Spring 5-16-2024

Improved rational approximation of near-to-far propagation kernels for the wave equation

Sampson Owusu

Sampson Owusu

Candidate

Mathematics and Statistics

Department

This thesis is approved, and it is acceptable in quality
and form for publication:

Approved by the Thesis Committee:

Prof. Stephen Lau, Department of Mathematics and Statistics, Chair

Prof. Jens Lorenz, Department of Mathematics and Statistics

Prof. James Degnan, Department of Mathematics and Statistics

IMPROVED RATIONAL APPROXIMATION OF NEAR-TO-FAR PROPAGATION KERNELS FOR THE WAVE EQUATION

by

SAMPSON OWUSU

BSc Mathematics, Kwame Nkrumah University of Science and Technology, 2020

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Mathematics

The University of New Mexico

Albuquerque, New Mexico

May, 2024

Dedication

Being polite or reasonable is the key to success for all mathematicians. Everyone has a strong interest in describing and understanding the world since it is so full of complex occurrences but also has such breathtaking beauty and grace. We have made constant attempts to establish striking connections between occurrences in order to understand both the interactions that we can see and the ones that are not visible to us since we have an inherent peculiarity to discover or understand something. From the smallest particle to the big picture, we have accomplished remarkable technological feats that have at least given us a peek at the reality of the inner workings of the universe. These achievements have all been made possible by special logical dialects referred to as Mathematics. As a result, we are compelled to accept the existence of a supernatural force that operates independently of us and outside the realm of reason and understanding. Inborn belief in supernatural forces that defy human comprehension permeates all sentient beings. They are required by conscience to have this viewpoint.

Acknowledgments

This work is dedicated to the Almighty God for seeing me through from the start of the project to the end. In light of the foregoing, I would like to express my sincere gratitude to Prof. Stephen Lau, my Advisor, for his unwavering encouragement, admirable forbearance, and guidance that have enabled me to reach this point. I would also like to thank the rest of the UNM Department of Mathematics and Statistics faculty and staff, particularly Ana Parra Lombard, for their tremendous support during my time in the Department. Finally, I want to express my gratitude to my family and friends for their encouragement. I acknowledge the reader too for taking the time to read through the thesis. May God bless you all.

IMPROVED RATIONAL APPROXIMATION OF NEAR-TO-FAR PROPAGATION KERNELS FOR THE WAVE EQUATION

by

SAMPSON OWUSU

BSc Mathematics, Kwame Nkrumah University of Science and Technology, 2020

M.S., Mathematics, University of New Mexico, 2023

Abstract

The 3-space, 1-time dimensional scalar wave equation, or 3+1 wave equation, describes the propagation of scalar or acoustic waves. The unforced homogeneous equation admits a class of outgoing solutions relative to a chosen fixed center, so called “multipole” solutions. This thesis examines near-to-far signal propagation in the context of these multipole solutions. Given a time-series (history of values) for a multipole solution recorded at a radius r_1, near-to-far signal propagation recovers the corresponding time-series at larger radius $r_2 \gg r_1$. This propagation takes into account both the appropriate time delay $r_2 - r_1$ and corrections to the wave shape. Mathematically, the propagation is described by a Laplace convolution involving the time-series at r_1 and a “kernel” which is a weighted sum of time-dependent exponential functions. This thesis studies the Alpert-Greengard-Hagstrom [5] algorithm for rational approximation. It applies the algorithm, with some improvements not considered in [4], to approximation of a near-to-far kernel for $l = 64$ (a moderately large l). As an exercise, it also considers approximation of the scaled Bessel function $J_1(t)/t$. Precisely, the thesis considers approximation of its Laplace transform as a rational function.

TABLE OF CONTENTS

Chapter 1

Introduction

Wave phenomena appear in many guises in nature. Perhaps the characteristic feature of waves is that they can propagate long distances without dispersal. Tsunamis, giant water waves created by undersea seismic events, move great distances across oceans. Long-distance propagation of electromagnetic waves supports modern communication. Gravitational waves generated by astrophysical events propagate many light-years across space. Mathematical description and approximation of the long-distance propagation of waves is therefore of considerable applied interest.

This work studies sphere-to-sphere propagation [1, 2, 6, 3, 4] of wave data for the ordinary three-space dimensional wave equation

$$\left(-\partial_t^2 + \partial_x^2 + \partial_y^2 + \partial_z^2\right) \psi(t, x, y, z) = 0.$$

To describe sphere-to-sphere propagation, we introduce spherical polar coordinates

$$x = r \sin\theta \sin\phi, \qquad y = r \sin\theta \cos\phi, \qquad z = r \cos\theta,$$

where r is the radius, θ is the polar angle, and ϕ is the azimuthal angle. Given a particular spherical harmonic [7] angular function $Y_{\ell m}(\theta, \phi)$, we define the "radial mode"

$$\Psi_{\ell m}(t, r) = r \int_0^{2\pi} \int_0^{\pi} \bar{Y}_{\ell m}(\theta, \phi) \psi(t, r \sin\theta \sin\phi, r \sin\theta \cos\phi, r \cos\theta) \sin\theta d\theta d\phi.$$

Consider two radial values $r_1 < r_2$. The references above examine recovery of the time-series solution, or "signal", $\Psi_{\ell m}\left(t, r_2\right)$ from the time-series $\Psi_{\ell m}\left(t, r_1\right)$. Since in practice we assume

$r_2 \gg r_1$, such recovery is also referred to as near-to-far (NtF) signal propagation. It is known (and we review the derivation below) that

$$\Psi_{\ell m}\left(t+\left(r_2-r_1\right), r_2\right)=\int_0^t \Phi_{\ell}\left(t-t^{\prime}, r_1, r_2\right) \Psi_{\ell m}\left(t^{\prime}, r_1\right) d t^{\prime}+\Psi_{\ell m}\left(t, r_1\right),$$

where the NtF propagation kernel

$$\Phi_{\ell}\left(t, r_1, r_2\right)=\sum_{k=1}^{\ell} a_{\ell k}\left(r_1, r_2\right) e^{b_{\ell k} t / r_1}. \tag{1.1}$$

The numbers $a_{\ell k}\left(r_1, r_2\right)$ and $b_{\ell k}$ are complex and lie in the left-half plane. Since the NtF kernel is a sum of ℓ exponential terms, its Laplace transform

$$\widehat{\Phi}_{\ell}\left(s, r_1, r_2\right)=\sum_{k=1}^{\ell} \frac{a_{\ell k}\left(r_1, r_2\right)}{s-b_{\ell k} / r_1}. \tag{1.2}$$

is a sum of ℓ (simple) poles.

A key problem with the representations of $a_{\ell k}\left(r_1, r_2\right)$ in (1.49), first clearly identified in [3], is that in modulus the residues become exponentially large as ℓ grows, this is $\left|a_{\ell k}\left(r_1, r_2\right)\right| \propto e^{C \ell}$. For large ℓ the residues in modulus

$$\left\{\left|a_{\ell k}\left(r_1, r_2\right)\right|: k=1, \cdots, \ell\right\},$$

vary in size over many orders of magnitude. Thus, if the numbers $a_{\ell k}\left(r_1, r_2\right)$ are approximated as double precision numbers, then the above sums are plagued by inaccuracy due to finite-precision effects. To some extent, this problem can be mitigated through quadruple precision representation, but as ℓ increases eventually inaccuracies will also arise even with extended precision.

