boldsymbol{x}_2)\}_{i,j=0,\ldots,k-1}$$

$$\omega(\boldsymbol{x}) = \sqrt{\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2},$$

where the Fourier Frame is augmented by a weighted tensor product Chebyshev polynomial frame with k^2 degrees of freedom. The weight ω encodes the singular behavior in (5.25). The least squares system becomes

$$\begin{bmatrix} A_F & W A_T \end{bmatrix} \begin{bmatrix} x_F \\ x_T \end{bmatrix} = b, \qquad \Omega = [-1, 1] \tag{5.26}$$

where $A_F \in \mathbb{C}^{N_\Omega \times N_\Lambda}, A_T \in \mathbb{C}^{N_\Omega \times k^2}$ are, as before, the (oversampled) collocation matrices for the bases and the respective coefficients are $x_F \in \mathbb{C}^{N_\Lambda}, x_T \in \mathbb{C}^{k^2}$. The matrix W is a diagonal matrix consisting of ω evaluated in the sample points (recall that each row of A_F and A_T corresponds to a point in P_Ω).

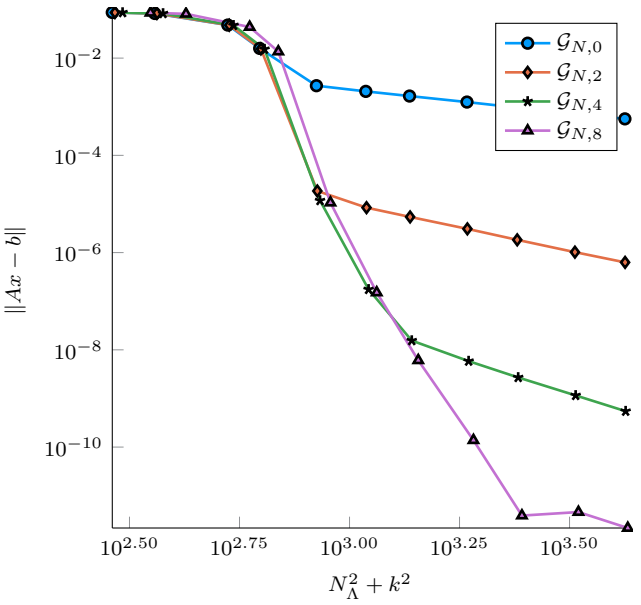
Figure 5.2 shows the result of applying Algorithm 3 to this problem for increasing $N_\Lambda + k$ and $k = 0, 2, 4, 8$. The regular Fourier frame $\Phi_{N,0}$ shows $O(N^{-1/2})$ convergence in the residual, as expected. The oscillatory term gets resolved by the Fourier frame at $N_\Lambda^2 \sim 10^3$. Afterward, convergence is algebraic, like in the example from the previous section.

5.1.3 Local accuracy improvement

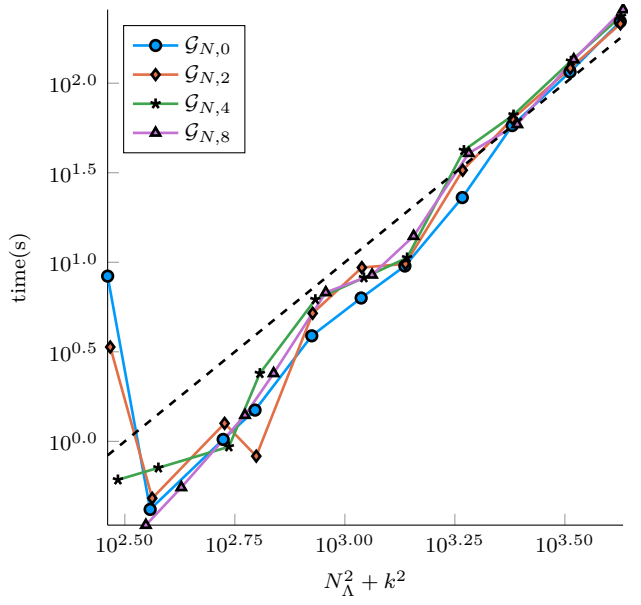
We now present an example that takes (5.11) in a different direction, by adding extra rows instead of columns. Augmenting the collocation grid P_Ω with a small point set P_χ of extra points results in extra matrix rows. Here, P_χ is a point set containing samples that are not on the regular grid, but still in Ω where the target function is known. Then the least squares system becomes

$$\begin{bmatrix} A_\Omega \\ A_\chi \end{bmatrix} x = \begin{bmatrix} b_\Omega \\ b_\chi \end{bmatrix}. \tag{5.27}$$

A possible application when approximating a function on an interval is to ensure that the endpoints are part of the collocation grid. However, an example where this can be used to greater effect is approximations on a domain Ω from



(a) Convergence



(b) Timings

Figure 5.2: Residual error and time complexity when using Algorithm 3 to approximate (5.25) using an N_Λ^2 term Fourier series augmented with k^2 weighted polynomials. Black dotted line in Fig. 5.2b shows $O(N_\Lambda^2)$ complexity.

§4.3. On the starlike domain shown in Fig. 4.13b, four different test functions where approximated and it was found that while the residual steadily decreases, the $\mathcal{L}_\infty$ error, measured in random samples, was significantly higher than for domains without sharp features. Experiments showed the pointwise error to be largest on the boundary, and especially in the tips of the star shaped domain. See Fig. 4.13a in the previous chapter for a contour plot of the pointwise error in one of the experiments.

A possible remedy is to manually add sample points P_χ to these undersampled regions. The effect is shown in Fig. 5.3 and it is very clear. Adding just 4 extra sample points to each of the tips lowers the $\mathcal{L}_\infty$ error – measured in 10^6 random points in Ω – from $\sim 10^{-1}$ to $\sim 10^{-4}$.

This is backed up by Fig. 5.4, where the $\mathcal{L}_\infty$ error with added gridpoints is consistently 100 to 1000 times smaller, up until the point the residual reaches machine precision. The timings confirm the negligible influence the added gridpoints have on the algorithm's performance.

5.1.4 Boundary value problems

In this section we return to the example from §5.1, where L_1 is a differential operator, and L_2 corresponds to the boundary conditions. This allows us to formulate differential equations in the form of (5.2). For a k-th order differential equation

$$L_1 f = \alpha_k \frac{d^k f}{dx^k} + \cdots + \alpha_1 \frac{df}{dx} + \alpha_0 f = h_1,$$

denote by $p_1(x) = \alpha_k x^k + \cdots + \alpha_1 x + \alpha_0$ its characteristic polynomial. Now since

$$f'(x) = \sum_{k=-\infty}^{\infty} \hat{f}[k] \left(\mathrm{e}^{\frac{\mathrm{i}k\pi x}{b-a}} \right)' = \sum_{k=-\infty}^{\infty} \frac{\mathrm{i}k\pi}{b-a} \hat{f}[k] \mathrm{e}^{\frac{\mathrm{i}k\pi x}{b-a}}$$

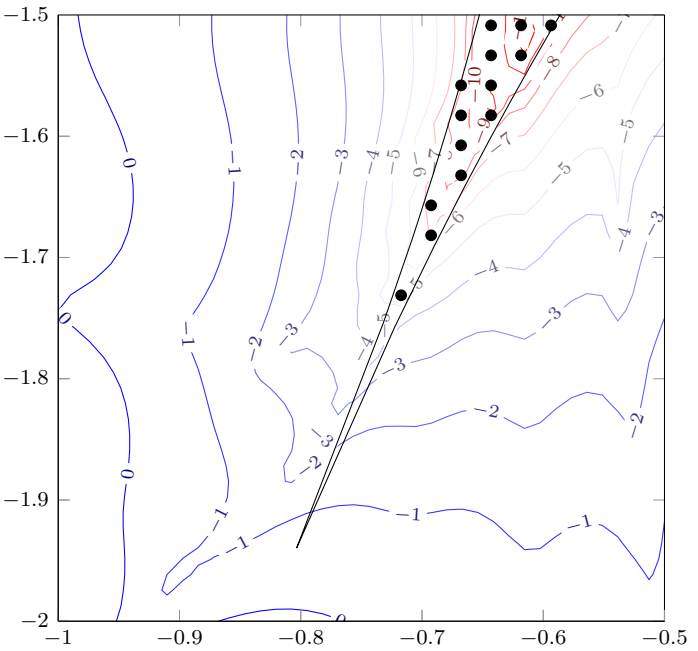
the differential operator on the Fourier coefficients of f is diagonal

$$\widehat{f'} = D\hat{f}, \qquad D_{k,k} = \frac{ik\pi}{b-a}, \quad k = -\infty, \ldots, \infty.$$

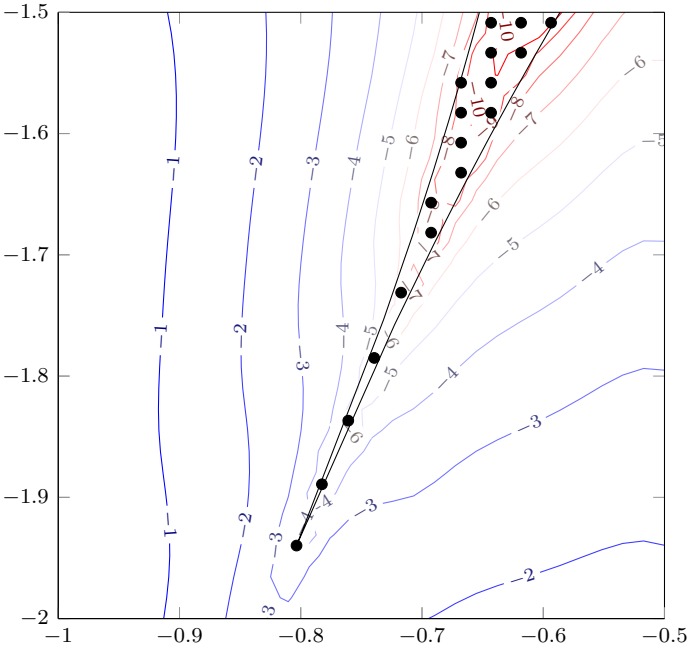
Denoting by $p(D)$ the characteristic polynomial as a matrix polynomial acting on D, the differential equation in coefficient space becomes

$$p_1(D)\hat{f} = \hat{h_1}.$$

In matrix form for a Fourier series with N_Λ degrees of freedom, this leads to a system $A_\Omega p(D)x = b$. A_Ω is as before the collocation matrix in P_Ω, and b is h_1

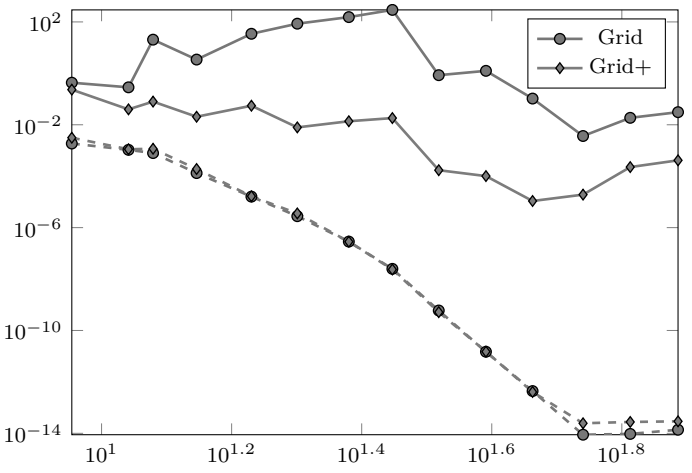


(a) Regular grid

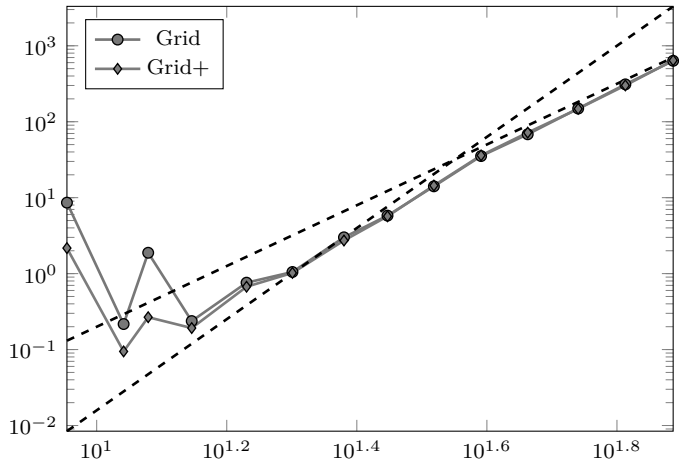


(b) Extra gridpoints

Figure 5.3: Contour plot of $\log_{10} \|\mathcal{F} - f\|_\infty$, when zoomed in on one of the star tips from Fig. 4.13b.



