FAST ALGORITHMS FOR THE COMPUTATION OF FOURIER EXTENSIONS OF ARBITRARY LENGTH

ROEL MATTHYSEN, DAAN HUYBRECHS

Abstract.
Fourier series of smooth, non-periodic functions on $[-1, 1]$ are known to exhibit the Gibbs phenomenon, and exhibit overall slow convergence. One way of overcoming these problems is by using a Fourier series on a larger domain, say $[-T, T]$ with $T > 1$, a technique called Fourier extension or Fourier continuation. When constructed as the discrete least squares minimizer in equidistant points, the Fourier extension has been shown shown to converge at least superalgebraically in the truncation parameter N. A fast $\mathcal{O}(N \log^2 N)$ algorithm has been described to compute Fourier extensions for the case where $T = 2$, compared to $\mathcal{O}(N^3)$ for solving the dense discrete least squares problem. We present two $\mathcal{O}(N \log^2 N)$ algorithms for the computation of these approximations for the case of general T, made possible by exploiting the connection between Fourier extensions and Prolate Spheroidal Wave theory. The first algorithm is based on the explicit computation of so-called periodic discrete prolate spheroidal sequences, while the second algorithm is purely algebraic and only implicitly based on the theory.

1. Introduction. Fourier series are a good choice for the approximation of a smooth periodic function on a bounded interval. They offer exponential convergence, good frequency resolution, and the approximation can be computed numerically via the FFT. However, when the function is smooth but non-periodic, the exponential convergence of a Fourier series over the interval is lost, and ringing artefacts known as the Gibbs phenomenon are introduced.

The Fourier extension technique (FE) [5, 6, 7, 8] aims to transfer the desirable properties of Fourier series for periodic functions to the non-periodic case. The principle is to approximate a non-periodic function that is defined on $[-1, 1]$ by a Fourier series that is periodic on $[-T, T]$. While the approximation may vary wildly in $[-T, -1[$ and $]1, T]$, under certain conditions it is guaranteed to converge exponentially to the original function within the interval. An illustration is shown in Figure 1.1, where the extension is seen to agree closely with the given function on $[-1, 1]$. Outside this interval, the extension is arbitrary, and in most cases defined by the solution method.

The main difficulty with this technique is the ill-conditioning of the restricted Fourier basis. Numerically, this leads to ill-conditioned linear systems, which are difficult to solve efficiently.
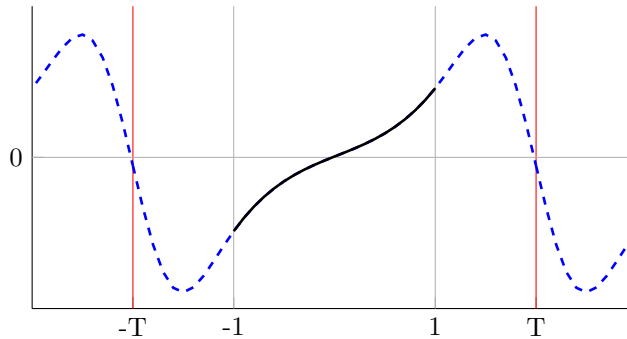

Fig. 1.1: A periodic extension to $[-T, T]$ of a smooth function on $[-1, 1]$.

These constructions have been known in embedded or fictitious domain methods for general bases. A study on the approximation properties of extensions in the Fourier basis by Boyd [5] revealed that inside the smaller interval, the extension can be exponentially converging to the function. Further, he proposed the truncated singular value decomposition as a robust method for computing extensions from equispaced data. The resulting scheme, named FPIC-SU, can compute extensions of functions in the smaller interval, that are exact almost up to machine precision. The convergence rate is only

---

limited by the smoothness of the function. Bruno et. al. used the same principle when approximating surfaces by Fourier series on extended domains[7]. This provided a starting point for the very efficient FC-Gram method [9, 25, 4].

Exponential error convergence of the FE problem was proven in [18] when inverting the Grammian matrix of the continuous least squares problem. Later, Adcock et. al extended the convergence analysis to the discrete least squares problem on equispaced data. At least superalgebraic convergence was proven for analytic functions, when using the truncated SVD [2].

From an implementation point of view, the cost of the full SVD required for the FPIC-SU is prohibitively large. Recently, an FE algorithm was introduced by Lyon [23] that computes extensions in $\mathcal{O}(N \log^2 N)$ time. However, this algorithm only produces extensions of double length, i. e. it is unique to the case $T = 2$. Due to the reliance on symmetries only present when T is a power of 2, the algorithm cannot easily be extended to arbitrary T.

In this paper we present fast algorithms for the computation of Fourier extensions of arbitrary extension length. One argument for varying the parameter T is found in the resolution power of the extension. The number of degrees of freedom per wavelength to represent an oscillatory function approaches the optimal value of 2 as T approaches 1 from above [1]. When $T = 2$, that number has already doubled. Other arguments may be performance related, as one may for example tune the length of the FFTs that are used in the computations, or a restriction on the data may be found in the application itself.

Our algorithms stem from connecting the FE problem with classical results from signal processing theory. We state how it is essentially equivalent to the problem of bandlimited extrapolation. Central in this discussion are the so-called Prolate Spheroidal Wave functions, originally introduced at Bell labs in the 1970s [30, 21, 22]. The study of these functions and their special properties has been an active domain in signal processing since.

The connection with Prolate Spheroidal Wave theory leads to explicit formulations for eigenvectors of the FE problem. Analysis of the FE problem learns that the relevant ill-conditioning can be captured using just $\mathcal{O}(\log N)$ of these vectors. Combined with a fast solver for the remaining well-conditioned problem this leads to $\mathcal{O}(N \log^2 N)$ solvers.

We explore two approaches in detail. In the first approach, a set of $\mathcal{O}(\log N)$ *discrete prolate spheroidal sequences* is explicitly computed. This is based on their known properties: the vectors are known to be eigenvectors of a tridiagonal matrix. The second approach is purely algebraic and is not explicitly based on prolate spheroidal wave theory. Hence, this approach is more generally applicable. Both approaches have comparable performance, with a slight edge for the first approach.

1.1. Overview of the paper. We formally state the FE problem in §2 and summarize relevant previous results. A concise overview of Prolate Spheroidal Wave functions is presented in §3, including subsequent work on discrete variants. The connection between these discrete variants and the FE problem is used to obtain fast algorithms in §4. Finally, we present numerical experiments to illustrate the numerical performance of these algorithms in §5.

2. Fourier extensions.

2.1. Problem formulation. For the remainder of this paper we will focus on infinitely differentiable non-periodic functions f on the interval $[-1, 1]$. Other intervals are easily dealt with through affine transformations.

The approximation is constructed on the extended interval $[-T, T]$, with T the extension parameter. For ease of notation denote

$$\phi_k(x) = \frac{1}{\sqrt{2T}} e^{\mathrm{i} \frac{k\pi}{T} x}.$$

Then

$$\mathcal{G}_N = \{\phi_k\}_{k=-n,\ldots,n}$$

2

is the set of $N = 2n + 1$ Fourier basis functions on the extended interval. The Fourier extension problem is now formalised as finding an approximation $F_N(f)$ such that

$$F_N(f) = \min_{g \in \mathcal{G}_N} ||f - g||_{[-1,1]}. \tag{2.1}$$

We note that the infinite set $\mathcal{G}_\infty$ is a basis for $\mathcal{L}^2$ on the extended interval $[-T, T]$, but it constitutes a *frame* on $[-1, 1]$ [12], i.e. it is redundant. This reflects the fact that a function can be extended in different ways. Once truncated to finite N, $\mathcal{G}_N$ is actually a basis on $[-1, 1]$, albeit a very ill-conditioned one. We refer to [2] for a detailed discussion of this point of view.

In an implementation context, it is more natural to consider the search for the Fourier coefficients $a \in \mathbb{C}^N$,

$$a = \min_{c \in \mathbb{C}^N} ||f - \sum_{k=-n}^{n} c_k \phi_k||_{[-1,1]}.$$

Depending on the norm to be minimised, the extension problem takes on a different formulation:

Discrete Fourier extensions. Most practical applications provide information about f as samples in a predefined set of points. The norm in (2.1) is then conveniently replaced by a discrete least squares norm

$$\tilde{F}_N(f) = \min_{g \in \mathcal{G}_N} \sum_l (f(x_l) - g(x_l))^2, \qquad x_l \in [-1, 1]. \tag{2.2}$$

Previous work distinguishes between uniform sampling, and an optimal sampling set called *mapped symmetric Chebychev nodes* [1]. The focus of this paper is on the case where the sampling points are uniform,

$$x_l = \frac{l}{m}, \qquad l = -m, \ldots, m,$$

for a total of $M = 2m + 1$ points. Typically, some oversampling γ is used so $M = \gamma N > N$. Including appropriate normalisation, this leads to the matrix formulation that is used throughout the rest of the paper. Let

$$A_{l,k} = \phi_k(x_l) = \frac{1}{\sqrt{2Tm}} e^{i\frac{\pi k l}{Tm}}, \quad b_l = \frac{1}{\sqrt{m}} f(x_l), \qquad l = -m, \ldots, m, \quad k = -n, \ldots, n. \tag{2.3}$$

Note that matrix-vector products involving A can be performed very efficiently using FFTs of length $L = 2Tm$. Therefore L is assumed to be integer for the remainder of this paper.