The purpose of this thesis is to examine the above kernels for large ℓ, with the sole focus on $\ell=64$ as a key example. It studies approximation of $\widehat{\Phi}_{64}\left(s, r_1, r_2\right)$ *as a smaller pole sum*

$$\widehat{\Phi}_{64}\left(s, r_1, r_2\right) \simeq \sum_{k=1}^{d} \frac{\alpha_k}{s-\beta_k},$$

where $d \leq 64$. Due to finite-precision effects, these smaller poles sums can, perhaps paradoxically, yield more accurate representations of $\widehat{\Phi}_{64}\left(s, r_1, r_2\right)$ than is achievable by a full 64-term pole sum. One goal here is to understand how accurately $\hat{\Phi}_{64}\left(s, r_1, r_2\right)$ can be represented as a sum of d-poles, assuming that the d residues $\{\alpha_k\}_{k=1}^{d}$ and locations $\{\beta_k\}_{k=1}^{d}$ are stored as double precision numbers. Note, however, that the computer code which produce the residues and locations utilizes quadruple precision (with subsequent truncation of the results to double precision representation).

The outline of this thesis is as follows. Chapter 1 deals with the derivation of the radial wave equation (RWE) from the $3+1$ wave equation in spherical polar coordinates that supports our demonstration of multipole solutions of the RWE. Using the Laplace transform it examines the structure of near-to-far propagation kernels. Chapter 2 considers the approximation of functions, using the theoretical results and some implementations (numerical algorithms). It considers both polynomial and proper rational approximation. Chapter 3 focuses on numerical examples. A final section summarizes our results and conclusions.

Chapter 2

Near-to-far signal propagation for the wave equation

2.1 Multipole solutions for the radial wave equation

This subsection gives a quick derivation of radial wave equation (RWE) from the $3+1$ wave equation in spherical polar coordinates, and proves a multipole expansion formula for solutions to the RWE. Part of this background chapter parallel the results given in [8].

Let us consider the $3-$space, $1-$time dimensional wave equation given by:

$$\frac{\partial^2\psi}{\partial x^2}+\frac{\partial^2\psi}{\partial y^2}+\frac{\partial^2\psi}{\partial z^2}-\frac{\partial^2\psi}{\partial t^2}=0. \tag{2.1}$$

The Laplacian has the following form in spherical polar coordinates:

$$\nabla^2\psi=\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial\psi}{\partial r}\right)+\frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial\psi}{\partial\theta}\right)+\frac{1}{r^2\sin^2\theta}\frac{\partial^2\psi}{\partial\phi^2}, \tag{2.2}$$

where θ is the polar and ϕ is the azimuthal angle. We also write

$$\nabla^2\psi=\frac{\partial^2\psi}{\partial r^2}+\frac{2}{r}\frac{\partial\psi}{\partial r}+\frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left[\sin\theta\frac{\partial\psi}{\partial\theta}\right]+\frac{1}{r^2\sin^2\theta}\frac{\partial^2\psi}{\partial\phi^2}. \tag{2.3}$$

Therefore, the wave equation expressed in spherical polar coordinates is:

$$\frac{\partial^2\psi}{\partial t^2}=\frac{\partial^2\psi}{\partial r^2}+\frac{2}{r}\frac{\partial\psi}{\partial r}+\frac{1}{r^2}\left[\frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial\psi}{\partial\theta}\right)+\frac{1}{\sin^2\theta}\frac{\partial^2\psi}{\partial\phi^2}\right]. \tag{2.4}$$

Let

$$\left(\frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\sin\theta\frac{\partial}{\partial\theta}+\frac{1}{\sin^2\theta}\frac{\partial^2}{\partial\phi^2}\right)=A, \tag{2.5}$$

where A is the Laplace operator on the unit-radius sphere. A as defined has negative eigenvalues. Then, we have:

$$\frac{\partial^2 \psi}{\partial t^2} = \frac{\partial^2 \psi}{\partial r^2} + \frac{2}{r}\frac{\partial \psi}{\partial r} + \frac{1}{r^2} A\psi. \tag{2.6}$$

We assume a multipole expansion for the wave field of the form:

$$\psi(t,x,y,z) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l} \frac{1}{r}\Psi_{lm}(t,r) Y_{lm}(\theta,\phi), \tag{2.7}$$

where $x = r\sin\theta\cos\phi, y = r\sin\theta\sin\phi, z = r\cos\theta$ and

$$Y_{lm}(\theta,\phi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}} P_l^m(\cos\theta) e^{im\phi} \tag{2.8}$$

is the standard spherical harmonic, with $P_l^m(\cos\theta)$ the associated Legendre function. This is a formula in [7]. The eigenfunctions $Y_{lm}(\theta,\phi)$ obey the identity:

$$AY_{lm}(\theta,\phi) = -l(l+1)Y_{lm}(\theta,\phi). \tag{2.9}$$

It is sufficient to examine a single mode to derive the radial wave equation (RWE). Therefore, set $\psi(t,x,y,z) = \frac{\Psi(t,r)}{r} Y_{lm}(\theta,\phi)$, and substitute (2.9) into formula from (2.4), then we have:

$$\frac{1}{r}\frac{\partial^2 \Psi}{\partial t^2} = \frac{1}{r^2}\frac{\partial}{\partial r} r^2 \frac{\partial}{\partial r}\left(\frac{\Psi}{r}\right) - \frac{\ell(\ell+1)}{r^2}\frac{\Psi}{r}. \tag{2.10}$$

We also have:

$$\begin{aligned} \frac{1}{r}\frac{\partial}{\partial r} r^2 \frac{\partial}{\partial r}\left(\frac{\Psi}{r}\right) =& \frac{1}{r}\frac{\partial}{\partial r}(r\Psi' - \Psi) \\ =& \Psi'' + \frac{\Psi'}{r} - \frac{\Psi'}{r} = \Psi'', \end{aligned}$$

and therefore,

$$\frac{\partial^2 \Psi}{\partial t^2} = \frac{\partial^2 \Psi}{\partial r^2} - \frac{l(l+1)}{r^2}\Psi. \tag{2.11}$$

This is known as the radial wave equation, and it has an "effective potential", i.e.

$$V(r) = \frac{l(l+1)}{r^2}. \tag{2.12}$$

When $l = 0$, the equation is

$$\frac{\partial^2 \Psi}{\partial t^2} = \frac{\partial^2 \Psi}{\partial r^2}, \tag{2.13}$$

the simple $1+1$ wave equation describing waves moving on, say, a string or wire. A general solution to equation (2.13) is given by the superposition,

$$\Psi(t,r) = g(t+r) + f(t-r), \tag{2.14}$$

of incoming $g(t+r)$ and outgoing $f(t-r)$ waves. Our goal now is to write down a formula for $l \neq 0$ which is analogous to the outgoing solution $f(t-r)$. A formula generalizing $g(t+r)$ could also be given, but we are only interested in the outgoing solution. Since we are now concerned with how the solution Ψ depends on l, we adorn Ψ_l with an index l. Our goal now is to prove the following result.