(a) Errors



(b) Timings

Figure 5.4: Errors and timings when approximating $f(x, y) = \mathrm{e}^{x+y}$ on the star-shaped domain from Fig. 4.13b, for increasing N_Λ. The full lines in Fig. 5.4a show the L_∞ error, the dotted lines the residual $\|\tilde{A}x - b\|/\|b\|$. Grid denotes the regular grid P_Ω, Grid+ denotes $P_\Omega \cup P_\chi$. The dotted lines in Fig. 5.4b show $O(N_\Lambda^2)$ and $O(N_\Lambda^3)$ complexity.

sampled in P_Ω. The boundary conditions can then be added as extra rows in the problem matrix. Let

$$\begin{aligned} L_2 f(x) &= h_2(x), && x \in \delta\Omega_2 \\ &\vdots \\ L_k f(x) &= h_k(x), && x \in \delta\Omega_k \end{aligned}$$

denote the boundary conditions with operator L_k, right hand side h_k and subset of the boundary $\delta\Omega_k$. Let $p_k(x)$ be the characteristic polynomial of L_k. Then with $P_{\delta\Omega_k}$ appropriate sampling sets on the boundary, and $A_{\delta\Omega_k}$ the corresponding collocation matrix, the differential equation can be collocated as

$$\begin{bmatrix} A_\Omega p_1(D) \\ c_2 A_{\delta\Omega_2} p_2(D) \\ \vdots \\ c_k A_{\delta\Omega_k} p_k(D) \end{bmatrix} x = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_k \end{bmatrix} \tag{5.28}$$

This is equivalent to

$$f_{N_\Lambda} = \operatorname*{arg\,min}_{g \in \Phi_{N_\Lambda}} \left(\sum_{\boldsymbol{x} \in P_\Omega} |L_1 g(\boldsymbol{x}) - h_1(\boldsymbol{x})|^2 + \sum_{j=2}^{k} c_j^2 \sum_{\boldsymbol{x} \in P_{\delta\Omega_j}} |L_j g(\boldsymbol{x}) - h_j(\boldsymbol{x})|^2 \right), \tag{5.29}$$

the weighted least squares approach mentioned in §5.1.

Remark 5.4. We note here that it is unclear how to choose the sampling sets $P_{\delta\Omega_j}$ with respect to P_Ω, and how to choose the weights c_k. When Φ_{N_Λ} consists of certain RBFs, these questions are answered in [22] under some assumptions. These are that Ω has a piecewise C^m continuous boundary, Dirichlet boundary conditions ($k = 2, L_k = I$), sufficiently smooth h_i so that the exact solution f is smooth in some sense, and elliptic L_1. Then f_{N_Λ} converges to f when Ω and $\delta\Omega$ are sampled increasingly dense with some uniformity requirements, and the convergence rate depends on the weights c_k, the dimension of Ω and the smoothness of the RBFs.

Even though precise convergence results for (5.29) are unknown, we illustrate that the least squares system (5.28) can be solved efficiently using Algorithm 3. To do so, we first notice that the sampling sets $P_{\delta\Omega_j}$ are small in size compared to P_Ω when sampled with similar densities

$$\sup_{\boldsymbol{x} \in \Omega} \min_{\boldsymbol{y} \in P_\Omega} |\boldsymbol{x} - \boldsymbol{y}| \quad \approx \quad \sup_{\boldsymbol{x} \in \delta\Omega} \min_{\boldsymbol{y} \in P_{\delta\Omega}} |\boldsymbol{x} - \boldsymbol{y}|.$$

Since $p_k(D)$ is diagonal and can be applied fast, this satisfies the first requirement, that the matrix from (5.28) can be applied reasonably efficient. For the second requirement, we first note that $A_\Omega p_1(D)$ in general does not have the required singular value profile. However, $A_\Omega p_1(D)(p_1(D))^{-1} = A_\Omega$ does, with $(p_1(D))^{-1}$ the inverse of $p_1(D)$, assuming it exists. Rewrite (5.28) as

$$\begin{bmatrix} A_\Omega \\ c_2 A_{\delta\Omega,2} p_2(D)(p_1(D))^{-1} \\ \vdots \\ c_k A_{\delta\Omega_k} p_k(D)(p_1(D))^{-1} \end{bmatrix} y = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_k \end{bmatrix} \tag{5.30}$$

and set

$$x = (p_1(D))^{-1} y.$$

Then, following Theorem 5.2, and with $n_r = CN_{\delta\Omega} \leq \eta(\tau, N_\Lambda)$, the asymptotic complexity when using Algorithm 3 to solve this system is equal to that of solving $A_\Omega x = b$.

As an example we solve the Helmholtz equation

$$\Delta f(\boldsymbol{x}) + 100^2 f(\boldsymbol{x}) = \mathrm{e}^{-400((\boldsymbol{x}_1+0.3)^2+\boldsymbol{x}_2^2)^2}, \qquad \boldsymbol{x} \in \Omega, \tag{5.31}$$

$$\frac{\partial f}{\partial \boldsymbol{n}_{\boldsymbol{x}}}(\boldsymbol{x}) = 0, \qquad \boldsymbol{x} \in \delta\Omega_1,$$

$$f(\boldsymbol{x}) = 0, \qquad \boldsymbol{x} \in \delta\Omega_2. \tag{5.32}$$

As Ω we take a two-dimensional smooth star shaped domain with a circle of diameter 0.2 cut out, with homogenous Neumann boundary conditions on the outer boundary and homogenous Dirichlet boundary conditions on the inner boundary.

The solution using the procedure outlined in this section is shown in Fig. 5.5. The approximation uses a 71×71 set of Fourier basis functions on a $[-1.2, 1.2] \times [-1.2, 1.2]$ grid (slightly larger than pictured). The oversampling was $\varrho = 1.5$, the boundary points selected by finding pairs of grid points with only one point in Ω, and using bisection to find points on the boundary.

Remark 5.5. An extension of the Kansa method is to additionally require the governing equation to hold on the boundary points (PDECB, solving PDEs with Collocation on the Boundary) [37]. In the setting of (5.28) this would add extra conditions $A_{\delta\Omega} p_1(D)x = h_1(P_{\delta\Omega})$ to the problem matrix. In our limited experiments, this did not significantly improve the error for the governing equation, nor the boundary conditions.

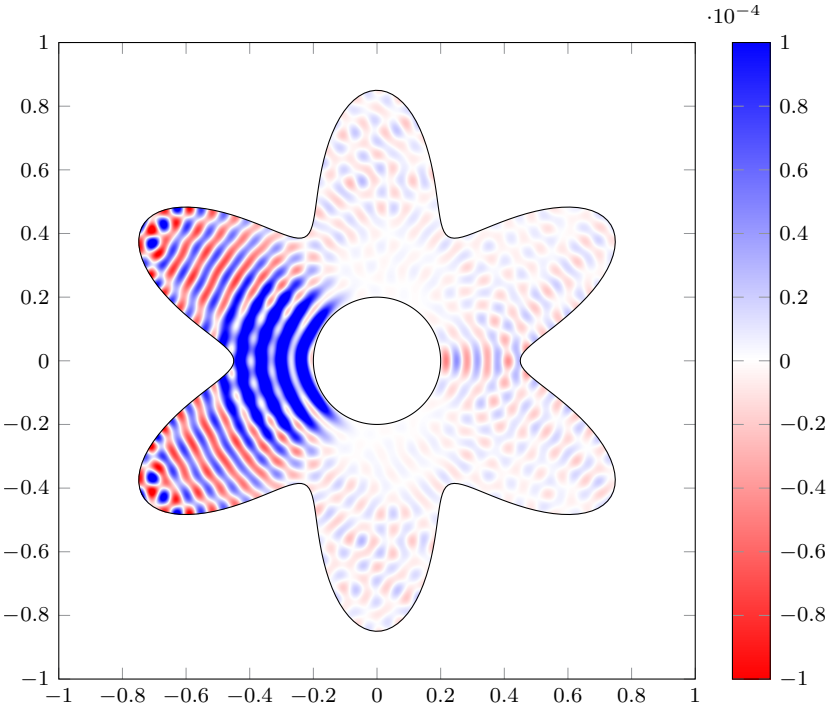


Figure 5.5: The solution of the Helmholtz equation (5.31), on smooth, non-simply connected domain.

Remark 5.6. The algorithm outlined here is a step towards a spectral method for arbitrary domains, that avoids the $\mathcal{O}(N_\Lambda^3)$ complexity of other domain-independent methods. For one-dimensional differential equations with variable coefficients very efficient spectral methods exist, based on Chebyshev series [77]. The Chebyshev approximations have so far however only been extended to rectangular [98] or spherical [99] regions.

5.2 Sobolev Smoothing

Up until now we have solved the least squares problem $Ax = b$ by computing (approximate) T-SVD solutions. With $U_{\alpha+\beta}$ as before the left singular vectors for which $\sigma_k > \tau$, the truncated SVD solution x satisfies

$$\|U_{\alpha+\beta}U_{\alpha+\beta}^*(Ax - b)\| = 0. \tag{5.33}$$

This solution is not unique. With V_γ the right singular vectors for which $\sigma_k < \tau$, we can add any $z \in \text{span}(V_\gamma)$ to x and the requirement will still hold. However, for the T-SVD solution $\|V_\gamma V_\gamma^* x\| = 0$. Therefore, among all solutions of (5.33), the T-SVD solution is the one with minimal ℓ^2 norm.

Recall from Figs. 2.3 and 4.1 that the functions corresponding to $z \in \text{span}(V_\gamma)$ have a fraction larger than $1-\tau$ of their $\mathcal{L}^2$ norm outside of Ω. The redundancy in the solution thus allows altering the behaviour on $R \setminus \Omega$. Where the T-SVD is the solution that is minimal outside Ω, we could conceivably choose an x so that the solution has other desirable properties, e.g. smoothness. This approach was introduced by Lyon in [71], and can be adapted[1] to the fast algorithm in [69]. In this section, we adapt Algorithm 3 in a similar way.

The first requirement is a notion of smoothness. Let H^k denote the k-th standard Sobolev space on R with norm

$$\|f\|_{\text{H}^k} = \left(\sum_{i=0}^{k} \int_R |f^{(i)}(x)|^2 dt \right)^{\frac{1}{2}}. \tag{5.34}$$

This space combines the function norm with norms of the derivatives up to k, resulting in a norm that indicates a degree of smoothness. It is a Hilbert space, a subspace of $\mathcal{L}^2_R$ containing functions f for which the Fourier coefficients satisfy

$$\sum_{l=-\infty}^{\infty} (1 + l^2 + l^4 + \cdots + l^{2k}) |\hat{f}[l]|^2 < \infty.$$

For a finite Fourier series $\hat{f}[l], l \in I_N$, the Sobolev norm can be computed as

$$\|f\|_{\text{H}^k} = \|D\hat{f}\|, \qquad D^2_{ij} = \begin{cases} 1 + j^2 + j^4 + \cdots + j^{2k}, & j = i \\ 0, & j \neq i \end{cases}, \quad j, i \in I_N \tag{5.35}$$

Here $\|D\hat{f}\|$ is the regular ℓ^2 norm of $D\hat{f}$. We can then define the Sobolev extension.

Definition 5.7. *The Sobolev extension, with D as in* (5.35)*, is the one with coefficients*

$$y = \arg\min_z \|Dz\|, \qquad s.\ t. \quad \|U_{\alpha+\beta} U^*_{\alpha+\beta}(Az - b)\| = 0. \tag{5.36}$$

[1] The idea for this fast smoothed algorithm was presented by M. Lyon at the ICERM research cluster on Sparse and Redundant representations, as part of the semester program on High Dimensional approximation, in 2014.

This extension is the smallest function on R in the H^k-norm, for which the coefficients satisfy (5.33).

With x the T-SVD solution, (5.36) can be restated as

$$y = \arg\min_z \|Dz\|, \qquad z = x - x_\gamma, \quad x_\gamma \in \operatorname{span}(V_\gamma).$$

Given an orthogonal basis Q for $\operatorname{span}(DV_\gamma)$, the solution to this problem is given by $y = (I - D^{-1}QQ^*D)x$, assuming D^{-1} exists. However, orthogonalizing DV_γ is a costly operation.

A possible approach stems from the realisation that V_β can be obtained in $\mathcal{O}(N_\Lambda \eta(\tau, N_\Lambda)^2)$ operations using Algorithm 1, and that the orthogonal complement of $DV_{\alpha+\gamma}$ is given by $D^{-1}V_\beta$. This allows for a projector $I - QQ^*$ onto $\operatorname{span}(DV_{\alpha+\gamma})$, using only the orthogonalised and scaled plunge region singular vectors DV_β.

The procedure is then as follows: given any x_β such that

$$\|U_\beta U_\beta^*(Ax_\beta - b)\| = 0, \tag{5.37}$$

and an orthogonal basis Q for $\operatorname{span}\{D^{-1}V_\beta\}$,

$$\tilde{x}_\beta = D^{-1}QQ^*Dx_\beta \tag{5.38}$$

is the minimiser of the Sobolev norm $\|D \cdot\|$ among all coefficients that satisfy (5.37). Indeed for any $x_{\alpha+\gamma} \in \operatorname{span}\{V_{\alpha+\gamma}\}$ we have $\langle Dx_{\alpha+\gamma}, D\tilde{x}_\beta\rangle = 0$. All that remains is then to find an $x_\alpha \in \operatorname{span}\{V_\alpha\}$ such that $\tilde{x}_\beta$ satisfies (5.33). As in the previous chapters, this can be approximated well by

$$x_\alpha = A^*(b - A\tilde{x}_\beta).$$

The full algorithm implementing this smoothing procedure is given in Algorithm 4.