The Fourier coefficients are then found by collocation, through solving the rectangular system

$$Aa \approx b \tag{2.4}$$

in a least squares sense. Note that due to the nature of the problem we are not interested in a specific solution a, just one of the possibly many solutions that lead to a small residual $||Aa - b||$. Solving this problem efficiently is the focal point of this paper.

Continuous Fourier extensions. When the norm is the regular $\mathcal{L}^2$-norm over $[-1, 1]$,

$$\overline{F}_N(f) = \min_{g \in \mathcal{G}_N} ||f - g||_{2,[-1,1]}$$

is known as the *continuous Fourier extension*. The resulting least-squares problem can be solved by formulating the Grammian matrix. Let

$$\bar{A}_{i,j} = \int_{-1}^{1} \phi_i(x)\overline{\phi_j(x)}dx = \frac{\sin\frac{\pi(i-j)}{T}}{\pi(i - j)}, \quad b_i = \int_{-1}^{1} f\overline{\phi_i(x)}dx, \qquad i, j = -n, \ldots, n. \tag{2.5}$$

Then the Fourier coefficients follow from

$$\bar{A}a = b. \tag{2.6}$$

3

2.2. Convergence and stability. The usefulness of approximation schemes hinges on two properties: the speed of convergence to the given function, and the stability of the required computations. Considerable effort has been put into quantifying these properties both analytically and numerically [18] [1] [2] [24]. Without going into too much detail, we recap the most important results.

First of all, a distinction should be made between the exact discrete and continuous FE solutions $\tilde{F}_N(f)$ and $\overline{F}_N(f)$, and their computer-implemented counterparts. Computing the exact solutions is known to be unstable, as they can grow unbounded outside the interval of interest. Numerical algorithms however will never compute these exact solutions. Due to regularisation, the numerical FEs $\tilde{G}_N(f)$ and $\overline{G}_N(f)$ are more stable, while maintaining the desired convergence behaviour. In [2], the truncated SVD was used as a model scheme for solving the ill-conditioned systems (2.3) and (2.5). Given the Singular Value Decomposition of the FE matrix $A = USV'$, the solution to $Aa = b$ is

$$a = VS^\dagger U'b, \qquad S_{i,i}^\dagger = \begin{cases} \frac{1}{S_{i,i}} & S_{i,i} > \tau \\ 0 & \text{otherwise.} \end{cases} \tag{2.7}$$

The truncation parameter τ then acts as a regularisation parameter. It is usually chosen close to the machine precision.

Stability. Following [2], stability is defined in terms of the absolute condition number of the FE mapping

$$\kappa(F_N) = \sup\{||F_N(b)|| : b \in \mathbb{C}^N, ||b|| = 1\},$$

where, with slight abuse of notation, $F_N(b)$ is the solution to the FE problem with right hand side b, and $||\cdot||$ is the regular l^2 norm over $[-1, 1]$. This condition number can be computed for the continuous and discrete FEs, both the exact and numerical versions. It was shown to grow exponentially in N for the exact solution to the continuous and discrete FE problem.

When looking at the numerical FEs the situation changes considerably. The condition number of the numerical continuous FE mapping is $\kappa(\overline{G}_N) \lesssim 1/\sqrt{\tau}$. The condition number of the numerical discrete FE $\kappa(\tilde{G}_N)$ is dependent on a constant $0 < a(\gamma; T) \leq 1$, independent of N, that satisfies $a(\gamma; T) \to 0$ as $\gamma \to \infty$ for fixed T. It is given by $\kappa(\tilde{G}_N) \lesssim \tau^{-a(\gamma;T)}, \quad \forall N \in N$. This means that for a sufficiently large oversampling factor γ, the condition number of the numerical FE mapping can be made reasonably close to 1.

Meanwhile, the condition number of the matrices A and $\bar{A}$ grows exponentially as $N \to \infty$, where $M \geq N$ for the discrete FE. 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 [2].

Convergence. Concerning convergence, results in [2] are valuable only for functions that are analytic in a region $\mathcal{D}(\rho^*)$ of the complex plane and continuous on its border. This region is a Bernstein ellipse under a transformation that allows Fourier extensions to be understood as polynomial approximations. For such functions f, the exact continuous and discrete FEs converge geometrically, with a speed

$$||f - \tilde{F}_N(f)|| \leq c_f \rho^{-N}.$$

Here $\rho = \min\{\rho^*, E(T)\}$ and c_f is proportional to $\max_{x \in \mathcal{D}(\rho)} |f(x)|$. $E(T)$ is known as the Fourier extension constant and is given by $\cot^2\left(\frac{\pi}{4T}\right)$. Note that for the exact discrete FE, there is an added requirement of scaling M as $O(N^2)$, to avoid the Runge phenomenon.

Under the same analyticity conditions, the error decay of the numerical counterparts $\overline{G}_N$ and $\tilde{G}_N$ can be broken down into several subregions:

1. If $N < N_2$, where N_2 is a function-independent breakpoint, $||f - \tilde{G}_N(f)||$ converges or diverges exponentially fast at the same rate as the exact solution.

2. When $N \leq N_0$ (continuous) or $N_2 \leq N \leq N_1 := 2N_0$ (discrete), where N_0 is another function-independent breakpoint depending, both $||f - \overline{G}_N(f)||$ and $||f - \tilde{G}_N(f)||$ decay like ρ^{-N}.

3. When $N = N_0$ or $N = N_1$, the errors are approximately

$$||f - \overline{G}_{N_0}(f)|| \approx c_f(\sqrt{\tau})^{d_f}, ||f - \tilde{G}_{N_1}(f)|| \approx c_f \tau^{d_f - a(\gamma;T)},$$

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 $\tau^{1-a(\gamma;T)}$ 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 (2.5), this makes the continuous FE unfit for practical use. However, the algorithms presented in §4 for the discrete FE can be adapted to this context with little extra effort. This is documented in §4.3.

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

2.3. Influence of the extension length. The main contributions of this paper are algorithms that add flexibility in the choice of extension length T. The increased resolution power was already cited as an argument to reduce T, but it is important to be aware of the possible consequences. Therefore, 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 ρ is limited by this constant. Increasing T thus increases the convergence rate, and vice versa.

The resolution power of a scheme, first studied by Gottlieb and Orszag [15] is a measure of the amount of point samples needed to resolve an oscillatory function to a certain precision. Let

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

for some small δ. Then F_N 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.

A theoretical argument shows that for the continuous FE this resolution constant increases with T [1]. 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, specifically

$$T(N, \epsilon_{tol}) = \frac{\pi}{4}\left(\arctan(\epsilon_{tol})^{\frac{1}{2N}}\right)^{-1}. \tag{2.8}$$

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 [8], Bruno et. al. suggest the values $T = 2$ and $\gamma = 2$ as a general rule of thumb, but at the same time notes that the optimal parameters are heavily function dependent. Note that increasing both the extension length T and the oversampling γ will likely increase the resolution constant r. Especially since it was shown in [1] 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 Chebychev points, it is indicative that the resolution constant for $T = 2, \gamma = 2$ will be considerably above the optimal value.

The precise interplay between T and γ on the one hand, and resolution power and conditioning on the other hand was studied in detail by Adcock and Rua [3]. They found that the condition number $\kappa(\tilde{G}_N)$ of the equispaced discrete FE depends only on the product $T\gamma$. Increasing either will lower the condition number. Thus as long as γ 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.

Furthermore, the resolution constant is also dependent on the product $T\gamma$, growing as $r(T) \sim T\gamma$. This illustrates the tradeoff between resolution power and conditioning. Numerical experiments in [3] 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 [3] is the possibility to vary T with M, to achieve optimal resolution power in a manner similar to (2.8). 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 paper warrant a closer look at the possible benefits from this method.