Claim 2.1.1. *For any sufficiently smooth function f, the following expression is an outgoing solution to the RWE.*

$$\Psi_l(t,r) = \sum_{k=0}^{l} \frac{C_{lk}}{r^k} f^{(l-k)}(t-r), \qquad C_{lk} = \frac{1}{2^k} \frac{(l+k)!}{k!(l-k)!}. \tag{2.15}$$

As an example, for $l = 3$, we have

$$\Psi_3(t,r) = f'''(t-r) + \frac{6}{r} f''(t-r) + \frac{15}{r^2} f'(t-r) + \frac{15}{r^3} f(t-r), \tag{2.16}$$

which solves

$$\frac{\partial^2 \Psi_3}{\partial t^2} = \frac{\partial^2 \Psi_3}{\partial r^2} - \frac{12}{r^2} \Psi_3. \tag{2.17}$$

The proof of the claim follows from two lemmas.

Lemma 2.1.2. *The formula* (2.15) *obeys the following identity:*

$$\Psi_l(t,r) = D_l^+ \Psi_{l-1}(t,r), \text{ where } D_l^+ = -\frac{\partial}{\partial r} + \frac{l}{r}. \tag{2.18}$$

Proof. Using (2.15) with l replaced by $l-1$, we have

$$\Psi_{l-1} = \sum_{k=0}^{l-1} \frac{1}{r^k} C_{l-1,k} f^{(l-1-k)}(t-r). \tag{2.19}$$

Now, applying D_l^+ to the last formula, we get:

$$\begin{aligned} D_l^+ \Psi_{l-1} &= \left[-\frac{\partial}{\partial r} + \frac{l}{r}\right] \left[\sum_{k=0}^{l-1} \frac{1}{r^k} C_{l-1,k} f^{(l-1-k)}(t-r)\right] \\ &= -\frac{\partial}{\partial r}\left[\sum_{k=0}^{l-1} \frac{C_{l-1,k}}{r^k} f^{(l-1-k)}(t-r)\right] + \frac{l}{r} \sum_{k=0}^{l-1} \frac{C_{l-1,k}}{r^k} f^{(l-1-k)}(t-r). \end{aligned} \tag{2.20}$$

We find

$$\begin{aligned} \text{term 1} =& \sum_{k=0}^{l-1} \frac{C_{l-1,k}}{r^k} f^{(l-k)}(t-r) + \sum_{k=0}^{l-1} \frac{k}{r^{k+1}} C_{l-1,k} f^{(l-1-k)}(t-r) \\ =& C_{l-1,0} f^{(l)}(t-r) + \sum_{k=1}^{l-1} \frac{C_{l-1,k}}{r^k} f^{(l-k)}(t-r) \\ &+ \sum_{k=0}^{l-1} \frac{k}{r^{k+1}} C_{l-1,k} f^{(l-1-k)}(t-r) \end{aligned} \tag{2.21}$$

For the second term of this term 1, let $k = p+1$, then

$$\sum_{k=1}^{l-1} \frac{C_{l-1,k}}{r^k} f^{(l-k)}(t-r) = \sum_{p=0}^{l-2} \frac{C_{l-1,p+1}}{r^{p+1}} f^{(l-1-p)}(t-r). \tag{2.22}$$

Combining the above expression and reindexing after also expanding the last term, we have

$$\begin{aligned} \text{term 1} =& C_{l-1,0} f^{(l)}(t-r) + \sum_{k=0}^{l-2} \frac{C_{l-1,k+1}}{r^{k+1}} f^{(l-1-k)}(t-r) \\ &+ \sum_{k=0}^{l-2} \frac{k}{r^{k+1}} C_{l-1,k} f^{(l-1-k)}(t-r) + \frac{(l-1)C_{l-1,l-1}}{r^l} f(t-r). \end{aligned} \tag{2.23}$$

Using the same approach, we find

$$\begin{aligned}\text{term } 2 =& \frac{l}{r}\sum_{k=0}^{l-1}\frac{C_{l-1,k}}{r^k}f^{(l-1-k)}(t-r)\\ =&\sum_{k=0}^{l-2}\frac{l}{r^{k+1}}C_{l-1,k}f^{(l-1-k)}(t-r)+\frac{l}{r^l}C_{l-1,l-1}f(t-r).\end{aligned}\tag{2.24}$$

Then the two terms in (2.20) combine to

$$\begin{aligned}D_l^+\Psi_{l-1} =&\text{term } 1+\text{term } 2\\ =&C_{l-1,0}f^{(l)}(t-r)\\ &+\sum_{k=0}^{l-2}\left[C_{l-1,k+1}+kC_{l-1,k}+lC_{l-1,k}\right]\frac{1}{r^{k+1}}f^{(l-1-k)}(t-r)\\ &+\left[(l-1)C_{l-1,l-1}+lC_{l-1,l-1}\right]\frac{1}{r^l}f(t-r)\end{aligned}\tag{2.25}$$

Now with $C_{l-1,0} =1, \quad (2l-1)\ C_{l-1,l-1}=C_{l,l},$ and

$$C_{l,k+1} =C_{l-1,k+1}+(k+l)C_{l-1,k},$$

we see that (2.25) may be written as follows:

$$D_l^+\Psi_{l-1} =f^{(l)}(t-r)+\sum_{p=1}^{l-1}\frac{C_{l-1,p}}{r^p}f^{(l-p)}(t-r)+\frac{C_{ll}}{r^l}f(t-r)\tag{2.26}$$

$$=\sum_{p=0}^{l}\frac{C_{lp}}{r^p}f^{(l-p)}(t-r).\tag{2.27}$$

Adding up the results, one finds

$$\Psi_l = D_l^+\Psi_{l-1.}\tag{2.28}$$

□

Lemma 2.1.3. *The expression* (2.15) *solves* (2.11)

Proof. We proceed by induction. We have already noted that $f(t-r)$ solves the $l=0$ RWE. Now assume Ψ_{l-1} obeys:

$$\frac{\partial^2 \Psi_{l-1}}{\partial t} = \frac{\partial^2}{\partial r^2}\Psi_{l-1} - \frac{l(l-1)}{r^2}\Psi_{l-1}, \tag{2.29}$$

then we want to show that $\Psi = D_l^+ \Psi_{l-1}$ solves (2.11)

To do so, we apply the raising operator D_l^+ on (2.29) to get

$$\frac{\partial^2}{\partial t^2} D_l^+ = D_l^+ \frac{\partial^2}{\partial r^2}\Psi_{l-1} - l(l-1)D_l + \frac{1}{r^2}\Psi_{l-1} \tag{2.30}$$

Next, a long calculation shows that

$$\frac{\partial^2}{\partial t^2} D_l^+ \Psi_{l-1} = \frac{\partial^2}{\partial r^2} D_l^+ \Psi_{l-1} - \frac{l(l-1)}{r^2} D_l^+ \Psi_{l-1} - \frac{2l}{r^2} D_l^+ \Psi_{l-1}. \tag{2.31}$$

Finally, because $l(l-1)+2l = l(l+1)$, we have

$$\frac{\partial^2 \Psi}{\partial t^2} = \frac{\partial^2 \Psi}{\partial r^2} - \frac{l(l+1)}{r^2}\Psi. \tag{2.32}$$

□

2.2 Laplace transform

Our goal here is to find the Laplace transforms of the outgoing solutions (2.15). To do this, we first define the Laplace transform:

$$\hat{\Psi}(s) = \int_0^\infty e^{-st}\Psi(t)\, dt. \tag{2.33}$$

We will need the following lemma.