Algorithm 4 Smoothed Implicit algorithm

$(AA^* - I)Ax_\beta = (AA^* - I)b$ (Algorithm 3)	$\triangleright\ \mathcal{O}(N_\Lambda r^2)$
$VSU^* = A^*(AA^* - I)$ (Algorithm 1)	$\triangleright\ \mathcal{O}(N_\Lambda r^2)$
$QR = D^{-1}V$ (reduced QR)	$\triangleright\ \mathcal{O}(N_\Lambda r^2)$
$\tilde{x_\beta} = D^{-1}QQ^*Dx_\beta$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$x_\alpha = A^*(b - A\tilde{x}_\beta)$	$\triangleright\ \mathcal{O}(N_\Lambda \log N_\Lambda)$
$x = x_\alpha + \tilde{x}_\beta$	$\triangleright\ \mathcal{O}(N_\Lambda)$

Figure 5.6 shows an interpretation of the intermediate results similar to Figs. 3.2 and 3.3. Figure 5.6a shows the Fourier series corresponding to x_β. Recall from

§3.4 that $x_\beta = V_\beta V_\beta^* x + r_\alpha + r_\gamma$, with r_α and r_γ introduced through the random vectors W. After the projection in (5.38), $\tilde{x}_\beta = V_\beta V_\beta^* x + w_\alpha + w_\gamma$, where w_α and w_γ are such that $\tilde{x}_\beta$ has minimal H^k norm (Fig. 5.6b). Indeed, while the solution is smoothed both in Ω and $R \setminus \Omega$, the region around the boundary is where all functions $\mathcal{T}_N x, x \in \operatorname{span}\{V_{\alpha+\gamma}\}$ are small, so the original solution is retained there. The final steps of the algorithm are the same as those of Algorithm 3, and are equivalent to extending the residual on Ω by zeros, and doing an inverse Fourier transform.

Remark 5.8. In Chapter 3 Algorithms 2 and 3 were shown to yield different solutions. Where Algorithm 2 approximates the T-SVD, Algorithm 3 returns a solution with extra contributions $r_\gamma \in \operatorname{span}\{V_\gamma\}$, due to the random vectors in W. However, Algorithm 4 can emulate Algorithm 2 if $D = I$. The intermediate solution x_β gets projected onto $\operatorname{span}\{V_\beta\}$, and the resulting x has minimal ℓ^2 norm.

5.2.1 Convergence of the smoothed extension

In [1, Remark 5.6] it was shown that the T-SVD solution of the discrete FE – after the initial superalgebraic or exponential convergence (see Theorem 2.24) – slowly converges to the dual frame solution. Recall from §1.4 that the dual frame coefficients are those that have minimal ℓ^2 norm. Therefore, the extension will converge to zero outside Ω as $N_\Lambda \to \infty$, albeit slowly.

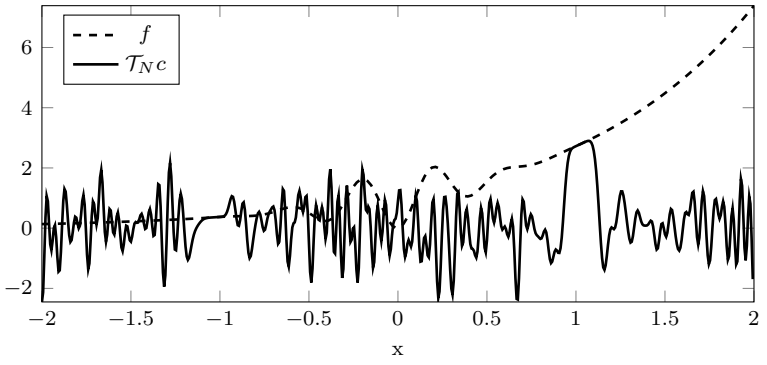
One can similarly wonder whether the exact solution obtained through Algorithm 4 converges. The most likely limit is the function that agrees with f on Ω and has overall minimal H^k norm. In the one-dimensional case, with $\Omega = [-1, 1]$, the optimal extension can be found explicitly as

$$\tilde{g} = \min_{g \in \mathcal{L}^2_{R\setminus\Omega}} \int_{R\setminus\Omega} \alpha_0(g)^2 + \alpha_1 \left(\frac{dg}{dx}\right)^2 + \alpha_2 \left(\frac{d^2 g}{dx^2}\right)^2 + \dots dx, \qquad (5.39)$$

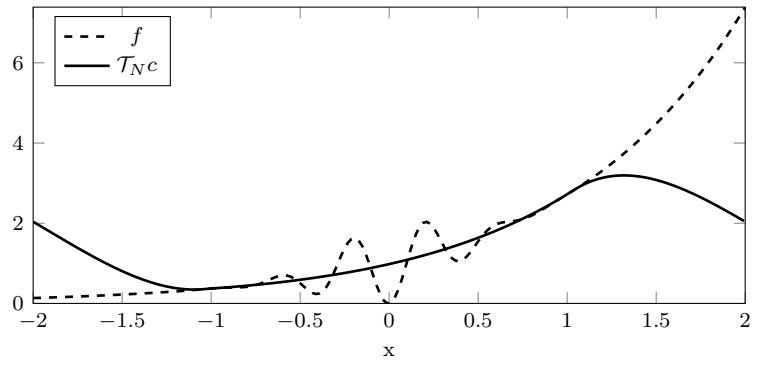
$$\tilde{g}(1) = f(1), \qquad \tilde{g}(-1) = f(1).$$

The minimiser of this functional can be found by recognizing (5.39) as an Euler-Lagrange equation, due to the periodicity on R. The extrema of a functional $\int_a^b \mathcal{V}(x, f, f', \dots)dx$ satisfy the differential equation

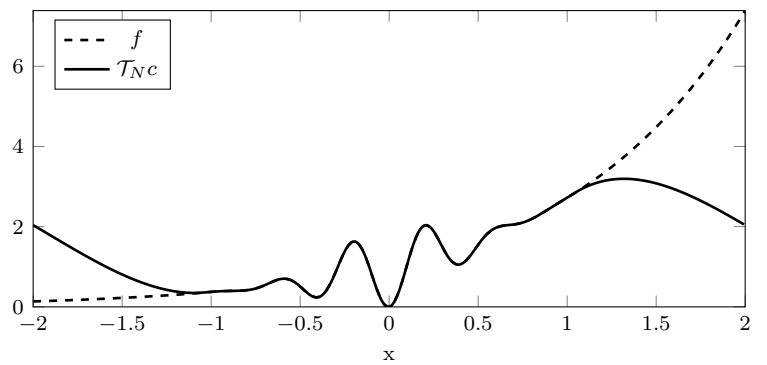
$$\frac{\partial \mathcal{V}}{\partial f} - \frac{d\partial \mathcal{V}}{dx \partial f'} + \frac{d^2 \partial \mathcal{V}}{dx^2 \partial f''} - \dots = 0,$$



(a) $\mathcal{T}_N x_\beta$



(b) $\mathcal{T}_N \tilde{x}_\beta$



(c) $\mathcal{T}_N(x_\alpha + \tilde{x}_\beta)$

Figure 5.6: Illustration of the different intermediate results in Algorithm 3. x_β represents the solution at the boundary, with added elements from the nullspace. As before, the residual vanishes smoothly at the boundary.

with boundary conditions obtained through interpolating f and its derivatives at a and b. It follows that

$$\alpha_0 \tilde{g} - \alpha_1 \frac{d^2\tilde{g}}{dx^2} + \alpha_2 \frac{d^4\tilde{g}}{dx^4} - \cdots = 0 \tag{5.40}$$

with boundary values

$$\tilde{g}(1) = f(1), \quad \tilde{g}(-1) = f(-1), \quad \tilde{g}'(1) = f'(1), \quad \tilde{g}'(-1) = f'(-1), \ldots$$

Following (5.34) $\alpha_i = 1$ when minimizing the Sobolev k norm. The minimiser $\tilde{g}$ is thus the solution of the differential equation (5.40) with $2k$ degrees of freedom. At each boundary the function value and first $k-1$ derivatives are interpolated.

We illustrate, without proof, that the smoothed extension indeed converges to $\tilde{g}$ as $N_\Lambda \to \infty$, when minimizing the H^2 norm. Equation (5.40) becomes

$$\tilde{g} + \frac{d^4\tilde{g}}{dx^4} = \frac{d^2\tilde{g}}{dx^2},$$

which for $T = 2$ has solutions

$$\tilde{g}(x) = c_1 \cos\left(\frac{x}{2}\right) \mathrm{e}^{\frac{\sqrt{3}}{2}x} + c_2 \cos\left(\frac{x}{2}\right) \mathrm{e}^{-\frac{\sqrt{3}}{2}x} + c_3 \sin\left(\frac{x}{2}\right) \mathrm{e}^{\frac{\sqrt{3}}{2}x} + c_4 \sin\left(\frac{x}{2}\right) \mathrm{e}^{-\frac{\sqrt{3}}{2}x}.$$

The coefficients c_i are found through interpolating the function values and first derivatives at the boundary. Figure 5.7 shows the smoothed extension of $f(x) = x^3 - 0.5x + \sin(10x)/10 + 2$ from $[-1, 1]$ to $[-2, 2]$ for N_Λ from 100 (yellow) to 1000 (blue). The exact minimiser $\tilde{g}$ is shown as the dashed line, and the smoothed extensions are indeed seen to converge towards this function.

Remark 5.9. When $\alpha_0 = \cdots = \alpha_{k-1} = 0$ and $\alpha_k = 1$, the Sobolev norm minimises the $\mathcal{L}^2$ norm of the k-th derivative. In this case, the solution to (5.40) is the Hermite interpolating polynomial. This is an interesting parallel to approaches where the extension is explicitly computed as Hermite interpolants [101], or other polynomials [16].

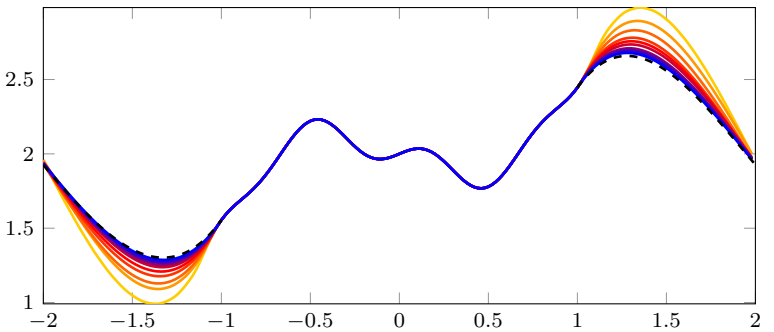


Figure 5.7: Convergence of the extension (yellow to blue for increasing N_Λ) to $\tilde{g}$ (dashed), when using Algorithm 4 to approximate $f(x) = x^3 - 0.5x + \sin(10x)/10 + 2$ with minimal H^2 norm.

Chapter 6

Implementation

The algorithms and applications from the previous chapters have been implemented and made publicly available in a JULIA package called FrameFun [1] and its dependencies BasisFunctions and Domains. The style is influenced by the MATLAB software package Chebfun[2] [32] and the JULIA package ApproxFun[3] [78], based on [77].

The idea is to represent functions internally as expansions in function sets. An end user can then manipulate function objects without having to worry about the specifics of function approximation and representation. Where the Chebfun package achieves this by representing functions using (piecewise) Chebyshev polynomials, FrameFun allows representations in frames, using the algorithms from the previous chapters where possible.

As the development of this software is ongoing, we do not provide excessive implementation or syntax details, since they are easily out of date. Instead we opt to describe the design philosophy, and how JULIA helps to write flexible and extendable code.

The third section previously appeared as [60].

[1] `https://github.com/daanhb/FrameFun.jl`
[2] `http://www.chebfun.org`
[3] `https://github.com/JuliaApproximation/ApproxFun.jl`

6.1 BasisFunctions

BasisFunctions represents the concepts from Chapter 1, and aims to stay close to the mathematical interpretation. The main objects are function sets and their spans, and operators that map elements from one finite span to another. The goal is to have operators that can be applied efficiently, but that are general enough to deal with new function sets when they are introduced.

6.1.1 FunctionSets

A FunctionSet object is any set of functions with a finite size, corresponding to a set $\{\phi_i\}_{I_N}$ from Chapter 1. Every FunctionSet can be indexed (not necessarily from 1 to N), and returns a subset of its elements or a single element. A FunctionSet is parametrised as FunctionSet{T} by a domain type T. This type corresponds to the type of a domain in the Domains package, and it is the type of the expected argument to the elements of the function set. For example, a Fourier basis with domain type an interval $[-1, 1]$ of BigFloats will return a complex BigFloat $e^{ik\pi x}$ when the k-th element is evaluated in a BigFloat $-1 \leq x \leq 1$.