REMARK 1. *The fast algorithms in this paper all depend on the application of the FFT, requiring $L = T(M-1)$ to be integer. Strictly speaking, this means that T is restricted to be a rational number and, hence, T is not completely arbitrary. Furthermore, in applications where the number of function samples M is fixed in advance, T can only vary in increments of $1/(M-1)$. For large M, the latter restriction is not very limiting. The restriction to a rational number, say $T = p/q$, means that the methods will be more practical if the denominator q is not too large, since large q would necessitate large values of L and M.*

3. Prolate spheroidal wave functions and discrete variants. A long-standing problem in signal processing theory is that of bandlimited extrapolation. The problem is, assuming some portion of a bandlimited signal is known, to accurately predict the missing data. In a first subsection we explain how the FE problem is a specific variant of this problem. The subsequent sections then explore the theory of Prolate Spheroidal Wave functions, that plays a major role in bandlimited extrapolation, for further use in the FE algorithms.

3.1. Discrete Fourier extensions and bandlimited extrapolation. The discrete Fourier extension is closely related to discrete band limited extrapolation, i.e., to the problem of reconstructing a discrete bandlimited signal from a number of data samples. Simply put, we are looking for a vector y with discrete Fourier transform Y such that

$$y[k] \approx f[k] \qquad k \in S_{M,t}$$
$$Y[l] = 0 \qquad l \notin S_{N,\omega},$$

where f is the sample data, and $S_{M,t}$ and $S_{N,\omega}$ are sampling sets in the discrete time and frequency domains, of sizes M and N respectively. Depending on the problem parameters the solution might not be unique. In this case an additional minimal solution norm constraint can be added [19].

A popular method is the Papoulis-Gerschberg algorithm [14] [26]. 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 [31]. They operate at a cost of $O(N \log N)$ operations per iteration, where the number of iterations scales with the bandwidth of the signal.

Besides these iterative methods, considerable attention was given to direct methods for solving this discrete problem. One such method is the one proposed by Jain and Ranganath [19], where both the time and frequency sampling sets are contiguous, i. e. $S_t = -m, \ldots, m$ and $S_\omega = -n, \ldots, n$. They

commence by writing the data as a function of the unknown coefficients:

$$f = Jy = D_M B_N y \tag{3.1}$$

$$D_{M,i,j} = \delta_{i,j}, \quad |i|, |j| \le m \tag{3.2}$$

$$B_{N,p,l} = \frac{1}{L} \sum_{k=-n}^{n} \exp \mathrm{i} \frac{(p-l)2\pi k}{L} = \frac{1}{L} \frac{\sin\left(\frac{(p-l)N\pi}{L}\right)}{\sin\left(\frac{(p-l)\pi}{L}\right)}. \tag{3.3}$$

Here, B_N is a $L \times L$ circulant matrix that represents a discrete low-pass filter, and D_M is a $M \times L$ selection operator. $\delta_{i,j}$ is the kronecker delta. D'_M is an extension operator that pads a sequence of length M with zeros to length L.

The direct methods of Jain and Ranganath then consist of solving $Jy = f$ in a least squares sense by formulating the normal equations

$$J'Jy = J'f.$$

Since $J'J$ is symmetric positive definite and Toeplitz, the Levinson-Trench algorithm can be applied to compute the inverse of $J'J$ in $O(L^2)$ operations. They also suggested another approach, using the singular value decomposition of J to solve the least-squares problem. The resulting singular vectors were named periodic discrete prolate spheroidal sequences (P-DPSSs), after the prolate spheroidal wave functions (PSWFs) arising in continuous bandlimited extrapolation.

A closer look at the matrix A from the discrete Fourier extension (2.3) makes the relation with (3.1) apparent. Adopting the notation for the DFT length $L = 2Tm$,

$$
\begin{aligned}
(AA')_{pq} &= \frac{1}{L} \sum_{k=-n}^{n} \exp \mathrm{i} \frac{(p-q)2\pi k}{L}, \qquad p, q = -m, \ldots, m \\
&= (D_M B_N D'_M)_{pq} = (JJ')_{pq},
\end{aligned} \tag{3.4}
$$

where the last line follows from the idempotency of B_N. Consequently, A and J share the same left singular vectors, and the same singular values. Essentially, discrete Fourier extension is a reformulation of the bandlimited extrapolation problem with the N frequency coefficients as unknowns, instead of the extrapolated signal. The focus has also shifted from determining an accurate extension to the approximation of the given data samples.

The next sections concern the PSWFs and P-DPSSs, and how they are natural solutions for the bandlimited extrapolation problem. The groundwork for this theory was detailed in a series of papers by Slepian, Landau and Pollak from 1961 onwards [30, 21, 22, 27, 28]. An overview is given in [29].

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

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

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

$$\mathcal{D}f(x) = \hat{f}(x) = \begin{cases} f(x) & |x| \le T \\ 0 & |x| > T \end{cases} \qquad \mathcal{B}f(x) = \int_{-\Omega}^{\Omega} F(\xi) e^{i2\pi\xi x} d\xi, \tag{3.5}$$

which project onto $\mathcal{L}^2_{[-T,T]}$ and PW_Ω, the Paley-Wiener space of bandlimited functions, respectively. Note that the bandlimiting operator can also be written as

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

7

The Heisenberg-Gabor limit states that no nonzero function can be simultaneously concentrated in both time and frequency, and so $\forall f \neq 0 : ||\mathcal{BD}f|| < ||f||$. However, one can look for nearly-invariant functions under this operator, functions for which $||\mathcal{BD}f||/||f||$ is as close to 1 as possible.

These are the eigenfunctions of the operator $\mathcal{BD}$, i. e. the solutions of the integral equation

$$\lambda\psi(x) = \int_{-T}^{T} \psi(s)\frac{\sin(2\pi\Omega(x-s))}{\pi(x-s)}ds. \tag{3.6}$$

Slepian and collaborators showed that this equation is solvable only for select values of λ, a countably infinite set $1 > \lambda_0 > \lambda_1 > \cdots > 0$. The corresponding eigenfunctions ψ_i were named Prolate Spheroidal Wave functions. The naming stems from the curious observation that these functions are solutions to the spheroidal wave equation

$$\left(1 - \frac{x^2}{T^2}\right)\frac{d^2\psi_i}{dx^2} - 2x\frac{d\psi_i}{dx} - (2\pi\Omega T)^2 x^2\psi_i = \theta_i\psi_i. \tag{3.7}$$

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 ψ_i[30].

Since $\mathcal{B}$ and $\mathcal{D}$ are idempotent operators, it is convenient to consider the ψ_i eigenfunctions of the Hermitian operator $\mathcal{BDB}$. Timelimiting both sides of (3.6), the timelimited functions $\hat{\psi}_i = \mathcal{D}\psi_i$ 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 ψ_i and the $\hat{\psi}_i$.

As eigenfunctions of a Hermitian operator, the ψ_i and $\hat{\psi}_i$ are orthogonal

$$\int_{-\infty}^{\infty} \psi_i(x)\psi_j(x)dx = \delta_{ij} \qquad \int_{-T}^{T} \hat{\psi}_i(x)\hat{\psi}_j(x)dx = \lambda_i\delta_{ij},$$

and they are complete in PW_Ω and $\mathcal{L}^2_{[-T,T]}$, respectively. The Prolate Spheroidal Wave functions thus form an orthogonal base for PW_Ω, and the eigenvalue λ_i represents the fraction of energy of ψ_i contained in $[-T, T]$.

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

$$g = \sum_i \frac{1}{\lambda_i}\langle\hat{\psi}_i, f\rangle_{[-T,T]}\psi_i \tag{3.8}$$

is a bandlimited function that agrees with f in the interval due to the completeness of the $\hat{\psi}_i$ in $\mathcal{L}^2_{[-T,T]}$. Furthermore, when truncating the sum, the first terms have the largest eigenvalues and capture the most energy inside the interval.

Using both (3.6) and (3.7), the PSWFs were further shown to have the following properties [30]:

i The ψ_i are eigenfunctions of the finite Fourier transform,

$$\int_{-T}^{T} e^{i2\pi t\xi}\psi_n(t)dt = i^n\left(\frac{\lambda T}{\Omega}\right)^{1/2}\psi_n\left(\frac{\xi T}{\Omega}\right).$$

ii The eigenvalues λ_i cluster near 1 for low values of i, and decay exponentially after a set breakpoint

$$\lambda_i \approx 1, i \ll 4\Omega T \quad \text{and} \quad \lambda_i \approx 0, i \gg 4\Omega T.$$

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

The first property illustrates the symmetry of time and frequency domain in this problem. The second property shows that a truncated approximation (3.8) will require $\approx 4\Omega T$ terms. The increasing irregularity of the $\hat{\phi}_i$ coupled with the exponential decay of the λ_i ensures that g will converge rapidly after this breakpoint for a sufficiently regular f.