Lemma 2.2.1. *Let $f \in C^\infty(\mathbb{R})$ obey $f(u) = 0$ for $u \notin [-B, -A]$, where $u = t - r$ is retarded time and $r > B$. Then*

$$\mathcal{L}\{f(\cdot - r)\} = \int_0^\infty e^{-st} f(t-r)\ dt$$
$$= e^{-sr} a(s), \tag{2.34}$$

where the function $a(s)$ is independent of r.

Proof. letting $u = t - r$; via change of variable,

$$\int_0^\infty e^{-st} f(t-r)\ dt = e^{-sr} \int_{-r}^\infty e^{-su} f(u)\ du. \tag{2.35}$$

When $t = 0, u = -r$ and when $t \to \infty, u \to \infty$. Then, $a(s) = \int_{-r}^\infty e^{-su} f(u)\ du$ or $a(s) = \int_{-B}^{-A} e^{-su} f(u)\ du$.

□

Clearly, $a(s)$ is independent of r because the function $f(u)$ is only supported on the interval $[-B, -A]$ which is independent of r. Notice that, $a(s)$ is an entire function, because the integral above is convergent for any $s \in \mathbb{C}$.

This lemma is proved in [8].

Claim 2.2.2. *Let f as in lemma 1.2.1, then, we have*

$$\int_0^\infty e^{-st} f^{(p)}(t-r)\ dt = a(s) s^p e^{-st}, \tag{2.36}$$

for any $s \in \mathbb{C}$. Let I be the integral on the left, since the function f is only supported on $[-B, -A]$ and is $C^\infty(\mathbb{R})$, we may integrate by parts with vanishing boundary terms. Our integral becomes,

$$I = s \int_0^\infty e^{-st} f^{(p-1)}(t-r)\ dt. \tag{2.37}$$

Clearly, by induction

$$\int_0^\infty e^{-st} f^{(p-1)}(t-r)\ dt = e^{-sr} s^p a(s). \tag{2.38}$$

The previous two lemmas show that the Laplace transform of (2.15) is:

$$\begin{aligned}\mathcal{L}\{\Psi_l(\cdot, r)\} = \hat{\Psi}_l(s,r) =& a(s)e^{-sr}\sum_{k=0}^{l} C_{lk}\frac{s^{l-k}}{r^k} \\ =& a(s)s^l e^{-sr}\sum_{k=0}^{l} C_{lk}\frac{1}{(sr)^k}.\end{aligned}$$

Otherwise expressed,

$$\hat{\Psi}_l(s,r) = a(s)s^l e^{-sr} W_l(sr), \qquad W_l(z) = \sum_{k=0}^{l} \frac{C_{lk}}{z^k}. \tag{2.39}$$

2.3 NtF kernels

This section examines the structure of near-to-far propagation kernels. Mostly adhering to the presentation from the preceding section (1.2), it uses equation (2.39) to derive an expression for a "kernel" which describes mapping the solution $\hat{\Psi}_l(s, r_1)$ at r_1 to $\hat{\Psi}_l(s, r_2)$ at $r_2 > r_1$. In the Laplace frequency $s-$domain, each kernel will be finite sum of simple poles in the complex $s-$plane. The function $W_l(z)$ is closely related to the modified Bessel function known as MacDonald's function. Indeed, by [7], we have

$$\sqrt{\frac{\pi}{2z}} K_{l+\frac{1}{2}}(z) = \left(\frac{\pi}{2z}\right) e^{-z} W_l(z). \tag{2.40}$$

It is known that $K_{l+1/2}(z)$, and so $W_l(z)$, has l simple zeros located in the left-half plane.

Also, let us denote these zeros by $\{b_{lj} : j = 1, \cdots, l\}$. For example, when $l = 1$,

$$W_1(z) = 1 + \frac{1}{z}. \tag{2.41}$$

So $b_{11} = -1$, when $l = 1$. When $l = 2$,

$$W_2(z) = 1 + \frac{3}{z} + \frac{3}{z^2}, \tag{2.42}$$

and then $\{b_{21}, b_{22}\}$ are the roots of the quadratic equation

$$z^2{+}3z + 3 = 0.$$

Therefore, $b_{21} = -\frac{3}{2} + \frac{i\sqrt{3}}{2}, \qquad b_{22} = -\frac{3}{2} - \frac{i\sqrt{3}}{2}.$

We will also consider the residues of the NtF kernel. The NtF kernel will afford us a procedure for converting a signal $\Psi(t, r_1)$ recorded at $r = r_1$ to the one $\Psi(t, r_2)$ recorded at $r = r_2$ with $r_2 > r_1$. As mentioned, our interest lies with a procedure in the time-domain, but its derivation uses the frequency domain. From equation (1.39), we have already seen that

$$\begin{aligned}\hat{\Psi}_l(s, r_1) =& a(s)s^l e^{-z_1} W_l(z_1),\\ \hat{\Psi}_l(s, r_2) =& a(s)s^l e^{-z_2} W_l(z_2),\end{aligned} \tag{2.43}$$

where $z_1 = sr_1, z_2 = sr_2$. When we assume that $r_2 > r_1 >$ support of the initial data. The relationship between the solution at different radii can be expressed as

$$\hat{\Psi}_l(s, r_2) = e^{-s(r_2 - r_1)} \left(\frac{W_l(sr_2)}{W_l(sr_1)} \right) \hat{\Psi}_l(sr_1). \tag{2.44}$$

When the terms are rearranged, we have

$$e^{(z_2 - z_1)} \hat{\Psi}_l(s, r_2) = \left[\frac{W_l(z_2)}{W_l(z_1)} - 1 \right] \hat{\Psi}_l(s, r_1) + \hat{\Psi}_l(s, r_1). \tag{2.45}$$

Now, we define

$$\hat{\Phi}_l(s, r_1, r_2) := \left[\frac{W_l(sr_2)}{W_l(sr_1)} - 1 \right], \tag{2.46}$$

and then we have

$$e^{s(r_2 - r_1)} \hat{\Psi}_l(s, r_2) = \hat{\Phi}_l(s, r_1, r_2) \hat{\Psi}_l(s, r_1) + \hat{\Psi}_l(s, r_1). \tag{2.47}$$

Remark 2.3.1. *Here the -1 factor ensures that the kernel $\hat{\Phi}_l(s, r_1, r_2)$ decays for large s, and this ensure the existence of the inverse Laplace transform.*