Examples of FunctionSets that are implemented in BasisFunctions are: Fourier bases, sine and cosine bases, polynomial bases (Jacobi, Laguerre, Hermite, monomials), rational polynomials, certain wavelets and splines. These can be mapped, combined into augmented sets as in §5.1, combined into tensor products as higher-dimensional sets, or combined piecewise based on a set of domains.

The space spanned by linear combinations of functions in a FunctionSet F with certain type T is a Span{T,F}. Our approximations thus contain such a span, and corresponding expansion coefficients. When evaluating the approximation, the input type is determined by the FunctionSet domain, the output type is determined by the output of the individual functions and the coefficient type.

Each FunctionSet can support additional functionality: derivatives of an expansion in terms of its coefficients, efficient evaluation on certain grids, or expansion arithmetic. It can also provide an efficient operator to convert a given function into coefficients. This can be the orthogonal projection onto the span, or the interpolant in a certain sample set.

Remark 6.1. There is no inherent difference between frames and bases in this implementation. Recall from §1.4 that a truncated linearly independent frame is a Riesz basis for its span, so it is unclear how to define these concepts without resorting to FunctionSets of infinite dimension. Even deciding whether

a FunctionSet is an orthogonal basis for its span is difficult: imagine adding the cosine basis and sine basis. There is no way to generically know this FunctionSet is an orthogonal basis for the union of the spans without checking orthogonality for each pair of elements.

6.1.2 Operators

Expansions in Spans of FunctionSets are manipulated through operators on the coefficients. An operator maps the elements of one Span to another Span, and the implementation of the operator may depend on the type of the coefficients. As these operators are often used multiple times throughout computations, we have the following guidelines when implementing operators:

- An operator should be executable as efficiently as possible.
- An operator should not allocate memory when it is applied, only when it is constructed.

An operator can either be applied in place, or memory allocated for the result can be passed as an optional argument. Operators can be transposed, scaled, added and composed, provided the corresponding Spans match. Note that this provides an inherent error check, as composing operators is only allowed if it makes sense mathematically, i.e. if their domains and ranges agree. Operators may also implement (pseudo)-inversion.

We illustrate our guidelines through the calculation of the operator $(AA^* - I)A$ from Algorithm 3. The domain of the operator are the Fourier coefficients, the range is in this case the Span of a discrete grid, where the (complex) coefficients represent function samples in the grid points. With F the FFT of length N_R, E_Λ an extension operator from P_Λ to $P_{\hat{R}}$ points and R_Ω a restriction operator from P_R to P_Ω, we have

$$A = R_\Omega F E_\Lambda.$$

This composite operator allocates a vector of length N_R for the intermediate result between E_Λ and F. The adjoint A^* is computed automatically, and similarly allocates a vector of length N_R. The FFTs use the FFTW library [40], that allows pre-allocation as well. The compositions AA^*, $AA^* - I$ and $(AA^* - I)A$ then each pre-allocate a vector of length N_Λ or N_Ω for their intermediate results. The application of $(AA^* - I)A$ to r columns of a random matrix W in Algorithm 3 is therefore allocation free.

A special type of operators are diagonal operators. Since diagonality is preserved under linear combinations, composition, and pseudo-inversion, a combination

of operators is diagonal if the individual operators are diagonal. This allows us to easily combine operators such as $c_2 A_{\delta\Omega,2} p_2(D)(p_1(D))^{-1}$ in (5.30), where $p_2(D)(p_1(D))^{-1}$ will resolve to a single diagonal operator automatically.

Through the power of multiple dispatch and parametrisation, this framework can be coded very efficiently. For example, there is a generic evaluation operator that evaluates any FunctionSet on any grid by evaluating at every grid point. However, when the FunctionSet is a Fourier basis and the grid is equispaced, the same evaluation operator method specialises to use the FFT. As another example, operators only need to implement a method to apply it to a vector of the correct size; all details regarding memory allocation and comparing spans are handled at the generic operator level.

6.2 FrameFun

In FrameFun the algorithms from Chapters 3 and 4 are implemented. They are represented as an operator that maps samples on some discrete grid to coefficients in the span of some FunctionSet. When the FunctionSet is a FE frame with bounding box R and domain Ω, the subgrid will be $P_\Omega = P_R \cap \Omega$. As we will see in the next section, only a representation of the characteristic function of Ω is needed to obtain P_Ω.

When using Algorithm 3, the algorithm operator stores the SVD of the low rank matrix $\tilde{A}$ when it is constructed. Every following application then only goes through the algorithm steps following this SVD. This way, computing a second approximation with the same parameters is significantly more efficient than the first one (see Remark 3.4).

6.3 Domains

In this section we detail how domains are represented internally in Domains through the use of characteristic functions. Since our methods only require a point set P_Ω, we have no need for elaborate domain descriptions such as triangularisations or parametrisations. The characteristic function is a natural description for many domains, and allows us to do efficient domain arithmetic as well.

6.3.1 The characteristic function

The *characteristic function* χ, or *indicator function*, of a domain $\Omega \subset \mathbb{R}^n$ is a function on $\mathbb{R}^n$ that has value 1 for points that belong to Ω and the value 0 for points that do not, i.e.,

$$\chi(\boldsymbol{x}) := \begin{cases} 1, & \boldsymbol{x} \in \Omega, \\ 0, & \boldsymbol{x} \notin \Omega. \end{cases} \tag{6.1}$$

It is convenient in implementations to associate boolean values with $\chi(\boldsymbol{x})$, so that it evaluates to true or false, rather than the numeric values 1 and 0.

Representing a domain by its characteristic function has a number of consequences. Two **advantages** are:

- The function is unique and well-defined for any domain, be it open or closed, connected or disconnected, punctured, empty, a discrete set, finite or infinite, a fractal, ...
- As we will see later on, the characteristic function is often easy to implement. For example, with $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2]$ in two dimensions, the halfopen domain bounded by the parabola $\boldsymbol{x}_2 = \boldsymbol{x}_1^2$ and the straight line $\boldsymbol{x}_2 = \boldsymbol{x}_1$ has characteristic function

  $$\chi(\boldsymbol{x}) = (\boldsymbol{x}_2 > \boldsymbol{x}_1^2) \,\&\, (\boldsymbol{x}_2 \leq \boldsymbol{x}_1). \tag{6.2}$$

  There is no need even to find the intersection points of both curves, as far as implementing the characteristic function is concerned.

Consequently, it is easy and very cheap to find the characteristic function of the domain that is bounded by, say, the level curves of a given function, even if the resulting domain is disconnected and contains many holes. This operation does not even require any numerical computation, as will be demonstrated later on.

Two **disadvantages** are:

- The characteristic function does not explicitly convey information about the boundary of the domain. This would be difficult for fractal domains, but it would be convenient to have for simpler domains, and essential to have for boundary value problems. An exception are the domains in §6.5.4.
- The least squares approximation scheme requires point evaluations inside the domain. Though the characteristic function is well-defined for domains that have no volume in $\mathbb{R}^n$, such as a line in $\mathbb{R}^2$ or a surface in $\mathbb{R}^3$, the concept is not suited for approximating functions on such domains.

6.3.2 Generating points

The least squares approximation scheme requires N_Ω point evaluations of the given function f inside the domain Ω. Thus, one needs a way to find N_Ω points that belong to Ω.

It is convenient at this stage too to have at hand a bounding box R, or the knowledge of any other region R that is easily sampled for which $\Omega \subset R$. Then, points inside Ω can be generated by sampling N_R points $\boldsymbol{y}_j$ of R and checking whether $\chi_\Omega(\boldsymbol{y}_j)$ is true. This results in a set of N points with $N \leq N_R$:

$$\{\boldsymbol{x}_j\}_{j=1}^{N} := \{\boldsymbol{y}_j \,|\, \chi_\Omega(\boldsymbol{y}_j) = 1, j = 1, \ldots, N_R\}$$

Only those points are retained and the procedure is repeated with denser samplings, corresponding to increasing values of N_R, until $N \geq N_\Omega$.

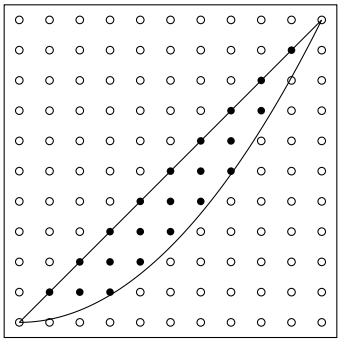


Figure 6.1: The characteristic function (6.2) evaluated in N_R points inside the bounding box R. The points are a subset of a structured equispaced grid on R.

For domains with non-zero volume in $\mathbb{R}^n$, it is guaranteed that N_Ω points will eventually be found if the sampling of R becomes uniformly denser. Though in principle any randomly chosen set of points $\{\boldsymbol{x}_j\}$ is sufficient, for efficiency reasons it is better to choose a structured set. In particular, in combination with the FE scheme we use a bounding box R and an equidistant grid on R. The main advantage is that Fourier series approximations can be evaluated efficiently on that grid with the FFT. In several examples further on, the characteristic function of a domain itself is defined in terms of a Fourier series, and in that case the characteristic function too can be evaluated efficiently on a structured grid using the FFT.

From the point of view of the approximation problem, it may be better to have more points clustered towards the boundary of the domain. However, even for

multivariate polynomial interpolation it is a very difficult problem to determine the best points on a general domain. Furthermore, since we make very few assumptions regarding our knowledge of the boundary, choosing more points near the boundary requires algorithmic work. Instead, we oversample.

6.3.3 Implementation

The elements that make up a domain include:

- a bounding box,
- a routine to evaluate the characteristic function at a single point $\boldsymbol{x}$,
- an optimised routine to evaluate the characteristic function on a regular grid of the bounding box.

The latter routine will be called the *grid evaluation* routine. It is not an essential part of the implementation, but it leads to much increased efficiency in particular when using the FE approximation scheme. The goal is not merely to exploit the speed gained from vectorisation, but to lower the computational complexity compared to calling the single evaluation routines many times.

For points on the boundary, the characteristic function can be true or false, corresponding to closed and open domains. This makes a difference in practice only in special circumstances, since in general the points that are sampled are unlikely to coincide with the boundary of the domain. In general, it is very difficult to distinguish between open and closed domains with the proposed techniques.

6.4 Computing with domains

6.4.1 Set operations

Basic set operations have rather obvious ramifications for the characteristic function. The union, difference and intersection of two domains give rise to logical relationships between the characteristic functions involved. Assume the

domains Ω_i, $i = 1, 2, 3$, have characteristic functions χ_i. Then we have

$$\Omega_3 = \Omega_1 \cup \Omega_2 \Rightarrow \chi_3(\boldsymbol{x}) = \chi_1(\boldsymbol{x}) \textbf{ or } \chi_2(\boldsymbol{x})$$

$$\Omega_3 = \Omega_1 \cap \Omega_2 \Rightarrow \chi_3(\boldsymbol{x}) = \chi_1(\boldsymbol{x}) \textbf{ and } \chi_2(\boldsymbol{x})$$

$$\Omega_3 = \Omega_1 \setminus \Omega_2 \Rightarrow \chi_3(\boldsymbol{x}) = \chi_1(\boldsymbol{x}) \textbf{ and not } \chi_2(\boldsymbol{x})$$

$$\Omega_3 = (\Omega_1 \cup \Omega_2) \setminus (\Omega_1 \cap \Omega_2) \Rightarrow \chi_3(\boldsymbol{x}) = \chi_1(\boldsymbol{x}) \textbf{ xor } \chi_2(\boldsymbol{x}).$$

These operations are easily implemented by definining χ_3 in terms of the supplied definitions of χ_1 and χ_2. Similarly, the grid evaluation routine of Ω_3 can be defined in terms of the grid evaluation routines in Ω_1 and Ω_2. This makes sure that a potentially fast implementation of this procedure for Ω_1 and Ω_2 leads to a fast implementation of this procedure also for Ω_3.

In Julia, this enables the following operations

```
» Ω3 = Ω1 & Ω2
» Ω3 = Ω1 | Ω2
» Ω3 = Ω1 \ Ω2
» Ω3 = xor(Ω1,Ω2)
```

by overloading the logical operators for domain objects.

6.4.2 Arithmetic operations

Domains can be translated and scaled by adding a vector and by multiplying by a scalar respectively. We have

$$\forall \boldsymbol{c} \in \mathbb{R}^n : \quad \Omega_2 = \Omega_1 + \boldsymbol{c} \Rightarrow \chi_2(\boldsymbol{x}) = \chi_1(\boldsymbol{x} - \boldsymbol{c})$$

$$\forall a \in \mathbb{R} : \quad \Omega_2 = a * \Omega_1 \Rightarrow \chi_2(\boldsymbol{x}) = \chi_1(\boldsymbol{x}/a).$$

It should be noted that while translation of a domain is independent of the location of the origin, scaling a domain like above does depend on the location of the origin. A circle centered around the origin would simply increase in size by a factor of a, but a circle centered at the point $[1; 0]$ would also move a factor a to the right.