3.3. Periodic Discrete Prolate Spheroidal Sequences. There are several possible discretisations for PSWFs. The most well-known is the one proposed by Slepian, where the Fourier transform is replaced with the discrete-time Fourier transform

$$G(f) = \sum_{n=-\infty}^{\infty} g[n]e^{-i2\pi fn}, \qquad g[n] = \frac{1}{2\pi} \int_{-1/2}^{1/2} G(f)e^{i2\pi fn}df.$$

The resulting generalisations are frequency domain functions on $[-1/2, 1/2]$ commonly referred to as discrete prolate spheroidal wave functions (DPSWFs), and infinite sequences in the time domain called discrete prolate spheroidal sequences (DPSSs). The properties from section 3.2 carry over, apart from some loss of symmetry between time and frequency domains [28].

Another approach was proposed independently by Jain and Ranganath [19] and Grünbaum [17], and elaborated on in [33]. They replace the Fourier transform with the DFT for sequences of length L:

$$H_k = \sum_{n=0}^{L-1} h[n]e^{-i2\pi kn/L} \qquad h[n] = \frac{1}{L} \sum_{k=0}^{L-1} H_k e^{i2\pi kn/L}.$$

The discrete analogues of the time- and bandlimiting operators are given by the matrices (3.2) and (3.3) from the discrete band limited extrapolation problem. Similar to the continuous PSWFs, the periodic discrete prolate spheroidal sequences (P-DPSS) ϕ_i are defined as the eigenvectors of the hermitian matrix operator $B_N D'_M D_M B_N$, and their limited versions $\hat{\phi}_i$ as the eigenvectors of $D_M B_N D_M$

$$B_N D'_M D_M B_N \phi_i = v_i \phi_i \qquad D_M B_N D'_M \hat{\phi}_i = v_i \hat{\phi}_i. \tag{3.9}$$

Since $D_M B_N D'_M$ is not of full rank when $M > N$, an additional requirement that $v_i \neq 0$ is imposed, following [33]. Consequently, there are only $\{\min(N, M)\}$ P-DPSSs.

The ϕ_i properties are similar to those of PSWFs:

i If $M \geq N$, the ϕ_i are complete in the space of bandlimited sequences spanned by B_N.

ii If $M \leq N$, the $\hat{\phi}_i$ are complete in $\mathbb{R}^M$.

iii The ϕ_i are doubly orthogonal

$$\phi_i \cdot \phi_j = \delta_{ij}, \quad D\phi_i \cdot D\phi_j = v_i \delta_{ij}$$

iv The P-DPSSs are eigenvectors of the finite DFT, but with the roles of M and N interchanged

$$D_N F \phi_{M,N,i} = \hat{\phi}_{N,M,i}.$$

Here, F is the DFT matrix of size L.

v Among sequences of length L, with frequency support in $[0, m] \cup [L - m, L]$, ϕ_0 is the most concentrated in $[0, M - 1]$. Among sequences of equally limited frequency support orthogonal to ϕ_0, ϕ_1 is the most concentrated in $[0, M - 1]$, and so on.

vi Like the eigenvalues of the PSWFs, the eigenvalues v_i are distinct and cluster exponentially near 1 and 0, in that

$$v_i \approx 1, i \ll \frac{NM}{L} \quad \text{and} \quad v_i \approx 0, i \gg \frac{NM}{L}.$$

The width of the plunge region where $v_i \in (\epsilon, 1 - \epsilon)$ grows as $\log(NM/L)$ [32].

vii The P-DPSSs satisfy a second order difference equation [33]

$$b_k \hat{\phi}_i[k - 1] + c_k \hat{\phi}_i[k] + b_{k+1} \hat{\phi}_i[k + 1] = \theta_i \hat{\phi}_i[k], \qquad k = 0, \ldots, M - 1, \tag{3.10}$$

with coefficients

$$b_k = \sin\left(\frac{\pi k}{L}\right) \sin\left(\frac{\pi(M - k)}{L}\right)$$

$$c_k = -\cos\left(\frac{\pi(2k + 1 - M)}{L}\right) \cos\left(\frac{\pi N}{L}\right).$$

9

Equation (3.9) clarifies their importance in both the discrete band limited extrapolation problem, and discrete Fourier extensions. From (3.4) it is obvious that the eigenvectors of AA' and JJ' are the limited P-DPSSs $\hat{\phi}_i$. Regarding the singular value decompositions of A and J, this determines the left singular vectors as $\hat{\phi}_i$ and the associated singular values as $\sqrt{v_i}$. For J, following

$$J'J = B_N' D_M' D_M B_N = B_N D_M B_N,$$

the right singular vectors are seen to be full P-DPSSs ϕ_i. Using these relations, Jains SVD method results in an extrapolated sequence

$$g[n] = \sum_{i=0}^{N-1} \frac{1}{\sqrt{v_i}} \left(\hat{\phi}_i \cdot f \right) \phi_i,$$

similar to the continuous extrapolation (3.8).

The discrete Fourier extension matrix A differs from J in the right singular vectors. Similar to (3.4)

$$A'A = \left\{ \sum_{l=-m}^{m} \exp \mathrm{i} \frac{(p-q)2\pi k}{L} \right\}, \qquad p, q = -n, \dots, n$$
$$= D_N B_M D_N.$$

The right singular vectors are then again limited P-DPSSs $\hat{\phi}_i$, where the parameters N and M have been interchanged. Using $\hat{\Phi}$ to represent the set of P-DPSSs and Υ for the diagonal matrix of eigenvalues, the SVD of A is given by

$$A = \hat{\Phi}_{M,N} \sqrt{\Upsilon} \hat{\Phi}_{N,M}'.$$

Calculating the coefficients of the discrete FE by truncating the SVD now corresponds to

$$a = \sum_{i=0}^{i_c} \frac{1}{\sqrt{v_i}} \left(\hat{\phi}_{N,M,i} \cdot f \right) \hat{\phi}_{M,N,i},$$

where i_c is determined by the cutoff parameter,

$$\sqrt{v_{i_c}} \geq \tau > \sqrt{v_{i_c+1}}.$$

This formulation of the FE together with the P-DPSS properties listed above leads to the fast algorithms in the next section.

4. Algorithms. In this section we present two approaches to computing the truncated SVD solution to the discrete Fourier extension problem $Ax = b$. Both approaches are based on a specific division in easier subproblems, which is explained below. They differ only in how they solve one of these subproblems. Detailed algorithms follow in sections 4.1 and 4.2.

From the previous section, the SVD of the FE matrix A is

$$A = U\Sigma V' = \hat{\Phi}_{M,N} \sqrt{\Upsilon} \hat{\Phi}_{N,M}'.$$

The cost of this full SVD is prohibitively large, growing as $\mathcal{O}(N^3)$. To reach the performance goal of $\mathcal{O}(N \log N)$ operations, two subproblems are identified and solved in succession. The key to this division is in the distribution of the singular values Σ, as shown in figure 4.1. Three distinct regions are visible. Following property (vi) of the P-DPSS, they are for some small cutoff τ:

- A region $I_\alpha := \{\sigma : 1 > \sigma > 1 - \tau\}$ where all singular values are 1 up to a tolerance τ. This region contains approximately $\mathcal{O}(NM/L)$ singular values.
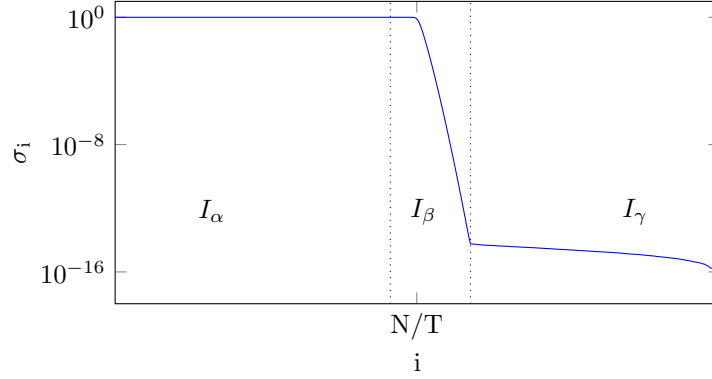
Fig. 4.1: The subdivision of the spectrum of A into three distinct intervals, with cutoff parameter $\tau = 1e - 14$. Due to rounding errors, the eigenvalues in region I_γ don't decay past machine precision.

- 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. The number of singular values in this region grows as $\mathcal{O}(\log{(NM/L)})$.
- A region $I_\gamma := \{\sigma : \tau \geq \sigma > 0\}$ where the singular values further decay exponentially.