We can verify that $\hat{\Phi}_l(s, r_1, r_2) = 0$ as $s \to \infty$, and as $r_2 \to \infty$ that

$$\hat{\Phi}_l(s, r_1, r_2) = \frac{1 - W_l(sr_1)}{W_l(sr_1)}. \tag{2.48}$$

Reference [8] has shown that

$$\hat{\Phi}_l(s, r_1, r_2) = \sum_{j=1}^{l} \frac{a_{lj}(r_1, r_2)}{s - b_{lj}/r_1},$$

where the b_{lj} are the roots considered in (2.42) and the residues are given by [4],

$$a_{lj}(r_1, r_2) = \frac{W_l(b_{lj} r_2 r_1^{-1})}{r_1 W_l'(b_{lj})}. \tag{2.49}$$

Therefore, by using the well-known properties of the Laplace transforms, then the inverse Laplace transform of (2.47) is:

$$\Psi_l(t + (r_2 - r_1), r_2) = \int_0^t \Phi_l(t - t', r_1, r_2) \Psi_l(t', r_1)\ dt' + \Psi_l(t, r_1). \tag{2.50}$$

where the time-domain kernel is a sum of exponentials [4, 6],

$$\Phi_l(t_1, r_1, r_2) = \sum_{k=1}^{l} a_{lk}(r_1, r_2) e^{(b_{lk}/r_1)t}. \tag{2.51}$$

Chapter 3

Approximation of functions

3.1 Polynomial Approximation (warm-up)

We start with the following classical problem.

3.1.1 Theoretical results

Problem 3.1.1. *Given $f \in C[a,b]$ either non-polynomial or a polynomial of degree $\gg d$, find the polynomial p of degree d which minimizes*

$$\|f - p\|^2_{L_2[a,b]} = \int_a^b |f(\xi) - p(\xi)|^2 d\xi. \tag{3.1}$$

Remark 3.1.2. *The optimal polynomial (solution to the problem) is characterized by the* Galerkin Condition: *$f - p$ is orthogonal to all polynomials of degree d or less. That is, for any polynomial q of degree d or less, we have:*

$$\int_a^b q(\xi)[f(\xi) - p(\xi)]d\xi = 0. \tag{3.2}$$

Since it is more closely related to the numerical approximations considered later, we now describe a discrete version of the problem. Introduce the grid $a = \xi_1 < \xi_2 < \cdots < \xi_N = b$ on $[a,b]$. Here,

$$\xi_k = a + \frac{(k-1)}{(N-1)}(b-a), \quad \text{for} \quad k = 1, 2, \cdots, N. \tag{3.3}$$

Assume $N \gg d$. Define the vectors

$$\vec{h}_j = \left(h_j(1), h_j(2), \cdots, h_j(N)\right)^\top, \tag{3.4}$$

by $\quad h_j(k) = \xi_k^{j-1}$. Then for example,

$$\vec{h}_1 = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}, \quad \vec{h}_2 = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_N \end{pmatrix}, \quad \vec{h}_3 = \begin{pmatrix} \xi_1^2 \\ \xi_2^2 \\ \vdots \\ \xi_N^2 \end{pmatrix}, \quad \text{etc.} \tag{3.5}$$

We also use the notations

$$\vec{\xi^0} = (1, 1, \cdots, 1)^\top \tag{3.6}$$

$$\vec{\xi^1} = \left(\vec{\xi_1}, \vec{\xi_2}, \cdots, \vec{\xi_N}\right)^\top \tag{3.7}$$

$$\vec{\xi^2} = \left(\vec{\xi_1^2}, \vec{\xi_2^2}, \cdots, \vec{\xi_N^2}\right)^\top \tag{3.8}$$

$$\vdots$$

$$\vec{\xi^d} = \left(\vec{\xi_1^d}, \vec{\xi_2^d}, \cdots, \vec{\xi_N^d}\right)^\top, \tag{3.9}$$

that is $\vec{h}_j = \vec{\xi}^{j-1}$.

Claim 3.1.3. *For $d \le N - 1$ the set $\left\{\vec{h}_1, \vec{h}_2, \ldots, \vec{h}_{d+1}\right\}$ is linearly independent.*

Proof. Consider

$$\sum_{j=1}^{d+1} c_j \vec{h}_j = 0. \tag{3.10}$$

This equation can be written in matrix form

$$V\vec{c} = \vec{0}, \tag{3.11}$$

where

$$V = \begin{pmatrix} 1 & \xi_1 & \xi_1^2 & \cdots & \xi_1^d \\ 1 & \xi_2 & \xi_2^2 & & \xi_2^d \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \xi_N & \xi_N^2 & & \xi_N^d \end{pmatrix} \in \mathbb{R}^{N\times(d+1)}. \tag{3.12}$$

The equation $V\vec{c} = \vec{0}$ implies $\widehat{V}\vec{c} = \vec{0}$, where $\widehat{V} \in \mathbb{R}^{(d+1)\times(d+1)}$ arises from V after truncation of all but the first $d+1$ rows. If $d = N-1$, then no truncation is required. It is known that $\det \hat{V} = \prod_{1\le i<j\le d+1} (\xi_j - \xi_i) \neq 0$ [7].Therefore, $V\vec{c} = \vec{0}$ is only solved by $\vec{c} = \vec{0}$. □

A discrete version of the continuum problem stated in Problem 3.1.1 is as follows.

Problem 3.1.4. *(discrete problem): Assume $d+1 < N$.*

Given $\vec{f} \in \mathbb{R}^N$ not in span $\left(\vec{h}_1, \vec{h}_2, \ldots, \vec{h}_{d+1}\right)$ *, find $\vec{p} \in$ span $\left(\vec{h}_1, \vec{h}_2, \ldots, \vec{h}_{d+1}\right)$ which minimizes*

$$\|\vec{f} - \vec{p}\|^2 = \sum_{k=1}^{N} |f(k) - p(k)|^2. \tag{3.13}$$

Theorem 3.1.5. *The optimal*

$$\vec{p} = \sum_{j=1}^{d+1} c_j \vec{h}_j$$

obeys $h_l^T(f-p)$, for all $l = 1, 2, \cdots, d+1$.

Proof. Consider

$$\vec{p} = \left[\vec{h}_1, \vec{h}_2, \ldots, \vec{h}_{d+1}\right] \begin{bmatrix} c_1 \\ \vdots \\ c_{d+1} \end{bmatrix}. \tag{3.14}$$

If V is the matrix $\left(\vec{h}_1, \ldots, \vec{h}_{d+1}\right)$, then $\vec{p} = V\vec{c}$, where $\vec{c} = (c_1, \ldots, c_{d+1})^\top$, and the stated Galerkin condition is $V^\top(\vec{f} - V\vec{c}) = \vec{0}$, equivalent to the normal equations $V^\top V\vec{c} = V^\top \vec{f}$.