Arithmetic operations are also easily implemented, by definining χ_2 in terms of the supplied definition of χ_1 and similarly for the grid evaluation routines.

In Julia we may write

```
» Ω2 = Ω1 + [1;0]

» Ω3 = 2*Ω1
```

Combined with the above, a moon-shaped domain can be defined in terms of a circle C with radius 1 by the statement

```
» moon = C \ (C + [1/2; 0])
```

Similarly, if C is centered around the origin, a domain with a hole is obtained by

```
» annulus = 2*C \ C
```

6.4.3 Implicitly defined or derived domains

Finding the level curves of a function, say the set of points where $f(\boldsymbol{x}) = 3$, requires algorithmic work and can become arbitrarily complicated depending on the complexity of the given function f. However, it is very easy to define the characteristic function of a domain that is bounded by this level curve. Say a function f is defined on Ω and the domain C is the open domain where $f(\boldsymbol{x}) > 3$. Then the characteristic function χ_C of C is given explicitly by

$$\chi_C(\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) > 3, & \forall \boldsymbol{x} \in \Omega, \\ 0, & \text{otherwise.} \end{cases}$$

The implementation of the characteristic function is defined in terms of the inequality $f(\boldsymbol{x}) > 3$, which is a boolean expression for each $\boldsymbol{x}$. The grid evaluation routine of C may be implemented in terms of the grid evaluation routine of f. Thus, if f can be evaluated efficiently via FFT for example, then the same holds for the characteristic function of the domain C.

In Julia, we may now write

```
» C = f > 3

» C = f > g

» C = cos(f .^ 2) - 3 < sqrt(pi)
```

where both f and g are existing functions. In the second statement, the domain C is in addition restricted to the intersection of the domains of f and g, such that it makes sense to compare f and g.

Interestingly, from the point of view of implementation, it is irrelevant whether or not the resulting domains are connected or not. The shape of the resulting domain can be truly arbitrary and does not effect the computational cost of this new characteristic function. Of course, the geometry of the domain does play a role in the approximation problem to be solved, see Chapter 4.

6.4.4 Deciding on the equivalence of domains

When given two characteristic functions χ_1 and χ_2, the problem of deciding whether they represent the same domain is a difficult one and requires careful consideration. It is of course not possible to check for each and every point $\boldsymbol{x} \in \mathbb{R}^2$ whether $\chi_1(\boldsymbol{x})$ equals $\chi_2(\boldsymbol{x})$. Two possible ways to treat this problem are as follows.

1) Verify equivalence up to a certain resolution The characteristic functions χ_1 and χ_2 are sampled on an equidistant grid with a certain specified resolution and covering both domains. Their equivalence at this resolution level is determined by their equivalence at the grid points.

2) Compare domain identifiers In JULIA it is easy to overload the equality operator by having it compare attributes of the domain type, for example the radius and center of two circles need to agree for them to be the same.

The first approach is costly and does not always give the right mathematical answer, in the sense that it may conclude equivalence for two domains that are not equivalent. It will never conclude inequivalence for equivalent domains. However, the approach applies to all domains and will always converge to the correct answer when increasing the resolution level.

The second approach is fast, but does not always give the right mathematical answer, as two domains may be constructed in similar ways but independently of each other. types will be different, though the domains may be the same. For example, the union of two rectangles may or may not be another rectangle. Avoiding this situation requires care from the user of the software.

6.5 Examples

6.5.1 Characteristic function

For some domains the characteristic function is simply the most convenient description. The Mandelbrot set is an example, defined by

$$\chi(\boldsymbol{x}) = \left(\limsup_{n\to\infty} |z_{n+1}| \leq 2 \right),$$

$$z_{n+1} = z_n^2 + \boldsymbol{x}_1 + \mathrm{i}\boldsymbol{x}_2, \qquad z_0 = 0.$$

An approximation of

$$f_m(\boldsymbol{x}) = \cos(20\boldsymbol{x}_1 + \mathrm{i}\boldsymbol{x}_2) - 5\boldsymbol{x}_1\boldsymbol{x}_2 \tag{6.3}$$

is shown in Fig. 6.2a. It was obtained using an equispaced grid on $[-2, 2] \times [-1.5, 1.5]$. Using the Fourier Extension technique, convergence up to a tolerance of 10^{-12} was achieved for 32×32 basis functions (Fig. 6.2b).

6.5.2 Domain arithmetic

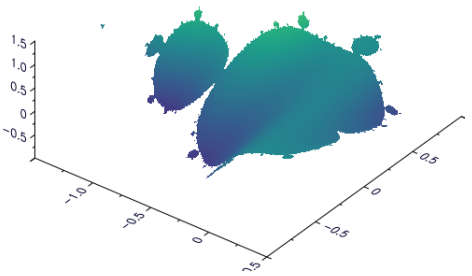
As an example of computing with domains, Fig. 6.3 shows an approximation on a ring, obtained by the Julia commands

```
» Ω3 = disk(0.9) \ disk(0.5)
```

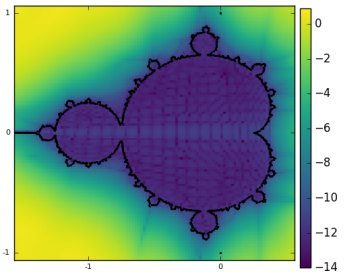
Of special note here is that the target function

$$f_r(\boldsymbol{x}) = \frac{\boldsymbol{x}_1}{\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2} \tag{6.4}$$

has a singularity at $(0, 0)$, enclosed in the domain. However, this has little influence on the approximation, as the exterior of the domain is never sampled. As Fig. 6.3b shows the approximation converges up to a tolerance of 10^{-10} for 32×32 basis functions.

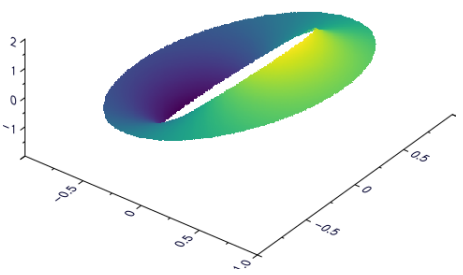


(a) Approximation

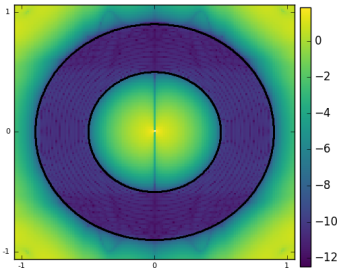


(b) Error

Figure 6.2: An approximation of f_m ((6.3)) on the Mandelbrot set. The right figure shows $\log_{10}(|f_m - \mathcal{P}^{\tau}_{N_\Omega,N_\Lambda} f_m|)$. The approximation error is very small precisely on the Mandelbrot set. In the extension region, the functions f_m and $\mathcal{P}^{\tau}_{N_\Omega,N_\Lambda} f_m$ are both defined and they can be evaluated and compared, but they bear no resemblance. In particular, $\mathcal{P}^{\tau}_{N_\Omega,N_\Lambda} f_m$ is periodic on the box, while f_m is not.

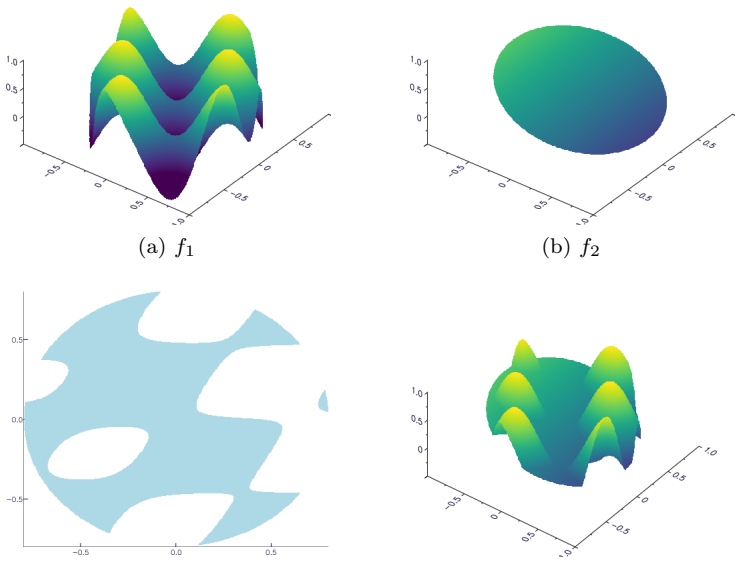


(a) Approximation



(b) Error

Figure 6.3: An approximation of f_r ((6.4)) on a ring-shaped domain. The right figure shows $\log_{10}(|f_f - \mathcal{P}^{\tau}_{N_\Omega,N_\Lambda} f_r|)$.



(a) f_1 (b) f_2

(c) $f_2 > f_1$ (d) $f = \max(f_1, f_2)$

Figure 6.4: A piecewise approximation of F ((6.5)), and the implicit domain $f_2 > f_1$.

6.5.3 Implicitly defined domains

More convoluted domains occur when trying to approximate a function such as

$$f_1(\boldsymbol{x}) = \sin(5\boldsymbol{x}_1 - 3\boldsymbol{x}_2)\sin(7\boldsymbol{x}_1)$$

$$f_2(\boldsymbol{x}) = -0.5\boldsymbol{x}_1 + 0.2$$

$$f(\boldsymbol{x}) = \max(f_1(\boldsymbol{x}), f_2(\boldsymbol{x})) \tag{6.5}$$

on a disk. Given approximations $\tilde{f}_1$ and $\tilde{f}_2$ of f_1 and f_2 on the full disk (Figs. 6.4a to 6.4b), $\tilde{f}$ is simply

$$\tilde{f}(\boldsymbol{x}) = \begin{cases} \tilde{f}_1(\boldsymbol{x}), & \tilde{f}_1(\boldsymbol{x}) \geq \tilde{f}_2(\boldsymbol{x}) \\ \tilde{f}_2(\boldsymbol{x}), & \tilde{f}_1(\boldsymbol{x}) < \tilde{f}_2(\boldsymbol{x}) \end{cases}.$$

In this case, evaluating $\tilde{f}$ (Fig. 6.4d) or the characteristic function (Fig. 6.4c) is straightforward, and fast on an equispaced grid, since only one full evaluation of $\tilde{f}_1$ and $\tilde{f}_2$ is required.

6.5.4 Polar coordinates

For some two-dimensional domains, such as the starlike shape from §5.1.4, the natural characteristic function is

$$\chi(\boldsymbol{x}) = \sqrt{\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2} < \phi(\theta), \qquad \theta = \operatorname{atan2}(\boldsymbol{x}_2, \boldsymbol{x}_1).$$

If $\phi(\theta)$ is a 2π periodic function, this defines a smooth domain. Moreover, the normal direction on a boundary point of this domain is given by

$$\boldsymbol{n}_{\boldsymbol{x}} = \left(\phi'(\theta)\sin(\theta) + \phi(\theta)\cos(\theta), -\phi'(\theta)\cos(\theta) + \phi(\theta)\sin(\theta)\right).$$

To obtain the required derivative ϕ', we can approximate ϕ by a Fourier series on $[-\pi, \pi]$, and apply a (diagonal) derivative operator to the coefficients to obtain an approximation for ϕ'. The Neumann boundary conditions for the examples in §5.1.4 and §7.5 were obtained in this fashion.

Chapter 7

Contributions and Future work

In this final chapter we compile a list of concrete realisations made in this thesis, along with some associated open problems. We then leave the reader with a few ideas for future work, and some remarks on work in progress.

7.1 Contributions

- We have explored the connection between Fourier frame approximations and Prolate Spheroidal Wave functions in Chapter 2. Though this connection was known, it had not yet been used in algorithm design. To the best of our knowledge the observation that the P-DPSS are eigenvectors of the discrete FE collocation matrix is new.
- We used a known property of Prolate Spheroidal Wave functions and their generalisations, to come up with Algorithm 2. Applicable only to one-dimensional Fourier extensions, this was nonetheless an improvement over existing fast algorithms, as it added flexibility in choosing the extension length.
- We developed Algorithm 3, that depends only on a particular singular value distribution in the collocation matrix. In Chapter 3 we showed the utility of this approach in one dimension. Then in Chapter 4 we showed that the two-dimensional FE problem has this singular value profile as

well. The one-dimensional approach has been used in [109] to compute fast convolutions of functions with compact support.

- In Chapter 5, Algorithm 3 was used as the foundation for more involved problems, such as approximation in augmented frames, solving boundary value problems, and smoothing the solution.
- In Chapter 6 we showed characteristic functions can represent domains to the degree needed by the algorithms. This minimal representation is highly flexible.
- We implemented this in an open-source Julia package, that allows straightforward experimentation with our algorithms. This is currently being developed and used by the researchers on frame approximations at KU Leuven.