Let the subdivision of the singular values and associated vectors be denoted by

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

The truncated SVD solution to the problem $Ax = b$ with truncation cutoff at τ is then

$$x = x_\alpha + x_\beta = V_\alpha \Sigma_\alpha^{-1} U_\alpha' b_\alpha + V_\beta \Sigma_\beta^{-1} U_\beta' b_\beta, \tag{4.1}$$

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.,

$$b_\alpha = U_\alpha U_\alpha' b,$$

and so on. Note that this solution method implicitly assumes b_γ to be below the required solution accuracy. If it is not, a large solution term x_γ is needed, with $\|x_\gamma\| \geq \frac{1}{\tau}\|b_\gamma\|$. In this case, we say the Fourier extension has not yet converged, and N should be increased.

Equation (4.1) splits the problem into two orthogonal subproblems. The isolation of V_β, S_β and U_β from the plunge region and efficient calculation of x_β is where the two approaches given in sections 4.1 and 4.2 differ.

The subsequent calculation of 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 $\Sigma_\alpha = \Sigma_\alpha^{-1} + \mathcal{O}(\tau)$ then yields x_α at the cost of a single fast multiplication with A' ($\mathcal{O}(N \log N)$).

11

4.1. Approach 1: explicit projection onto prolate sequences. Due to the intrinsic connection of the FE problem with DPSSs, it is possible to explicitly compute U_β, Σ_β and V'_β. The second order difference equation from (3.10) implies that

$$Z_{M,N}\hat{\phi}_{M,N,i} = \theta_i\hat{\phi}_{M,N,i}, \qquad Z_{M,N} = \begin{bmatrix} c_0 & b_1 & & \\ b_1 & c_1 & \ddots & \\ & \ddots & \ddots & b_{N-1} \\ & & b_{N-1} & c_{N-1} \end{bmatrix}. \tag{4.2}$$

The P-DPSSs can thus be found as eigenvectors of a tridiagonal matrix. The $\hat{\phi}_{M,N,i}$ that make up U are eigenvectors of $Z_{M,N}$. The dual matrix $Z_{N,M}$ yields the right singular vectors V. With the P-DPSSs known, the original singular value is found in $\mathcal{O}(N\log N)$ operations as

$$\sigma_i = \hat{\phi}'_{m,n,i}A\hat{\phi}_{n,m,i}$$

This approach is already in use for the regular DPSSs [16].

In this stage of the algorithm however, we are only interested in a subset of the singular values and vectors. Since the number of singular values in the plunge region is logarithmically small, we look to algorithms that calculate k eigenvalues and eigenvectors of a tridiagonal matrix Z in $\mathcal{O}(kN)$ operations [11]. 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 all $\theta : C_1 \leq \theta \leq C_2$.
- A index set $\{i_{\min}, \ldots, i_{\max}\}$: the algorithm then finds all $\theta : \theta_{i_{\min}} \leq \theta \leq \theta_{i_{\max}}$, 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 $\sigma_i \in I_\beta$. Denote the mappings between σ_i and θ_j, and between their indices i and j, by

$$\theta_j = \mathcal{G}_{N,M}(\sigma_i), \quad j = \mathcal{G}'_{N,M}(i) \Leftrightarrow \begin{cases} \hat{\phi}'_i Z_{N,M}\hat{\phi}_i = \theta_j \\ \hat{\phi}'_i A_{N,M}\hat{\phi}_i = \sigma_i. \end{cases}$$

The description of $\mathcal{G}_{N,M}(I_\beta)$ is difficult, as very little is known of this mapping. The only known trait is monotonicity, a trait shared with the PSWF and DPSWF equivalents. This is already very helpful, since it implies that $\mathcal{G}'_{N,M}(i) = i$.

THEOREM 1. *For $M \geq N$, the mapping $\mathcal{G}'_{N,M}$ is monotone,*

$$\forall i_1, i_2 : i_1 > i_2 \Leftrightarrow \mathcal{G}'_{N,M}(i_1) > \mathcal{G}'_{N,M}(i_2).$$

Proof. The proof follows a mechanism used by Slepian in both [30, p. 61] and [28, §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 $\bar{A}$ be the matrix of the continuous FE (2.5) with eigenvalues $\bar{\sigma}_i$, and $\bar{Z}_N$ the tridiagonal matrix (4.3) with eigenvalues $\bar{\theta}_i$

$$\bar{Z}_N = \begin{bmatrix} p\bar{c}_0 & \bar{b}_1 & & \\ \bar{b}_1 & \bar{c}_1 & \ddots & \\ & \ddots & \ddots & \bar{b}_{N-1} \\ & & \bar{b}_{N-1} & \bar{c}_{N-1} \end{bmatrix}, \qquad \bar{c}_i = \left(\frac{N-1}{2} - i\right)^2 \cos\frac{\pi}{T}, \qquad \bar{b}_i = \frac{i(N-i)}{2}. \tag{4.3}$$

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

$$j = \bar{\mathcal{G}}'(i) \Leftrightarrow \begin{cases} \bar{\phi}_i' \bar{Z}_N \bar{\phi}_i = \bar{\theta}_j \\ \bar{\phi}_i' \bar{A} \bar{\phi}_i = \bar{\sigma}_i \end{cases},$$

was proven to be monotone in [30, §4.1]. This result can be extended to the discrete FE case, since the limits of the entries of $A'A$ and $Z_{N,M}$ for large m can be written in terms of the corresponding matrices of the continuous problem:

$$\lim_{m \to \infty} (A'A)_{ij} = \frac{\sin \frac{(i-j)\pi}{T}}{\pi(i-j)} = \bar{A}_{ij}$$

$$\lim_{m \to \infty} b_k = \frac{\pi^2 k(N-k)}{L^2} = \frac{2\pi^2}{L^2} \bar{b}_k$$

$$\lim_{m \to \infty} c_k = \cos \frac{\pi}{T} \left(\frac{2\pi^2}{L^2} \left(\frac{N-1}{2} - k \right)^2 - 1 \right) = \frac{2\pi^2}{L^2} \bar{c}_k - \cos \frac{\pi}{T}$$

$$\lim_{m \to \infty} Z_{M,N} = \frac{2\pi^2}{L^2} \bar{Z}_N - \cos \frac{\pi}{T} I$$

The last line ensures that the eigenvalues of $Z_{M,N}$ are those of $\bar{Z}_N$ under a linear mapping. Since this mapping preserves the ordering of eigenvalues, the theorem holds in the limit $m \to \infty$. Further, when limited to the regime $T > 1, 2m + 1 \geq N$, we have the following:

1. The tridiagonal matrix $Z_{M,N}(m)$ with diagonal elements c_k and subdiagonal elements b_k

$$b_k = \sin \left(\frac{\pi k}{2Tm} \right) \sin \left(\frac{\pi(2m+1-k)}{2Tm} \right)$$

$$c_k = -\cos \left(\frac{\pi(k-m)}{Tm} \right) \cos \left(\frac{\pi N}{2Tm} \right).$$

   commutes with $A'A$ for integer values of m. However, by a substitution as in [10, Thm. 4.7], it is easy to see that this relationship holds for any real $m > n$.

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

$$\forall 1 \leq k \leq N - 1 : b_k^2 > 0,$$

   $Z_{M,N}(m)$ is a so-called normal Jacobi matrix and such matrices are known to have distinct eigenvalues [13, Ch. 2.1]. As a result of this distinctness, the eigenvectors can be chosen to be continuous in m as well [20, Ch. 2 §5.3].

3. To prove similar statements for the matrix $A'A$, we note that the inductive proof in [33, Prop. 5] for the distinctness of eigenvalues of $A'A$ is dependent only on it being symmetric, and the commuting $Z_{M,N}(m)$ being a normal Jacobi matrix. The eigenvalues $\lambda_i(m)$ are thus continuous and distinct $\forall m \geq n$. Then the eigenvectors can also be chosen continuous in m.

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

The required index set for the eigenvalues of $Z_{M,N}$ is exactly the index set corresponding to σ_i from the plunge region. From (vi) these are known to be centered around $\frac{NM}{L}$, and their number grows as $\mathcal{O}(\log N)$. All that remains is to determine the constants $C_{\min}$ and $C_{\max}$ so that

$$\sigma_j \in I_\beta \Leftrightarrow j \in \left[\max \left\{ N - \frac{NM}{L} - C_{\min} \log N, 1 \right\}, \min \left\{ N - \frac{NM}{L} + C_{\max} \log N, N \right\} \right].$$

13

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 is shown in Figure 4.2, with the value $N - \frac{NM}{L}$ as a dashed line. Experimentally, the choices $C_{\min} \geq 6$ and $C_{\max} \geq 3$ seem sufficient for all $\tau \geq \epsilon_{mach}$. The V_β can be obtained by refining $A'\hat{\phi}_{M,N,i}$ as eigenvectors of $Z_{N,M}$.