Since the columns of V are linearly independent, $V^\top V \in \mathbb{R}^{(d+1)\times(d+1)}$ is nonsingular. Let $\vec{c}_{LS}$ (LS for least squares) be the unique solution to the normal equations. For any $\vec{c}$, write $\vec{c} = \vec{c}_{LS} + (\vec{c} - \vec{c}_{LS}) := \vec{c}_{LS} + \vec{e}$. Then

$$\begin{aligned}
\|\vec{f} - \vec{p}\|^2 &= \|\vec{f} - V\vec{c}\|^2 \\
&= \left\|\vec{f} - V\left(\vec{c}_{LS} + \vec{e}\right)\right\|^2 \\
&= \left(\vec{f} - V\vec{c}_{LS} - V\vec{e}\right)^\top \left(\vec{f} - V\vec{c}_{LS} - V\vec{e}\right) \\
&= \left(\vec{f} - V\vec{c}_{LS}\right)^\top \left(\vec{f} - V\vec{c}_{LS}\right) + (V\vec{e})^\top V\vec{e} + 2(V\vec{e})^\top \left(\vec{f} - V\vec{c}_{LS}\right) \\
&= \left\|\vec{f} - V\vec{c}_{LS}\right\|^2 + \|V\vec{e}\|^2 + e^\top \left(V^\top\vec{f} - V^\top V\vec{c}_{LS}\right) \\
&= \left\|\vec{f} - V\vec{c}_{LS}\right\|^2 + \|V\vec{e}\|^2, \qquad (3.15)
\end{aligned}$$

where the last equality used the normal equations.

Now $\|V\vec{e}\|^2 \geq 0$, with equality only if $\vec{e}$ is the zero vector. This shows that $\vec{p} = V\vec{c}_{LS}$ is the unique minimizer. □

Theorem 3.1.6. *At the minimizer the gradient* $\left(\nabla_{\vec{c}}\|\vec{f} - \vec{p}\|^2\right)\Big|_{\vec{c}=\vec{c}_{LS}} = \vec{0}$ *vanishes.*

Indeed, for any $l = 1, 2, \cdots, d+1$

$$\begin{aligned}
\frac{\partial}{\partial c_l}\|\vec{f} - \vec{p}\|^2 &= \frac{\partial}{\partial c_l}\sum_{k=1}^{N}\left|f(k) - \sum_{j=1}^{d+1} c_j h_j(k)\right|^2 \\
&= -2\sum_{k=1}^{N}\left(f(k) - \sum_{j=1}^{d+1} c_j h_j(k)\right) h_l(k) \\
&= -2\langle \vec{f} - \vec{p}, \vec{h}_l\rangle. \qquad (3.16)
\end{aligned}$$

To find the optimal $\vec{p}$, we first produce an orthogonal basis $\{\vec{g}_1, \vec{g}_2, \ldots \vec{g}_{d+1}\}$.

For span $\left(\vec{h}_1, \vec{h}_2, \cdots, \vec{h}_{d+1}\right)$, we also want nested spans in the sense that

$$\begin{aligned}\operatorname{span}(\vec{g}_1) &= \operatorname{span}(\vec{h}_1)\\ &\vdots\\ \operatorname{span}(\vec{g}_1, \vec{g}_2, \ldots, \vec{g}_{d+1}) &= \operatorname{span}(\vec{h}_1, \vec{h}_2, \ldots, \vec{h}_{d+1}).\end{aligned} \tag{3.17}$$

Assume that we have produced such an orthogonal basis. Then

$$\vec{p} = \sum_{j=1}^{d+1} \alpha_j \vec{g}_j \quad \text{for} \qquad \alpha_j = \frac{\langle \vec{f}, \vec{g}_j \rangle}{\langle \vec{g}_j, \vec{g}_j \rangle}, \tag{3.18}$$

is the solution to the discrete problem. Indeed, we show that

$$\vec{f} - \vec{p} = \vec{f} - \sum_{j=1}^{d+1} \alpha_j \vec{g}_j, \tag{3.19}$$

is orthogonal to all $\vec{g}_1, \vec{g}_2, \cdots, \vec{g}_{d+1}$ and so all $\vec{h}_1, \vec{h}_2, \cdots, \vec{h}_{d+1}$. The justification is as follows.

$$\begin{aligned}\langle \vec{f} - \vec{p}, \vec{g}_l \rangle &= \langle \vec{f}, \vec{g}_l \rangle - \langle \vec{p}_j, \vec{g}_l \rangle \qquad (3.20)\\ &= \langle \vec{f}, \vec{g}_l \rangle - \sum_{j=1}^{d+1} \alpha_j \langle \vec{g}_j, \vec{g}_l \rangle\\ &= \alpha_\ell - \alpha_\ell\\ &= 0,\end{aligned}$$

where we have used

$$\langle \vec{g}_j, \vec{g}_l \rangle = \delta_{jl} \langle \vec{g}_j, \vec{g}_j \rangle. \tag{3.21}$$

Since by assumption any of the vectors in $\{\vec{h}_1, \vec{h}_2, \ldots, \vec{h}_{d+1}\}$ can be expanded in terms of the $\{\vec{g}_1, \vec{g}_2, \ldots, \vec{g}_{d+1}\}$, then it follows that $\langle \vec{f} - \vec{p}, \vec{h}_l \rangle = 0$ for $l = 1, 2, \ldots, d + 1$.

3.1.2 Implementation

The algorithm for constructing the $\{\vec{g}_1, \vec{g}_2, \ldots, \vec{g}_{d+1}\}$ basis is given in **Algorithm 1**. It relies on the Gram-Schmidt process and an underlying three-term recurrence specific to the problem at hand.This is a discrete version of the process which is starting with the monomial basis for polynomials, gives rise to the Legendre polynomials.

Algorithm 1 Discrete polynomial approximation of $\vec{f}$.

Input: fixed vector $\vec{f} \notin \text{span}\{\vec{h}_1, \ldots, \vec{h}_{d+1}\}$

Output: $\vec{\alpha} \in \mathbb{R}^{d+1}, c \in \mathbb{R}^{(d+1)\times 2}$

Description: Generates an orthogonal basis $\{\vec{g}_1, \ldots, \vec{g}_{d+1}\}$ for $\text{span}\{\vec{h}_1, \ldots, \vec{h}_{d+1}\}$ and computes as output the expression