These contributions have been published, or are in preparation for publishing, as

- Matthysen, R., and Huybrechs, D. Fast Algorithms for the computation of Fourier Extensions of arbitrary length. *SIAM J. Sci. Comput. 38*, 2 (2015), A899–A922
- Huybrechs, D., and Matthysen, R. *Computing with Functions on Domains with Arbitrary Shapes*. Springer International Publishing, Cham, 2017, pp. 105–117
- Matthysen, R., and Huybrechs, D. Function approximation on arbitrary domains using Fourier Extension frames. *SIAM J. Numer. Anal. (accepted)* (2018)
- Matthysen, R., and Huybrechs, D. Fast algorithms for augmented Fourier extensions. *(in preparation)* (2018)

However some open problems remain:

- The bound for two dimensional plunge regions in Theorem 4.17 overestimates the constant, judging by Fig. 4.12. In particular, we conjecture that τ^{-1} is replaceable by $\log \tau^{-1}$. However, Lemma 4.16 always leads to a factor τ^{-1}, that is also present in [106, 67]. Improving the constant will likely require a new proof technique such as that used recently in [111], using more than just the trace iterates.

- A deeper analysis of the weighted least squares formulation for boundary value problems is required, particularly with respect to the weights and density requirements for the sample sets in the domain and on the boundary.
- The error analysis for the algorithm in §3.6 should be extended to include perturbations of singular vectors as well.
- The two dimensional Fourier Extensions converge in the discrete ℓ^2 norm over P_Ω. However, little is known for convergence in the $\mathcal{L}^2_\Omega$-norm or $\mathcal{L}^\infty_\Omega$-norm. Judging from Fig. 4.13, the answer will be related to the domain geometry, at least for the latter norm.

7.2 Future work

This section outlines possible extensions of the ideas in this thesis that are currently under investigation. Some of them (§§7.3 and 7.4) are work in progress, as a joint effort by the researchers working on frame approximations at KU Leuven (Daan Huybrechs, Marcus Webb and Vincent Coppé).

7.3 Adaptivity

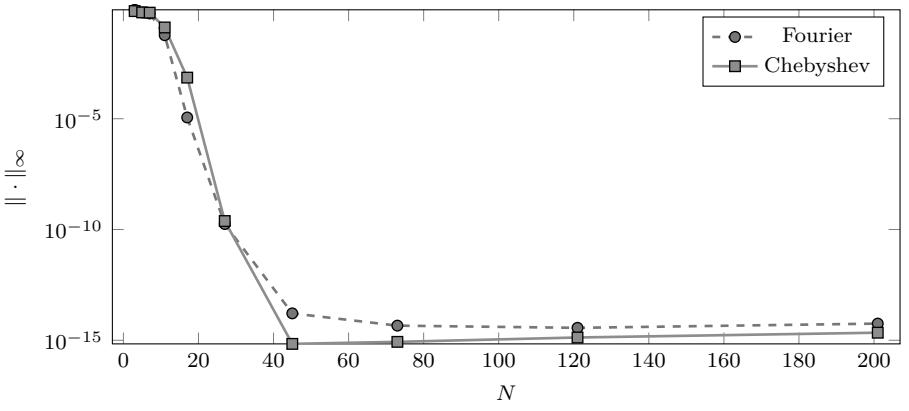
In the first chapter, we mentioned a desirable property of approximations in orthonormal bases: decay of the expansion coefficients. Knowing the decay rate allows for error estimations based on the last coefficients

$$\|f - \mathcal{P}_N f\|^2 = \sum_{i \notin I_N} |c_i|^2,$$

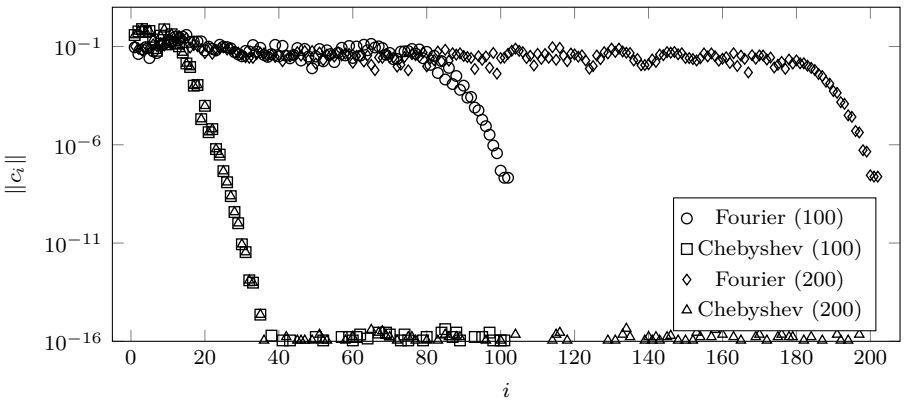
see §1.2. This way the expansion can be truncated when the coefficients have decayed to machine precision, leading to an optimal number of degrees of freedom in the representation, as in Chebfun [32].

A downside of the FE frame is the absence of such decay for the coefficients. Figure 7.1 compares approximations for the test function $f(x) = \mathrm{e}^x$ on $[-1, 1]$ using both a Chebyshev basis on $[-1, 1]$ and a FE frame on $[-2, 2]$. While the convergence behavior in Fig. 7.1a is comparable, that of the coefficients is not: the Chebyshev coefficients clearly reach machine precision before $N = 40$. The FE coefficients on the other hand show no clear decay pattern.

This makes FE approximations difficult to truncate, as there are no negligible coefficients. An optimal-length FE can still be obtained using adaptive



(a) error



(b) coefficient size

Figure 7.1: Approximation error as a function of N and coefficient size for 100 and 200 degrees of freedom, for Chebyshev and Fourier Frame approximations.

algorithms that can increase N until a certain error tolerance is met. Such a procedure would require both an incremental algorithm and efficient, robust error estimation. These are both current areas of research.

7.4 Polynomial spectrum mapping

Some frames give rise to collocation matrix spectra that have a plunge region, but no clustering, or more than one cluster. An example is the truncated frame on $[-1, 1]$

$$\Phi_{2N_\Lambda} = \left\{ \frac{1}{\sqrt{2}} \mathrm{e}^{\mathrm{i}\pi n x} \right\}_{I_N} \cup \{T_j(x)\}_{j=1,\ldots,N_\Lambda}. \tag{7.1}$$

combining Fourier and Chebyshev basisfunctions. When collocated on an equispaced grid, the collocation matrix has a spectrum where about a quarter of the singular values are above 1. The singular values do not cluster near 1, but they are bounded from above by the collocation matrix norm, in this case $\|A\|_2 = 4$. As upcoming work [103] shows, there is a plunge region $\eta(\tau, N_\Lambda, a) = \mathcal{O}(\log N_\Lambda)$ for this matrix, where

$$\eta(\tau, N_\Lambda, a) = \min(k - j) \quad \text{s.t.} \quad \sigma_j \geq a - \tau, \quad \tau > \sigma_k. \tag{7.2}$$

and $a = 1$. Unfortunately, it is not immediately clear how to isolate this plunge region. The mapping $A(A^*A - I)$ used in Algorithm 3 will not result in a low rank system. However, there exist functions $p(AA^*)$, so that the mapping $\mathcal{W}(\sigma) = p(\sigma^2)\sigma$ results in a low rank system. Specifically, with a as in (7.2) and $b = \|A\|_2$, $p(\sigma^2)\sigma$ should be small for $\sigma \in [a, b]$ and $\sigma \sim 0$. If p is a polynomial, it can be evaluated for AA^* in $\mathcal{O}(kN_\Lambda \log N_\Lambda)$ operations, with k the order of the polynomial. A possible choice of p is the Chebyshev polynomial on $[a^2, b^2]$, scaled so that $T_k(0) = 1$

$$P(AA^*) = \frac{T_k((b^2 + a^2 - 2AA^*)/(b^2 - a^2))}{T_k((b^2 + a^2)/(b^2 - a^2))}. \tag{7.3}$$

This polynomial has a maximum value of ε on $[a, b]$ if

$$\varepsilon \leq T_k((b^2 + a^2)/(b^2 - a^2)),$$

which happens for polynomial degree approximately

$$k \gtrsim \frac{\log \varepsilon/2}{\log((b - a)/(b + a))} + 1. \tag{7.4}$$

This way the matrix $P(AA^*)A$ is of low numerical rank, as all singular values in $[0, \varepsilon] \cup [a - \varepsilon, b]$ are mapped to ε or below, see Fig. 7.2 for an illustration

of the mapped spectrum. As before, when subtracting the image under A of the solution, the right hand side has only contributions remaining that are in $\text{span}(\{U_k\}), \sigma_k \in [a, b]$. This means that an iterative solver like LSQR will converge a lot quicker, with error estimate

$$\|Ax_m - b\| = \left(\frac{b-a}{b+a}\right)^m \|Ax_0 - b\|,$$

as opposed to (3.1). We also note that the Chebyshev iteration method could be used here as well, which is an iterative method that relies on building a polynomial such as (7.3) [45]. This method requires an estimate of the spectrum bounds a and b over other iterative algorithms, but these are necessary for (7.3) as well and are usually known from the frame definition. This leads to a modified Algorithm 3, shown as Algorithm 5. Assuming the iterative method converges quickly, the cost is dominated by forming the matrix $T_k(AA^*)AW$ in the second step, with an $\mathcal{O}(kN_\Lambda \log^2 N_\Lambda)$ cost for a logarithmically growing plunge region.

Algorithm 5 Use of the polynomial spectrum mapping.

$W = \text{rand}(N_\Lambda, r + 20)$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$\tilde{A} = T_k(AA^*)AW$	$\triangleright\ \mathcal{O}(krN_\Lambda \log N_\Lambda)$
$USV^* = \tilde{A}$	$\triangleright\ \mathcal{O}(N_\Lambda r^2)$
$y = V(S_\tau^\dagger(U^*(T_k(AA^*)b)))$	$\triangleright\ \mathcal{O}(N_\Lambda r + kN_\Lambda \log N_\Lambda)$
$x_\beta = Wy$	$\triangleright\ \mathcal{O}(N_\Lambda r)$
$Ax_\alpha = (b - Ax_\beta)$ (Iterative method)	$\triangleright\ \mathcal{O}(N_\Lambda \log N_\Lambda)$
$x = x_\alpha + x_\beta$	$\triangleright\ \mathcal{O}(N_\Lambda)$

Note that this algorithm yields exactly the projection algorithm when used for Fourier Extensions or similar frames that have a single singular value cluster. However, a disadvantage is that the degree k from (7.4) is rather high. To map $[1, 4]$ to machine precision, a polynomial of degree about 70 is needed. This obviously increases the cost considerably. Though numerical experiments indicate this is a feasible approach, additional efforts are needed to analyse the error, the influence of the iterative solvers, and to attempt to lower the considerable extra factor added to the algorithm cost.

7.5 Integration with time-stepping methods

Another possible future application is using the boundary value problem solver from §5.1.4 as a spectral method in space, and combining it with a finite order method in time. Recall that for the Fourier coefficients, a differential operator

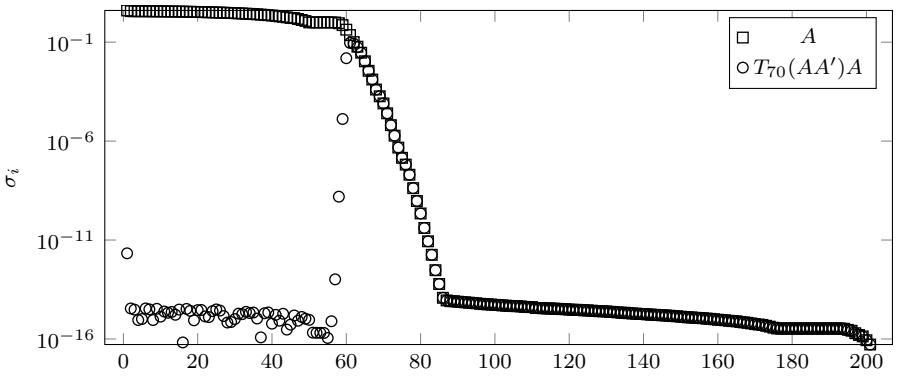


Figure 7.2: The spectrum of the collocation matrix A for the frame (7.1), along with the spectrum of $T_{70}(AA^*)A$. Note that the spectrum of A is different from that in Fig. 2.2, the singular values do not cluster near 1.

with constant coefficients $\hat{L}_s = p_s(D)$ is diagonal. When looking at a partial differential equation of the form

$$\frac{\partial u}{\partial t} = L_1 u, \tag{7.5}$$

linear systems involving the operator $\hat{L}_1$ on the coefficients arise when discretising the partial differential equation in time using an implicit method. Let $u(t,x)$ be the solution to (7.5), and let $\hat{u}_j$ denote the Fourier coefficients of $u_j(x)$ at a specific time t_j. Discretizing (7.5) using the trapezoidal rule with timestep Δt yields

$$u_{n+1} = u_n + \Delta t/2(L_1 u_{n+1} + L_1 u_n)$$

$$(I - \frac{\Delta t}{2} L_1) u_{n+1} = (I + \frac{\Delta t}{2} L_1) u_n.$$

Boundary conditions are incorporated using the weighted least squares approach, as in §5.1.4. Noting the similarity to (5.29), the conditions in the coefficient space

$$A_\Omega (I - \frac{\Delta t}{2} \hat{L}) \hat{u}_{n+1} = A_\Omega (I + \frac{\Delta t}{2} \hat{L}) \hat{u}_n, \tag{7.6}$$

$$c_j A_{\delta\Omega_j} \hat{L}_j \hat{u}_{n+1} = h_j, \qquad j = 2, \ldots, k \tag{7.7}$$

constitute a boundary value problem such as those in §5.1.4. Solving these weighted least squares formulations was done through Algorithm 3, calculating

an approximate pseudo-inverse of the system

$$\begin{bmatrix} A_\Omega \\ c_2 A_{\delta\Omega,2} p_2(D)(I - \frac{\Delta t}{2}\hat{L})^{-1} \\ \vdots \\ c_k A_{\delta\Omega_k} p_k(D)(I - \frac{\Delta t}{2}\hat{L})^{-1} \end{bmatrix} \left((I - \frac{\Delta t}{2}\hat{L})\hat{u}_{n+1} \right) = \begin{bmatrix} A_\Omega(I + \frac{\Delta t}{2}\hat{L})\hat{u}_n \\ h_{2,n+1} \\ \vdots \\ h_{k,n+1} \end{bmatrix}.$$

Note that this pseudo-inverse can be applied in $O(N\eta(\tau, N_\Lambda))$ time, compared to the $O(N\eta(\tau, N_\Lambda)^2)$ calculation cost (see Remark 3.4). This makes a single time-step considerably more efficient than the pre-computation step, which is approximately the cost of an approximation problem.