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

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

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

REMARK 2. *Monotonicity of the map $\mathcal{G}_{N,M}$ is also observed to hold for integer M smaller than integer N. Specifically, a variant of Theorem 1 holds for the M nonzero singular values σ_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 altered algorithm is without proof.*
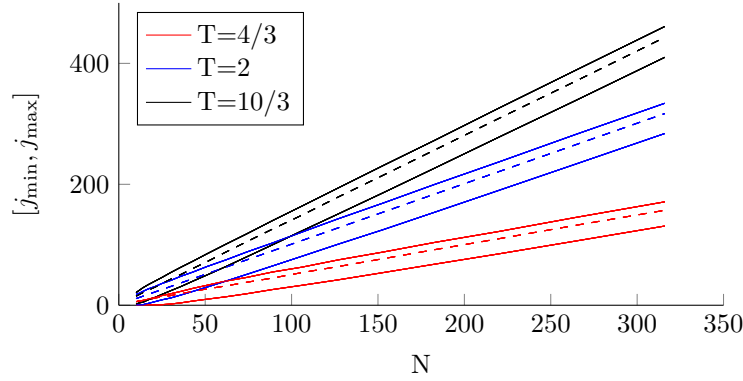

Fig. 4.2: The behaviour of the index set of the plunge region. The minimal and maximal index of the plunge region are shown as solid lines, for different values of T. The point NM/L, which is known to lie in the interval, is shown as a dashed line.

Combining x_β with the calculation of x_α described above leads to a fast $\mathcal{O}(N \log^2 N)$ algorithm. A bare-bones version incorporating remark 2 is given in Algorithm 1. The *trideig* function should return eigenvectors of a tridiagonal matrix given an index interval, e.g. the LAPACK routine *dstevx*.

The matrix-vector multiplications with A and A' should be implemented using a combination of length L unitary FFTs F and F' and selection matrices D_M and D_N:

$$A = D_M F' D_N' \qquad A' = D_N F D_M'.$$

Here, D_N selects the subset of frequencies that correspond to the Fourier series approximation from the longer FFT, i.e., the $2N + 1$ smallest frequencies. The matrix D_M selects the function samples that lie in the approximation interval from the equispaced samples on the extended interval. We have used a symmetric embedding $[-1, 1] \in [-T, T]$ in this paper, but other choices are possible. We refer to the MATLAB implementation of these matrix-vector products in the appendix for the details of the implementation.

4.2. Approach 2: an implicit projection method. The second approach to calculating x_β is more universal, or more generally applicable, since it depends solely on the steep singular value profile discussed earlier and illustrated in Fig. 4.1. As such, it is extensible to any ill-conditioned basis that exhibits a similar profile.

Algorithm 1 Explicit projection on DPSSs

$U_\beta = \mathrm{trideig}(Z_{M,N}, \{j_{\min}, j_{\max}\})$
$V_\beta = \mathrm{trideig}(Z_{N,M}, \{j_{\min}, j_{\max}\})$
$\Sigma_\beta = (U_\beta' A V_\beta) > \tau$
$x_\beta = V_\beta \Sigma_\beta^{-1} U_\beta' b$
$x_\alpha = A'(b - Ax_\beta)$
$x = x_\alpha + x_\beta$

The approach is based on the observation that multiplying both the FE matrix A and right hand side by a factor

$$P = (AA' - I) \tag{4.4}$$

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

$$P = U(\Sigma^2 - I)U', \tag{4.5}$$

and

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

Note here that the mapping

$$\mathcal{P}(\sigma) = \sigma^3 - \sigma$$

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

$$PA = U_\beta(\Sigma_\beta^3 - \Sigma_\beta)V_\beta' + \mathcal{O}(\tau). \tag{4.6}$$

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

$$PAx = Pb \tag{4.7}$$

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 to solve (4.7) very efficiently.

Assume W a uniform random matrix of dimensions $N \times R$, where $R = C \log N + D$ is a conservative estimate for the rank of PA. From the previous section, $C = C_{\min} + C_{\max} \geq 9$ is sufficient, with $D \sim 10$ minimizing the chance of failure of the random matrix algorithm. The column space of PAW then approximates the column space of PA very well.

Therefore, solving the following small linear system

$$PAWy = Pb$$

and letting

$$x_W = Wy$$

one obtains a solution to (4.7) a cost of $\mathcal{O}(MR^2)$. It follows from (4.5) and (4.6) that x_β is recovered exactly. On the other hand, this solution process introduces additional solution terms from the nullspace of PA. Write x_W as $x_\beta + r_\alpha + r_\gamma$. Then as before we calculate

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

Then $x = x_W + A'(b - Ax_W)$ boils down to

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

so that $||Ax - b|| = \mathcal{O}(\tau)$. The total cost of this algorithm is again $\mathcal{O}(N \log^2 N)$ operations. A step by step version is given in Algorithm 2. To illustrate the simplicity of the algorithm, a MATLAB implementation is provided in Appendix, where the main computations comprise fewer than 10 lines of code.

Algorithm 2 Implicit projection on DPSSs

$PAWy = Pb$
$x_\beta = Wy$
$x_\alpha = A'(b - Ax_\beta)$
$x = x_\alpha + x_\beta$

4.3. Adaptation for the continuous FE. As mentioned in section 3.3, the continuous FE also has numerous connections with Prolate Spheroidal Wave theory. The solution is most easily expressed in terms of the eigenvectors of $\bar{A}$, the DPSSs

$$\bar{A} = \Psi\Lambda\Psi.$$

The eigenvalues λ_i have a similar profile to that of Figure 4.1. Our second approach thus applies immediately, and Algorithm 2 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 (3.10). In particular, the matrix $\bar{A}$ commutes with the tridiagonal matrix (4.3).

It follows that, with minor modifications, Algorithm 1 can also be used to solve the continuous FE problem.

5. Numerical Results. In this section we apply the algorithms from the previous section to a number of test problems. All tests were performed in MATLAB , single threaded. 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 [3], the product $L = 2T\gamma n$ is held constant when varying T in order to maintain a fixed condition number. This means the oversampling $\gamma = M/N$ varies between experiments. For our values of T we use

$$M_1 \approx \frac{4}{1.1}N, \quad M_2, = 2N \quad M_3 \approx \frac{4}{3.8}N,$$

rounded to odd M.

5.1. Computational complexity. Figure 5.1a shows execution time for increasing degrees of freedom of the algorithm for different values of T. The figure confirms the theorized $\mathcal{O}(N \log^2 N)$ asymptotic complexity of our algorithms. It also shows execution speed is on par with the current fast algorithm by Lyon. Also, as in Lyons algorithm, the majority of the work is in computing a low-rank matrix decomposition related to A, in this case the middle part of the SVD. This can provide a significant speedup when multiple approximations are needed with the same parameters.

Figure 5.1b compares our two approaches in more detail. The figure shows the execution time per degree of freedom. The difference between algorithms is very much dependent on implementation, but

16

it highlights a trend seen for different values of T. Both algorithms appear to be slightly faster for larger T, which is somewhat counterintuitive. After all, the cost of the FFT operations is $\mathcal{O}(L)$, which is held constant for all experiments. However, both approaches solve a subproblem with a cost that grows as $\mathcal{O}(M)$. This is either the tridiagonal eigenvalue problem with a matrix of size $M \times M$, or a linear solve of an $M \times \log M$ system. Due to our scaling of M, these costs actually decrease with T.

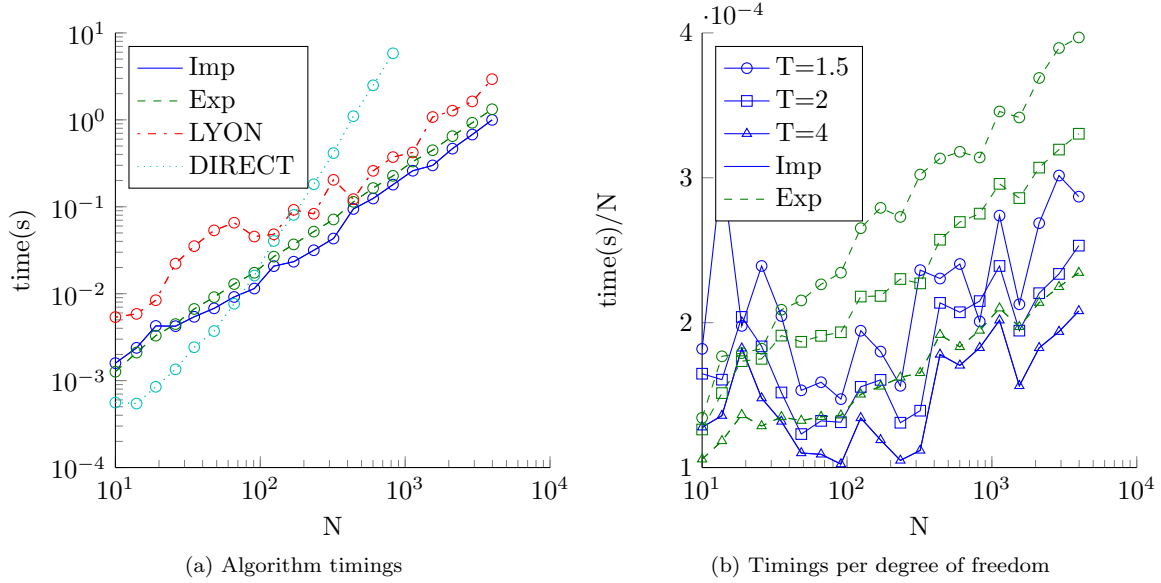

(a) Algorithm timings

(b) Timings per degree of freedom

Fig. 5.1: Execution time for increasing degrees of freedom N, for MATLAB implementations of the Explicit and Implicit projection algorithms, the Lyon algorithm and a direct solver.