$$\vec{p} = \sum_{j=1}^{d+1} \alpha_j \vec{g}_j,$$

where $\vec{p}$ is the optimal approximation of $\vec{f}$ in the least squares sense. Since $\vec{p} \in \text{span}\{\vec{h}_1, \ldots, \vec{h}_{d+1}\}$, its components are $p_j \in p(\xi_j)$, where $p(\xi)$ is a polynomial of degree d. **Algorithm 1** also computes the list of inner products c which can be used to evaluate $p(\xi)$ at arbitrary point ξ (not necessarily one of the grid points ξ_j)

```
$\alpha = 0_{(d+1)\times 1}$
$c = 0_{(d+1)\times 2}$
$\vec{g}_{\text{old}} = 0_{N\times 1}$
$\vec{g}_{\text{now}} = 0_{N\times 1}$
$\vec{g}_{\text{new}} = 0_{N\times 1},\ \vec{g}_{\text{new}}(:) = 1$
$\alpha_1 = \langle \vec{f}, \vec{g}_{\text{new}}\rangle / \langle \vec{g}_{\text{new}}, \vec{g}_{\text{new}}\rangle$
$\vec{g}_{\text{old}} = \vec{g}_{\text{now}}$
$\vec{g}_{\text{now}} = \vec{g}_{\text{new}}$
$\vec{v} = \vec{\xi}. * \vec{g}_{\text{now}} \in \text{span}(\vec{h}_1, \ldots, \vec{h}_{k+1}) = \text{span}(\vec{g}_1, \ldots, \vec{g}_{k+1})$
$c_{21} = \langle \vec{v}, \vec{g}_{\text{now}}\rangle / \langle \vec{g}_{\text{now}}, \vec{g}_{\text{now}}\rangle$
$c_{22} = 0$
$\vec{g}_{\text{new}} = \vec{v} - c_{21}\vec{g}_{\text{now}} - c_{22}\vec{g}_{\text{old}}$
$\alpha_2 = \langle f, \vec{g}_{new}\rangle / \langle \vec{g}_{new}, \vec{g}_{new}\rangle$
for $k = 2$ up to $d$ do
    $\vec{v} = \vec{\xi}. * \vec{g}_{\text{now}}, \vec{v}$ is candidate for $g_{k+1}$
    $\vec{v} = \beta\vec{g}_{k+1} + c_{21}\vec{g}_k + c_{22}\vec{g}_{k-1} +$ nothing else
    $c_{k+1,1} = \langle \vec{v}, \vec{g}_{\text{now}}\rangle / \langle \vec{g}_{\text{now}}, \vec{g}_{\text{now}}\rangle$
    $c_{k+1,2} = \langle \vec{v}, \vec{g}_{\text{old}}\rangle / \langle \vec{g}_{\text{old}}, \vec{g}_{\text{old}}\rangle$
    $\vec{g}_{\text{new}} = \vec{v} - c_{k+1,1}\vec{g}_{\text{now}} - c_{k+1,2}\vec{g}_{\text{old}}$
    $\alpha_{k+1} = \langle \vec{f}, \vec{g}_{new}\rangle / \langle \vec{g}_{new}, \vec{g}_{new}\rangle$
    $\vec{g}_{\text{old}} = \vec{g}_{\text{now}}$
    $\vec{g}_{\text{now}} = \vec{g}_{\text{new}}$
end for
```

From **Algorithm 1,** the component $p_k = \sum_{j=1}^{d+1} c_j h_j(k)$ is a polynomial function $p(\xi_k)$ evaluated on the grid points. However, we also want to evaluate $p(\xi)$ and $p'(\xi)$ at a value of ξ

which is not necessary equal to one of the grid points ξ_j. This is achieved with the following algorithm.

Algorithm 2 Input: $\vec{\alpha} \in \mathbb{R}^{d+1}, c \in \mathbb{R}^{(d+1)\times 2}$ from **Algorithm 1**, and $\xi \in \mathbb{R}$. Output: $p = p(\xi)$ and $dp = p'(\xi)$

```
$g_{old} = 1$
$g_{now} = \xi - c_{21} g_{old}$
$dg_{old} = 0$
$dg_{now} = dg_{old}$
$p = \alpha_1 g_{old} + \alpha_2 g_{now}$
$dp = \alpha_1 dg_{old} + \alpha_2 dg_{now}$
for $k = 2 : n$ do
    $g_{new} = \xi g_{now} - c_{k+1,1} g_{now} - c_{k+1,2} g_{old}$
    $dg_{new} = \xi dg_{now} + g_{now} - c_{k+1,1} dg_{now} - c_{k+1,2} dg_{old}$
    $p = p + \alpha_{k+1} g_{new}$
    $dp = dp + \alpha_{k+1} dg_{new}$
    $g_{old} = g_{now}$
    $g_{now} = g_{new}$
    $dg_{old} = dg_{now}$
    $dg_{now} = dg_{new}$
end for
```

Remark 3.1.7. *We could have the Algorithm* 1 *generate and store all* $\vec{g}_j$, *but instead, only three such vectors are stored at a time. The process relies on a three-term recurrence.*

3.2 Rational Approximation

Rational approximation of a function f expresses the function as the ratio of two polynomials

$$f(s) \simeq \frac{P(s)}{Q(s)}, \tag{3.22}$$

where for our applications s is the complex variable which arises in the Laplace transform. The approximation will be valid for s values in a region of the complex plane. For our approximations, we will demand that $\deg(Q) = d = \deg(P) - 1$, so that $P(s)/Q(s)$ is a proper rational function. Before discussing how to construct rational approximations, we first describe the class of functions we shall consider.

3.2.1 Complex-valued functions with the parity property

The constructed approximations will be valid on $s = x + iy$, where $x = \varepsilon > 0$ is a small positive number, and in some cases $x = 0$ is also allowed. To focus on the y-dependence, introduce the notation

$$\mathcal{F}(y) = f(x+iy), \quad \mathcal{P}(y) = P(x+iy), \quad \mathcal{Q}(y) = Q(x+iy).$$

Definition 3.2.1. *We say that a complex function $h(y) = h^R(y) + ih^I(y)$ has the parity property provided its real part $h^R(y)$ has even parity in y and its imaginary part $h^I(y)$ has odd parity.*

As an example, consider the function $f(s) = 1/\sqrt{1+s^2}$, viewed as a function $\mathcal{F}(y) = 1/\sqrt{1+(\varepsilon+iy)^2}$ of y.

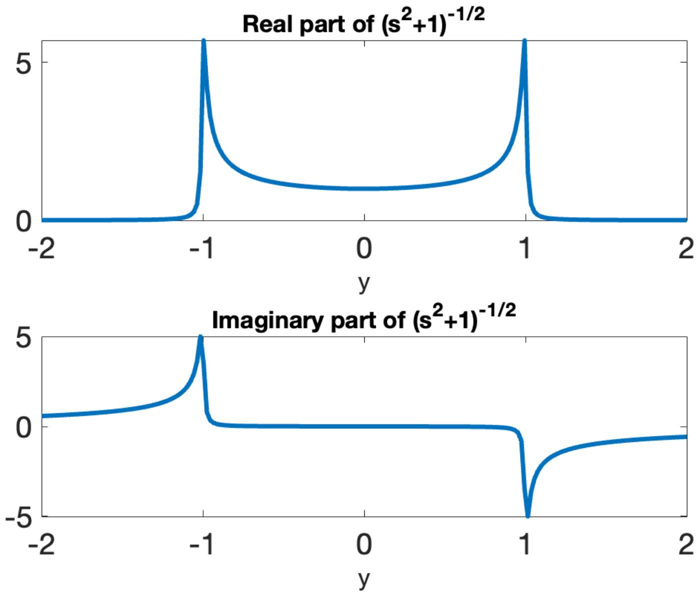

Figure 3.1: Real and imaginary parts of $\mathcal{F}(y) = 1/\sqrt{1+(\varepsilon+iy)^2}$ for $\varepsilon = 10^{-2} = 0.01$, showing that $\mathcal{F}(y)$ obeys the parity property.