There are questions remaining regarding the stability of this method. Preliminary results show the possibility of eigensolutions to (7.6) and (7.7) for which $u_{n+1} = \lambda u_n, \lambda > 1$, where this behaviour is unphysical. However, this only occurs for some combination of differential operators L_k, least squares weights c_k and time step Δt. Further, the λ's did not exceed $1 + \sqrt{\tau}$. For $\tau = \varepsilon_{\text{mach}}$, it then takes $\log_{1+\sqrt{\varepsilon_{\text{mach}}}}(2) \sim 10^8$ steps for the initial vector to double in magnitude. The interplay between the different parameters causing this behaviour is a topic of current research.

As an example of a partial differential equation that can easily be solved through this method, we solve the two-dimensional wave equation

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \tag{7.8}$$

on a star-shaped domain, with homogenous Neumann boundary conditions. The inital condition is the Gaussian

$$f(x, y) = \frac{1}{2}\mathrm{e}^{-20(x^2+y^2)}.$$

$\hat{L}_1$, as before, is an elementwise operator that is easily inverted. To discretise the second order derivative in time we introduce the extra variable

$$v = \frac{\partial u}{\partial t}.$$

Using again the trapezoidal rule for what is now a system of differential equations yields

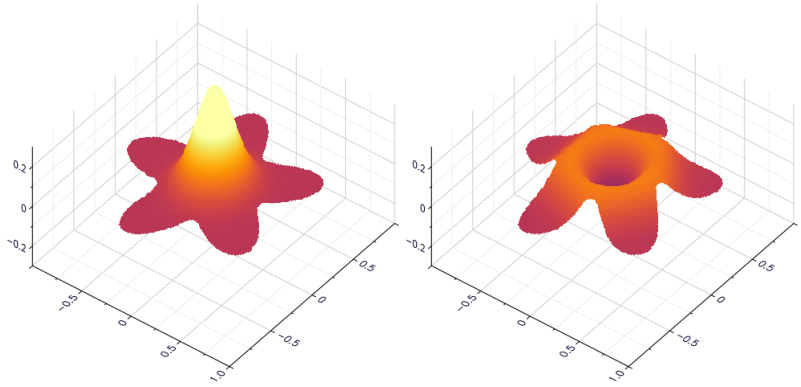
$$u_{n+1} = u_n + \frac{\Delta t}{2}(v_{n+1} + v_n)$$

$$v_{n+1} = v_n + \frac{\Delta t}{2}(Lu_{n+1} + Lu_n),$$

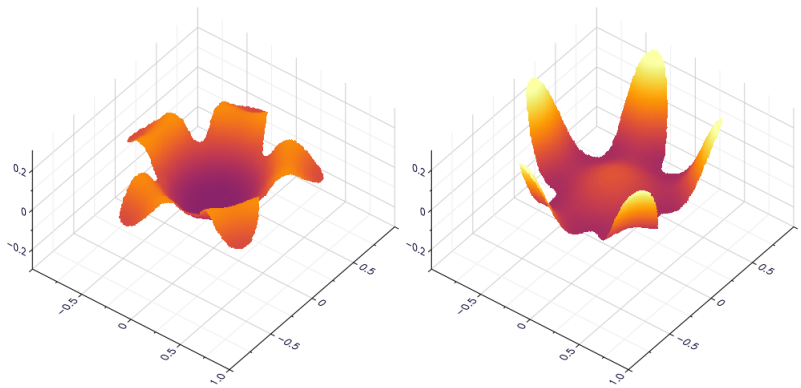
from which v_{n+1} can be eliminated, resulting in

$$u_{n+1} = u_n + \frac{\Delta t}{2}\left(2v_n + \frac{\Delta t}{2}(L_1 u_{n+1} + L u_n)\right).$$

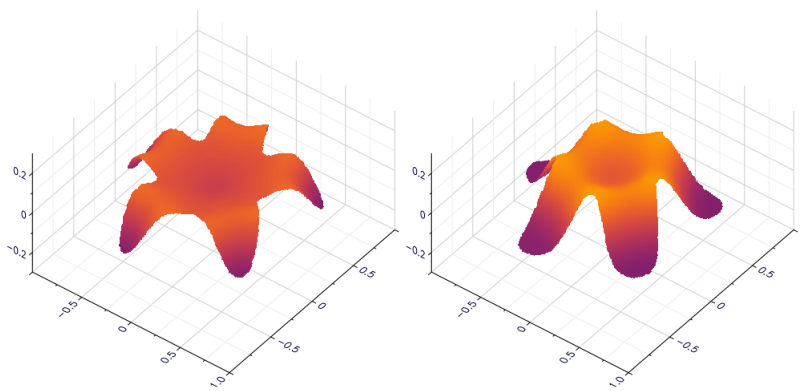
Proceeding as before for u_{n+1} and updating v_{n+1} accordingly leads to the results shown in Fig. 7.3. The parameters for this simulation were $\tau = 10^{-14}$, $N_\Lambda = 31^2$ and $\varrho = 2$, $\Delta t = 0.005$ and $c_j = 1$.



(a) $k = 0$ (b) $k = 40$



(c) $k = 80$ (d) $k = 120$



(e) $k = 160$ (f) $k = 180$

Figure 7.3: The solution $u_k(x, y)$ to the wave equation (7.8), after k timesteps.

Bibliography

[1] Adcock, B., and Huybrechs, D. Frames and numerical approximation. ArXiV:1612.04464.

[2] Adcock, B., and Huybrechs, D. On the resolution power of Fourier extensions for oscillatory functions. *J. Comput. Appl. Math. 260* (2014), 312 – 336.

[3] Adcock, B., and Huybrechs, D. Frames and numerical approximation II: generalized sampling. *ArXiv e-prints* (Feb. 2018).

[4] Adcock, B., Huybrechs, D., and Martín-Vaquero, J. On the numerical stability of Fourier extensions. *Found. Comp. Math. 14* (2014), 635–687.

[5] Adcock, B., and Ruan, J. Parameter selection and numerical approximation properties of Fourier extensions from fixed data. *J. Comput. Phys. 273* (Sept. 2013), 1–23.

[6] Albin, N., and Bruno, O. P. A spectral FC solver for the compressible Navier-Stokes equations in general domains I: Explicit time-stepping. *J. Comput. Phys. 230*, 16 (2011), 6248–6270.

[7] Atkinson, K. E. *Spherical harmonics and approximations on the unit sphere: an introduction.* Lecture notes in mathematics, 2044. Springer, Berlin ; New York, 2012.

[8] Baglama, J., Reichel, L., and Richmond, D. An augmented LSQR method. *Numer. Algorithms 64*, 2 (Oct. 2013), 263–293.

[9] Borcea, L., Papanicolaou, G., and Vasquez, F. G. Edge illumination and imaging of extended reflectors. *SIAM J. Imag. Sci. 1*, 1 (2008), 75–114.

[10] Boyd, J. P. *Chebyshev and Fourier spectral methods*, revised ed. Dover Publications, 2001.

[11] Boyd, J. P. A Comparison of Numerical Algorithms for Fourier Extension of the First, Second, and Third Kinds. *J. Comput. Phys. 178*, 1 (may 2002), 118–160.

[12] Boyd, J. P. Prolate spheroidal wavefunctions as an alternative to Chebyshev and Legendre polynomials for spectral element and pseudospectral algorithms. *J. Comput. Phys. 199*, 2 (sep 2004), 688–716.

[13] Boyd, J. P. Fourier embedded domain methods: extending a function defined on an irregular region to a rectangle so that the extension is spatially periodic and C^∞. *Appl. Math. Comput. 161*, 2 (2005), 591–597.

[14] Bruno, O. Fast, high-order, high-frequency integral methods for computational acoustics and electromagnetics. In *Topics in Computational Wave Propagation*, vol. 31. Springer, Berlin, 2003, pp. 43–82.

[15] Bruno, O., Han, Y., and Pohlman, M. Accurate, high-order representation of complex three-dimensional surfaces via Fourier continuation analysis. *J. Comput. Phys. 227*, 2 (dec 2007), 1094–1125.

[16] Bruno, O. P., and Lyon, M. High-order unconditionally stable FC-AD solvers for general smooth domains I. Basic elements. *J. Comput. Phys. 229*, 6 (mar 2010), 2009–2033.

[17] Bueno-Orovio, A. Fourier embedded domain methods: Periodic and C^∞ extension of a function defined on an irregular region to a rectangle via convolution with Gaussian kernels. *Appl. Math. Comput. 183*, 2 (2006), 813–818.

[18] Buhmann, M. *Radial basis functions: theory and implementations*. Cambridge monographs on applied computational mathematics 12. Cambridge University Press, Cambridge, 2003.

[19] Casazza, P. G., and Christensen, O. Approximation of the inverse frame operator and applications to gabor frames. *Journal of Approximation Theory 103*, 2 (April 2000), 338–356.

[20] Chamzas, C. *On the extrapolation of band-limited signals*. PhD thesis, Polytechnic Institute of New York, 1980.

[21] Chen, L. *Digital and Discrete Geometry: Theory and Algorithms*. Springer Publishing Company, Incorporated, 2014.

[22] Cheung, K.-C., Ling, L., and Schaback, R. H^2–Convergence of least-squares kernel collocation methods. *ArXiv e-prints* (Jan. 2018).

[23] Christensen, O. Frames and the projection method. *Applied and Computational Harmonic Analysis 1*, 1 (December 1993), 50–53.

[24] Christensen, O. Frames containing a riesz basis and approximation of the frame coefficients using finite-dimensional methods. *Journal of Mathematical Analysis and Applications 199*, 1 (April 1996), 256–270.

[25] Christensen, O. *Frames and Bases: An Introductory Course*. Applied and Numerical Harmonic Analysis. Birkhäuser Boston, Boston, 2008.

[26] Christensen, O. *An Introduction to Frames and Riesz Bases*, 2nd ed. 2016. ed. Applied and Numerical Harmonic Analysis. Springer International Publishing : Imprint: Birkhäuser, Cham, 2016.

[27] Christensen, O., and Strohmer, T. The finite section method and problems in frame theory. *Journal of Approximation Theory 133*, 2 (2005), 221–237.

[28] de Boor, C. *A practical guide to splines*. Applied mathematical sciences 27. Springer, New York (N.Y.), 1978.

[29] Demmel, J., Gu, M., Eisenstat, S., Slapničar, I., Veselić, K., and Drmač, Z. Computing the singular value decomposition with high relative accuracy. *Linear Algebra and its Applications 299*, 1 (1999), 21 – 80.

[30] Demmel, J. W., Marques, O. A., Parlett, B. N., and Vmel, C. Performance and accuracy of lapack's symmetric tridiagonal eigensolvers. *SIAM Journal on Scientific Computing 30*, 3 (2008), 1508–1526.

[31] Dhillon, I. S. *A New Algorithm for the Symmetric Tridiagonal Eigenvalue Eigenvector Problem*. PhD thesis, University of California, Berkeley, 1997.

[32] Driscoll, T. A., Hale, N., and Trefethen, L. N. *Chebfun Guide*. Pafnuty Publications, 2014.

[33] Duffin, R., and Schaeffer, A. A class of nonharmonic Fourier series. *Trans. Amer. Math. Soc. 72* (1952), 341–366.

[34] Eckhoff, K. S. Accurate and efficient reconstruction of discontinuous functions from truncated series expansions. *Mathematics of Computation 61*, 204 (October 1993), 745–763.

[35] Edelman, A., McCorquodale, P., and Toledo, S. The future fast Fourier transform? *SIAM Journal on Scientific Computing 20*, 3 (1998), 1094–1114.