5.2. Convergence. The accuracy of the solution obtained through our algorithms is shown in Figures 5.2a through 5.2d. The accuracy is measured as the maximum pointwise error over an equisampled grid ten times denser than the one used for construction. This is measured for increasing number of degrees of freedom N for four test functions:

- A well-behaved, smooth function to show convergence in near optimal conditions,

$$f_1(x) = x^2.$$

- A highly oscillatory function, to show the resolution power of Fourier extensions for oscillatory functions,

$$f_2(x) = \text{Ai}(76x).$$

- A function with a pole in the complex plane near the real interval $[-1, 1]$, to show convergence at a slower, but still exponential rate,

$$f_3(x) = \frac{1}{1.1 - x^2}.$$

- A function with discontinuous first derivative, to see the breakdown of the algorithm to algebraic convergence speeds,

$$f_4(x) = |x|.$$

17

The convergence behaviour seen is in accordance with [18], [2] and [3]. For $T = 2$, it also very closely agrees with the Lyon algorithm and the direct solver. Convergence for functions analytic in $[-1, 1]$ is at least geometric, even when singularities are present near the real interval. Following the earlier arguments about resolution power from §2.3, Fourier extensions of oscillatory functions start to converge sooner for lower values of T. Note that this is in terms of degrees of freedom, and that we increased the oversampling for lower T to maintain conditioning.

When the approximant is in C^k, having continuous derivatives up to $f^{(k)}$, the convergence rate becomes algebraic at a rate of $\mathcal{O}(N^{-k+1})$. The convergence of order $\mathcal{O}(N^{-1})$ for the C^0 function is apparent from Figure 5.2d.

(a) $f_1(x) = x^2$

(b) $f_2(x) = \text{Ai}(76x)$
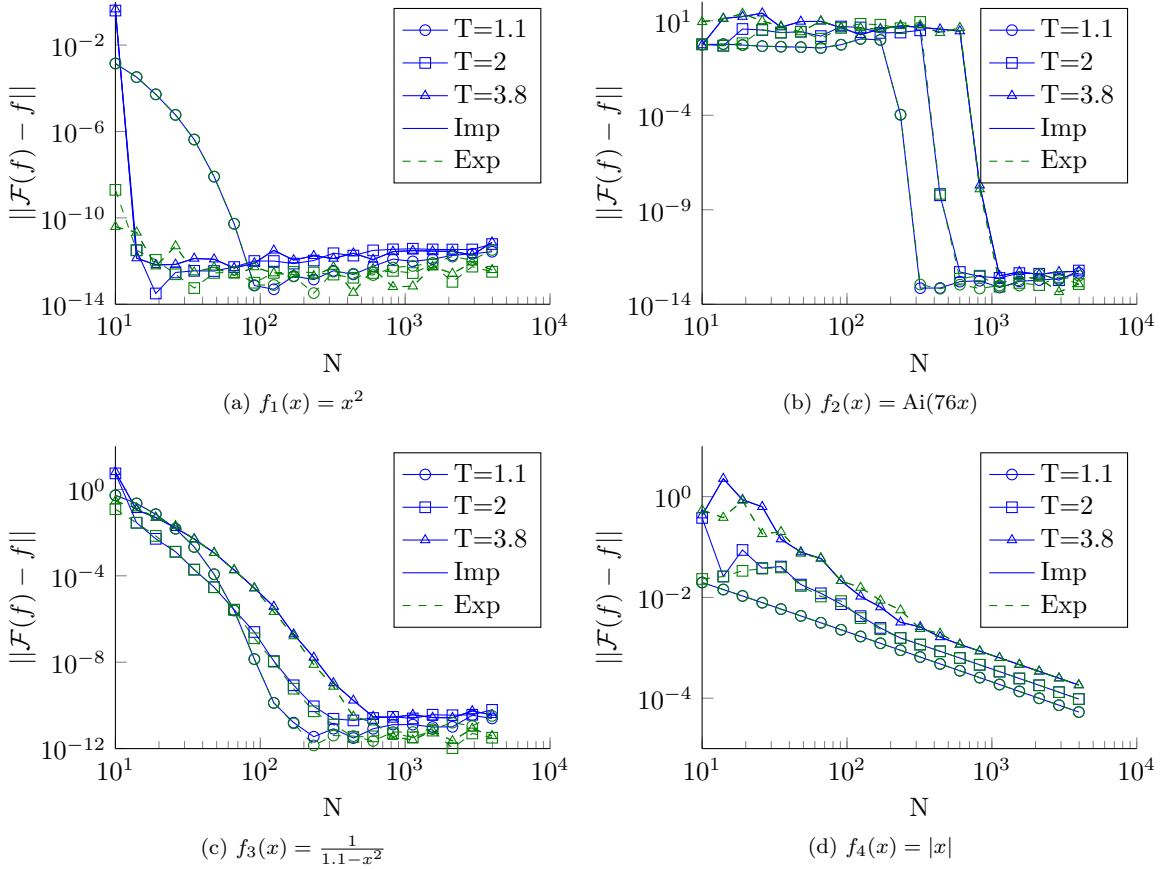
(c) $f_3(x) = \frac{1}{1.1-x^2}$

(d) $f_4(x) = |x|$

Fig. 5.2: The L_∞ norm of the error, computed by oversampling the solution by a factor 10, for the various testfunctions

5.3. Robustness. To ensure there is no accumulation of error for very large N, Figure 5.3a shows the continuation of an accuracy experiment for N up to 10^5. No error accumulation is visible, the error fluctuates around the cutoff threshold.

Figure 5.3b shows the challenging problem of approximating an oscillatory function with a frequency that increases with N, $f(x) = \sin(Nx)$. The error for the Algorithm using $T = 1.1$ and $T = 2$ stays close to machine precision, if maybe slightly increasing. However, for $T = 3.8$, the maximum frequency in the Fourier basis is lower than the frequency of the signal for every N, so there cannot be any convergence. This experiment shows that even a very oscillatory signal such as $f(x) = \sin(10^5 x)$ can be accurately approximated, as long as there are sufficient degrees of freedom for the chosen value
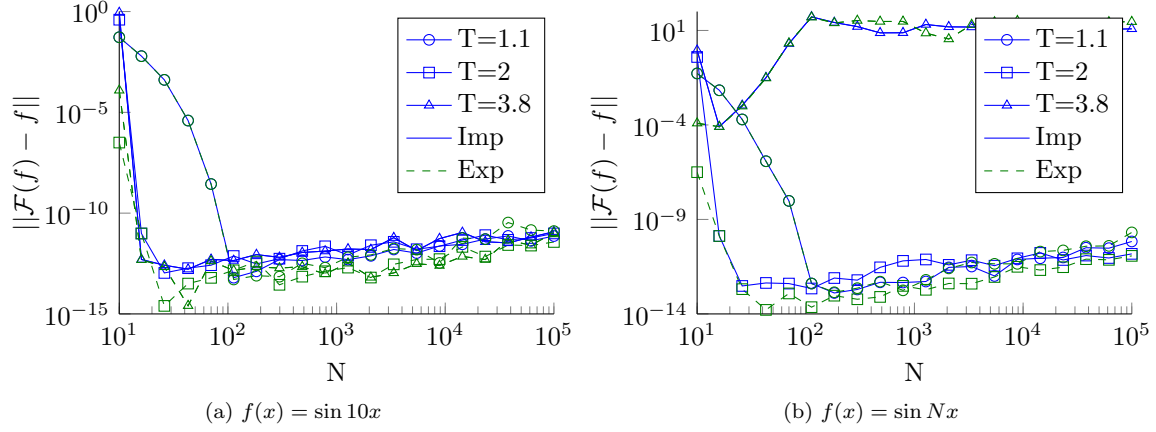
of T.

Fig. 5.3: Illustration of the robustness of FE approximations for large N.

6. Conclusion. In this paper, we have studied the relation between Fourier extensions and a problem from signal processing called bandlimited extrapolation. The solutions to both of these problems can be written in terms of Prolate Spheroidal Wave functions, or any of their discrete variants. Through this connection, the peculiar properties of these Prolate Spheroidal Wave functions can be used as a foundation for fast algorithms to compute Fourier extensions.

Two algorithms were presented, one based on the property that discretised Prolate functions satisfy a second order difference equation, one based on the property that these functions have a particular distribution of singular values. Of these two algorithms, the first has a slight advantage in speed and accuracy. However, the second algorithm is only dependent on the singular value distribution, making it more robust to generalizations. It is also quicker and more straightforward to implement.

The merits of these algorithms lie in facilitating the use of schemes with flexible extension length, whether this is dictated by the application, or gives optimal results for the end-user. Furthermore, the solution methods are functionally equivalent to the truncated SVD method, so previous theory on that solution method still applies. Lastly, the connection with Prolate Spheroidal Wave Theory and the associated projection method paves the way for a number of possible generalizations.

MATLAB code 1 A full matlab implementation of the implicit algorithm.

```matlab
function sol = fastfe(f,n,T,m)
    % Constants
    b=f((-m:m)/m)';
    L=round(T*2*m);
    C=min(ceil(8*log(2*n+1))+10,2*n);

    % Fast matrix operations
    fA=@(x) fast_mv_A(x,n,L,m);
    fAT=@(x) fast_mv_A_transpose(x,n,L,m);
    P=@(y) fA(fAT(y))-y;

    % Isolating the middle singular values
    R=2*rand(2*n+1,C)-1+2i*rand(2*n+1,C)-1i;
    [U,S,V]=svd(P(fA(R)),0);
    pinvS=diag(1./diag(S));
    y=V*(pinvS*(U'*(P(b))));
    x1=R*y;

    % Remaining computation
    b1=fA(x1);
    b2=b-b1;
    x2=fAT(b2);
    sol=(x1+x2)*sqrt(1/m);
end

% Fast matrix vector product
function z = fast_mv_A(y, n, L, m)
    w = [y(n+1:2*n+1,:); zeros(L-(2*n+1),size(y,2)); y(1:n,:)];
    v = L*ifft(w);
    z = 1/sqrt(L)*v([end-m+1:end 1:m+1],:);
end

% Fast matrix vector product (transpose)
function z = fast_mv_A_transpose(y, n, L, m)
    w = [y(m+1:2*m+1,:); zeros(L-(2*m+1),size(y,2)); y(1:m,:)];
    v = fft(w);
    z = 1/sqrt(L)*v([end-n+1:end 1:n+1],:);
end
```

REFERENCES

[1] B. Adcock and D. Huybrechs. On the resolution power of Fourier extensions for oscillatory functions. *J. Comput. Appl. Math.*, 260:312–336, 2014.

[2] B. Adcock, D. Huybrechs, and J. Martín-Vaquero. On the numerical stability of Fourier extensions. *Found. Comp. Math.*, 14(4):635–687, 2014.

[3] B. Adcock and J. Ruan. Parameter selection and numerical approximation properties of fourier extensions from fixed data. *J. Comput. Phys.*, 273:453–471, 2014.

[4] N. Albin and O. P. Bruno. A spectral FC solver for the compressible Navier–Stokes equations in general domains I: Explicit time-stepping. *J. Comput. Phys.*, 230(16):6248–6270, 2011.

[5] J. P. Boyd. A comparison of numerical algorithms for Fourier extension of the first, second, and third kinds. *J. Comput. Phys.*, 178(1):118–160, 2002.

[6] J. P. Boyd. 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):591–597, 2005.

[7] O. Bruno. Fast, high-order, high-frequency integral methods for computational acoustics and electromagnetics. In *Topics in Computational Wave Propagation*, volume 31, pages 43–82. Springer, Berlin, 2003.

[8] O. P. Bruno, Y. Han, and M. M. Pohlman. Accurate, high-order representation of complex three-dimensional surfaces via Fourier continuation analysis. *J. Comput. Phys.*, 227(2):1094–1125, 2007.

[9] O. P. Bruno and M. Lyon. High-order unconditionally stable FC-AD solvers for general smooth domains I. Basic elements. *J. Comput. Phys.*, 229(6):2009–2033, 2010.

[10] C. Chamzas. *On the extrapolation of band-limited signals*. PhD thesis, Polytechnic Institute of New York, 1980.

[11] I. S. Dhillon. *A New Algorithm for the Symmetric Tridiagonal Eigenvalue Eigenvector Problem*. PhD thesis, University of California, Berkeley, 1997.

[12] R. J. Duffin and A. C. Schaeffer. A class of nonharmonic Fourier series. *Trans. Amer. Math. Soc.*, pages 341–366, 1952.

[13] F. R. Gantmakher and M. G. Kreĭn. *Oscillation matrices and kernels and small vibrations of mechanical systems*, volume 345. American Mathematical Soc., 2002.

[14] R. Gerchberg. Super-resolution through error energy reduction. *Optica Acta*, 21(9):709–720, 1974.

[15] D. Gottlieb and S. A. Orszag. *Numerical analysis of spectral methods: theory and applications*, volume 26. Siam, 1977.

[16] D. M. Gruenbacher and D. R. Hummels. A simple algorithm for generating discrete prolate spheroidal sequences. *IEEE Trans. Signal. Process.*, 42(11):3276–3278, 1994.

[17] F. A. Grünbaum. Eigenvectors of a Toeplitz matrix: discrete version of the prolate spheroidal wave functions. *SIAM J. Algebraic Discrete Methods*, 2(2):136–141, 1981.

[18] D. Huybrechs. On the Fourier extension of nonperiodic functions. *SIAM J. Numer. Anal.*, 47(6):4326–4355, 2010.

[19] A. K. Jain and S. Ranganath. Extrapolation algorithms for discrete signals with application in spectral estimation. *IEEE Trans. Acoust. Speech Signal Process.*, 29(4):830–845, 1981.

[20] T. Kato. *Perturbation theory for linear operators*. Springer Science & Business Media, 2012.

[21] H. J. Landau and H. O. Pollak. Prolate spheroidal wave functions, Fourier analysis and uncertainty-II. *AT&T Bell Labs. Tech. J.*, 40(1):65–84, 1961.

[22] H. J. Landau and H. O. Pollak. Prolate Spheroidal Wave Functions, Fourier Analysis and Uncertainty-III: The Dimension of the Space of Essentially Time-and Band-Limited Signals. *AT&T Bell Labs. Tech. J.*, 41(4):1295–1336, 1962.

[23] M. Lyon. A fast algorithm for Fourier continuation. *SIAM J. Sci. Comput.*, 33(6):3241–3260, 2011.

[24] M. Lyon. Approximation error in regularized SVD-based Fourier continuations. *Appl. Numer. Math.*, 62(12):1790–1803, 2012.

[25] M. Lyon and O. P. Bruno. High-order unconditionally stable FC-AD solvers for general smooth domains II. Elliptic, parabolic and hyperbolic PDEs; theoretical considerations. *J. Comput. Phys.*, 229(9):3358–3381, 2010.

[26] A. Papoulis. A new algorithm in spectral analysis and band-limited extrapolation. *IEEE Trans. Circuits and Systems*, 22(9):735–742, 1975.

[27] D. Slepian. Prolate spheroidal wave functions, Fourier analysis and uncertainty-IV: extensions to many dimensions; generalized prolate spheroidal functions. *AT&T Bell Labs. Tech. J.*, 43(6):3009–3057, 1964.

[28] D. Slepian. Prolate spheroidal wave functions, Fourier analysis, and uncertaintyV: The discrete case. *AT&T Bell Labs. Tech. J.*, 57(5):1371–1430, 1978.

[29] D. Slepian. Some comments on Fourier analysis, uncertainty and modeling. *SIAM Rev.*, 25(3):379–393, 1983.

[30] D. Slepian and H. Pollak. Prolate spheroidal wave functions, fourier analysis, and uncertaintyi. *AT&T Bell Labs. Tech. J.*, 40(1):43–63, 1961.

[31] T. Strohmer. On discrete band-limited signal extrapolation. *Contemporary Mathematics*, 190:323–323, 1995.

[32] R. Wilson. Finite prolate spheroidal sequences and their applications I: generation and properties. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(6):787–795, 1987.

[33] W. Y. Xu and C. Chamzas. On the periodic discrete prolate spheroidal sequences. *SIAM J. Appl. Math.*, 44(6):1210–1217, 1984.