[36] Falconer, K. *Fractal Geometry*, vol. 11. John Wiley & Sons, Ltd, Chichester, UK, sep 1990.

[37] Fedoseyev, A., Friedman, M., and Kansa, E. Improved multiquadric method for elliptic partial differential equations via pde collocation on the boundary. *Computers & Mathematics with Applications 43*, 3 (2002), 439 – 455.

[38] Fornberg, B., and Flyer, N. *A primer on radial basis functions with applications to the geosciences*. SIAM, 2015.

[39] Fornberg, B., and Piret, C. On choosing a radial basis function and a shape parameter when solving a convective PDE on a sphere. *J. Comput. Phys. 227*, 5 (2008), 2758–2780.

[40] Frigo, M., and Johnson, S. G. The design and implementation of FFTW3. *Proceedings of the IEEE 93*, 2 (2005), 216–231. Special issue on "Program Generation, Optimization, and Platform Adaptation".

[41] Fuselier, E. J., Narcowich, F. J., Ward, J. D., and Wright, G. B. Error and stability estimates for surface-divergence free rbf interpolants on the sphere. *Mathematics of Computation 78*, 268 (2009), 2157–2186.

[42] Gantmakher, F. R., and Kreĭn, M. G. *Oscillation matrices and kernels and small vibrations of mechanical systems*, vol. 345. American Mathematical Soc., 2002.

[43] Gelb, A., and Tanner, J. Robust reprojection methods for the resolution of the Gibbs phenomenon. *Applied and Computational Harmonic Analysis 20*, 1 (Jan. 2006), 3–25.

[44] Gerchberg, R. Super-resolution through error energy reduction. *Optica Acta: International Journal of Optics 21*, 9 (nov 1974), 709–720.

[45] Golub, G. H., and Van Loan, C. F. *Matrix computations*, 3rd ed. ed. Johns Hopkins studies in the mathematical sciences. Johns Hopkins university press, Baltimore, 1996.

[46] Gottlieb, D., and Orszag, S. A. *Numerical analysis of spectral methods: theory and applications*, vol. 26. Siam, 1977.

[47] Gottlieb, D., and Shu, C. On the Gibbs phenomenon and its resolution. *SIAM review 39*, 4 (1997), 644–668.

[48] Gruenbacher, D., and Hummels, D. A simple Algorithm for generating discrete prolate spheroidal sequences. *Signal Processing, IEEE Transactions on 42*, 11 (1994), 3216–3218.

[49] Grünbaum, F., and Miranian, L. The magic of the prolate spheroidal functions in various setups. *Proc. SPIE 4478* (2001), 152.

[50] Grünbaum, F. A. Toeplitz Matrices commuting with tridiagonal matrices. *Linear Algebra and its Applications 36* (1981), 25–36.

[51] Gustafsson, B., Kreiss, H.-O., and Oliger, J. *Time dependent problems and difference methods*. Pure and applied mathematics. Wiley, New York (N.Y.), 1995.

[52] Halko, N., Martinsson, P., and Tropp, J. Finding structure with randomness : Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review 53*, 2 (2011), 217–288.

[53] Hansen, P. C. The discrete picard condition for discrete ill-posed problems. *BIT Numerical Mathematics 30*, 4 (Dec 1990), 658–672.

[54] Hansen, P. C. Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank. *SIAM Journal on Scientific and Statistical Computing 11*, 3 (May 1990), 503–518.

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

[56] Hansen, P. C. *Rank-deficient and discrete ill-posed problems : numerical aspects of linear inversion*. SIAM monographs on mathematical modeling and computation. SIAM, Philadelphia (Pa.), 1998.

[57] Hein, S., and Zakhor, A. Theoretical and numerical aspects of an SVD-based method for band-limiting finite-extent sequences. *Signal Processing, IEEE Transactions on 42*, 5 (1994), 1227–1230.

[58] Hogan, J., and Lakey, J. *Duration and Bandwidth Limiting: Prolate Functions, Sampling, and Applications*. Birkhäuser Boston, 2012.

[59] Huybrechs, D. On the Fourier extension of nonperiodic functions. *SIAM J. Numer. Anal. 47*, 6 (2010), 4326–4355.

[60] Huybrechs, D., and Matthysen, R. *Computing with Functions on Domains with Arbitrary Shapes*. Springer International Publishing, Cham, 2017, pp. 105–117.

[61] Jain, A., and Ranganath, S. Extrapolation algorithms for discrete signals with application in spectral estimation. *Acoustics, Speech and Signal Processing 29*, 4 (aug 1981), 830–845.

[62] Kansa, E. Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics—I surface approximations and partial derivative estimates. *Computers and Mathematics with Applications 19*, 8 (1990), 127–145.

[63] Kansa, E. Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics—II solutions to parabolic, hyperbolic and elliptic partial differential equations. *Computers and Mathematics with Applications 19*, 8 (1990), 147–161.

[64] Kato, T. *Perturbation theory for linear operators*. Springer Science & Business Media, 2012.

[65] Landau, H. On Szegö's eigenvalue distribution theorem and non-Hermitian kernels. *J. Anal. Math. 28*, 1 (1975), 335–357.

[66] Landau, H., and Pollak, H. Prolate spheroidal wave functions, Fourier analysis and uncertainty - II. *Bell System Tech. J.* (1961).

[67] Landau, H., and Widom, H. Eigenvalue distribution of time and frequency limiting. *J. Math. Anal. Appl. 77*, 2 (1980), 469–481.

[68] Liberty, E., Woolfe, F., Martinsson, P.-G., Rokhlin, V., and Tygert, M. Randomized algorithms for the low-rank approximation of matrices. *Proc. Natl. Acad. Sci. USA 104*, 51 (dec 2007), 20167–72.

[69] Lyon, M. A fast algorithm for Fourier continuation. *SIAM J. Sci. Comput. 33*, 6 (2011), 3241–3260.

[70] Lyon, M. Approximation error in regularized SVD-based Fourier continuations. *Appl. Numer. Math. 62*, 12 (dec 2012), 1790–1803.

[71] Lyon, M. Sobolev smoothing of SVD-based Fourier continuations. *Appl. Math. Lett. 25*, 12 (dec 2012), 2227–2231.

[72] Lyon, M., and Bruno, O. P. High-order unconditionally stable FC-AD solvers for general smooth domains II. Elliptic, parabolic and hyperbolic PDEs; theoretical considerations. *J. Comput. Phys. 229*, 9 (may 2010), 3358–3381.

[73] Matthysen, R., and Huybrechs, D. Fast Algorithms for the computation of Fourier Extensions of arbitrary length. *SIAM J. Sci. Comput. 38*, 2 (2015), A899–A922.

[74] Matthysen, R., and Huybrechs, D. Fast algorithms for augmented Fourier extensions. *(in preparation)* (2018).

[75] Matthysen, R., and Huybrechs, D. Function approximation on arbitrary domains using Fourier Extension frames. *SIAM J. Numer. Anal. (accepted)* (2018).

[76] Mirsky, L. Symmetric gage functions and unitarily invariant norms. *Quarterly Journal of Mathematics 11* (1960), 50–59.

[77] Olver, S., and Townsend, A. A fast and well-conditioned spectral method. *SIAM Review 55*, 3 (2013), 462–489.

[78] Olver, S., and Townsend, A. A practical framework for infinite-dimensional linear algebra. *Proceedings of the 1st Workshop for High Performance Technical Computing in Dynamic Languages* (September 2014), 57–62.

[79] Osipov, A., Rokhlin, V., and Xiao, H. *Prolate Spheroidal Wave Functions of Order Zero*, vol. 187 of *Applied Mathematical Sciences*. Springer US, Boston, MA, 2013.

[80] Paige, C. C., and Saunders, M. A. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Softw. 8*, 1 (Mar. 1982), 43–71.

[81] Papoulis, A. A new algorithm in spectral analysis and band-limited extrapolation. 735–742.

[82] Platte, R. B., and Gelb, A. A Hybrid Fourier–Chebyshev method for partial differential equations. *J. Sci. Comput. 39*, 2 (may 2009), 244–264.

[83] Platte, R. B., Trefethen, L. N., and Kuijlaars, A. B. J. Impossibility of Fast Stable Approximation of Analytic Functions from Equispaced Samples. *SIAM Review 53*, 2 (Jan. 2011), 308–318.

[84] Pollack, H., and Slepian, D. Prolate spheroidal wave functions, Fourier analysis and uncertainty -III. *Bell System Tech. J.* (1961).

[85] Pollack, H., and Slepian, D. Prolate spheroidal wave functions, Fourier analysis and uncertainty -IV. *Bell System Tech. J.* (1961).

[86] Queiró, J. F. On the interlacing property for singular values and eigenvalues. *Linear Algebra and its Applications 97*, Supplement C (1987), 23 – 28.

[87] Simons, F. J., and Dahlen, F. A. Spherical slepian functions and the polar gap in geodesy. *Geophys. J. Int. 166*, 3 (2006), 1039–1061.

[88] Simons, F. J., and Wang, D. V. Spatiospectral concentration in the Cartesian plane. *GEM - International Journal on Geomathematics 2*, 1 (2011), 1–36.

[89] Slepian, D. Prolate spheroidal wave functions, Fourier analysis, and uncertainty -V: The Discrete Case. *Bell System Tech. J* (1978).

[90] Slepian, D. Some comments on fourier analysis, uncertainty and modeling. *SIAM Review 25*, 3 (1983), 379–393.

[91] Slepian, D., and Pollak, H. Prolate spheroidal wave functions, Fourier analysis and uncertainty—I. *Bell System Tech. J.* (1961).

[92] Sobolev, A. V. Wiener–Hopf operators in higher dimensions: The Widom conjecture for piece-wise smooth domains.

[93] Sobolev, A. V. Quasi-classical asymptotics for pseudodifferential operators with discontinuous symbols: Widom's conjecture. *Funct. Anal. Appl. 44*, 4 (2010), 313–317.

[94] Stewart, G. W. Perturbation theory for the singular value decomposition. technical report CS-TR 2539, university of Maryland, 1990.

[95] Strohmer, T. On discrete band-limited signal extrapolation. *Contemporary Mathematics 190* (1995), 323–323.

[96] Swartz, B., and Wendroff, B. The relation between the galerkin and collocation methods using smooth splines. *SIAM Journal on Numerical Analysis 11*, 5 (1974), 994–996.

[97] Temes, G. C. The prolate filter: An ideal lowpass filter with optimum step-response. *Journal of the Franklin Institute 293*, 2 (1972), 77–103.

[98] Townsend, A., and Trefethen, L. N. An extension of Chebfun to two dimensions. *SIAM Journal on Scientific Computing 35*, 6 (2013), C495–C518.

[99] Townsend, A., Wilber, H., and Wright, G. B. Computing with functions in spherical and polar geometries I. the sphere. *SIAM Journal on Scientific Computing 38*, 4 (2016), 403–425.

[100] Trefethen, L. N. *Approximation Theory and Approximation Practice*. Society for Industrial and Applied Mathematics, 2013.

[101] Van Buggenhout, N. Fast Fourier extension, 2017.

[102] Varah, J. The prolate matrix. *Linear Algebra Appl. 187* (jul 1993), 269–278.

[103] Webb, M. On the plunge region for trig-like bases. *To Appear* (2018).

[104] Weyl, H. Das asymptotische verteilungsgestez der eigenwert linearer partieller differentialgleichungen (mit einer anwendung auf der theorie der hohlraumstrahlung). *Mathematische Annalen 71* (1912), 441–479.

[105] Widom, H. *On a Class of Integral Operators with Discontinuous Symbol*. Birkhäuser Basel, Basel, 1982, pp. 477–500.

[106] Wilson, R. Finite prolate spheroidal sequences and their applications 1: Generation and properties. *Pattern Analysis and Machine Intelligence, IEEE ...*, 6 (1987), 787–795.

[107] Witten, R., and Candès, E. Randomized algorithms for low-rank matrix factorizations: Sharp performance bounds. *Algorithmica 72*, 1 (May 2015), 264–281.

[108] Woolfe, F., Liberty, E., Rokhlin, V., and Tygert, M. *A fast randomized algorithm for the approximation of matrices*, vol. 25. Nov. 2008.

[109] Xu, K., Austin, A. P., and Wei, K. A fast algorithm for the convolution of functions with compact support using Fourier extensions. *SIAM J. Sci. Comput* (2017).

[110] Xu, W., and Chamzas, C. On the periodic discrete prolate spheroidal sequences. *SIAM J. Appl. Math. 44*, 6 (1984), 1210–1217.

[111] Zhu, Z., Karnik, S., Davenport, M. A., Romberg, J., and Wakin, M. B. The eigenvalue distribution of discrete periodic time-frequency limiting operators. *IEEE Signal Processing Letters 25*, 1 (Jan 2018), 95–99.

[112] Zohar, S. The solution of a toeplitz set of linear equations. *Journal of the ACM (JACM) 21*, 2 (April 1974), 272–276.

FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
NUMA
Celestijnenlaan 200A box 2402
B-3001 Leuven