This function is the Laplace transform of $J_0(t)$ [7]. The basic variable $s = x + iy$ has the parity properity, and we conclude by induction that any power $s^j = (x+iy)^j$ does too. Provided the coefficients $p_j, q_j \in \mathbb{R}$ for all $j = 0, 1, \ldots, d-1$, the polynomials

$$\mathcal{P}(y) = P(x+iy) = \sum_{j=0}^{d-1} p_j(x+iy)^j, \quad \mathcal{Q}(y) = Q(x+iy) = \sum_{j=0}^{d-1} q_j(x+iy)^j + (x+iy)^d \quad (3.23)$$

both have the parity property, since they are linear combinations of functions each with the property.

Lemma 3.2.2. *Let $\mu(y) > 0$ be a positive weight function which also has even parity in y, and define the inner product*

$$\langle h, g\rangle_\mu = \int_{-y_{\max}}^{y_{\max}} h(y)\bar{g}(y)\mu(y)dy.$$

If h and g both have the parity property, then this inner product is a real number.

Proof. In detail the inner product in question is

$$\langle h, g\rangle_\mu = \int_{-y_{\max}}^{y_{\max}} \left\{ \left[h^R(y)g^R(y) + h^I(y)g^I(y)\right] + i\left[h^I(y)g^R(y) - h^R(y)g^I(y)\right] \right\} \mu(y)dy.$$

Since $\left[h^I(y)g^R(y) - h^R(y)g^I(y)\right]\mu(y)$ is of odd parity, and the integration is symmetric about $y = 0$, we see that

$$\langle h, g\rangle_\mu = \int_{-y_{\max}}^{y_{\max}} \left[h^R(y)g^R(y) + h^I(y)g^I(y)\right]\mu(y)dy \in \mathbb{R}.$$

□

3.2.2 Nonlinear problem and its linearization

The polynomials appearing in our approximations will arise as

$$P_{\text{opt}}, Q_{\text{opt}} = \arg\min_{P,Q} \int_{-y_{\max}}^{y_{\max}} \left| \frac{P(x+iy)}{Q(x+iy)} - f(x+iy) \right|^2 dy,$$

as indicated with $\deg(Q) = d = \deg(P) - 1$. In practice, the integral should be discretized in a fashion similar to the first subsection on polynomial approximation. However, we bypass these details for now and proceed with the continuum formulation. Anticipating that $\mathcal{F}(y)$ with have the parity property, we have choosen symmetric integration limits here.

The problem above is nonlinear, and would be difficulty to solve directly. We therefore trade it for a sequence of linear minimizations

$$\begin{aligned} P_{\text{opt}}^{(k+1)}, Q_{\text{opt}}^{(k+1)} = \underset{P^{(k+1)},Q^{(k+1)}}{\arg\min} \int_{-y_{\max}}^{y_{\max}} \left|P^{(k+1)}(x+iy) - Q^{(k+1)}f(x+iy)\right|^2 \mu_k(y)dy \\ \text{where } \mu_k(y) = \frac{1}{\left|Q^{(k)}(x+iy)\right|^2}. \end{aligned} \tag{3.24}$$

This procedure requires an initial degree-d polynomial $Q^0(s)$ in order to define the initial weight function μ_0. The procedure is a form of fixed-point iteration, but we shall not address its convergence as $k \to \infty$. Rather we will confirm in numerical experiments that $P^{(k+1)}$ and $Q^{(k+1)}$ "settle down" for k sufficiently large.

Adopting the notation (3.23), the linear problem (3.24) amounts to finding P, Q as described which minimize the following cost function:

$$\begin{aligned} C(P, Q) := C\left(p_0, \ldots, p_{d-1}, q_0, \ldots, q_{d-1}\right) &= \int_{-y_{\max}}^{y_{\max}} |\mathcal{P}(y) - \mathcal{Q}(y)\mathcal{F}(y)|^2 \mu(y) dy \\ &= \int_{-y_{\max}}^{y_{\max}} |P(x+iy) - Q(x+iy)f(x+iy)|^2 \mu(y) dy. \end{aligned} \tag{3.25}$$

Theorem 3.2.3. *Define the functions*

$$h_n(y) = \begin{cases} (x+iy)^{(n-2)/2} & \text{for } n = 2, 4, \ldots, 2d \\ (x+iy)^{(n-1)/2} f(x+iy) & \text{for } n = 1, 3, \ldots, 2d-1. \end{cases}$$

Then, because $\mathcal{F}(y)$ has the parity property by assumption, the optimal choice of P, Q is characterized by the conditions

$$\langle -\mathcal{P} + \mathcal{Q}\mathcal{F}, h_n \rangle_\mu = 0$$

for all $n = 1, 2, 3, \ldots, 2d-1, 2d$.

Proof. Let $\Delta\mathcal{P}(y) - \Delta\mathcal{Q}(y)\mathcal{F}(y)$ arise as any linear combination with real coefficients of the functions $h_n(y)$ for all $n = 1, 2, 3, \ldots, 2d-1, 2d$. Then $\Delta\mathcal{P}(y) - \Delta\mathcal{Q}(y)\mathcal{F}(y)$ will have the parity property. Consider the cost function (3.25), now associated with $\mathcal{P}(y) + \Delta\mathcal{P}(y)$ and $\mathcal{Q}(y) + \Delta\mathcal{Q}(y)$. Since $\Delta\mathcal{Q}(y)$ does not include the top degree-d term, the expressions $\mathcal{P}(y) - \mathcal{Q}(y)\mathcal{F}(y)$ and $[\mathcal{P}(y) + \Delta\mathcal{P}(y)]-$ $[\mathcal{Q}(y) + \Delta\mathcal{Q}(y)]\mathcal{F}(y)$ have the same form. We may then consider the cost

$$\begin{aligned}
C(P+\Delta P, Q+\Delta Q) &= \int_{-y_{\max}}^{y_{\max}} |\mathcal{P}(y)-\mathcal{Q}(y)\mathcal{F}(y)+\Delta\mathcal{P}(y)-\Delta\mathcal{Q}(y)\mathcal{F}(y)|^2\mu(y)dy \\
&= \int_{-y_{\max}}^{y_{\max}} |\mathcal{P}(y)-\mathcal{Q}(y)\mathcal{F}(y)|^2\mu(y)dy \\
&+ \int_{-y_{\max}}^{y_{\max}} |\Delta\mathcal{P}(y)-\Delta\mathcal{Q}(y)\mathcal{F}(y)|^2\mu(y)dy \\
&+ \int_{-y_{\max}}^{y_{\max}} [\mathcal{P}(y)-\mathcal{Q}(y)\mathcal{F}(y)]\overline{[\Delta\mathcal{P}(y)-\Delta\mathcal{Q}(y)\mathcal{F}(y)]}\mu(y)dy \\
&+ \int_{-y_{\max}}^{y_{\max}} [\Delta\mathcal{P}(y)-\Delta\mathcal{Q}(y)\mathcal{F}(y)]\overline{[\mathcal{P}(y)-\mathcal{Q}(y)\mathcal{F}(y)]}\mu(y)dy.
\end{aligned} \tag{3.26}$$

Now using the definition (3.25) and **Lemma 3.2.2** to conclude that the inner products encountered here are real, we find that

$$\begin{aligned}
C(P+\Delta P, Q+\Delta Q) = C(P,Q) &+ 2(\mathcal{P}-\mathcal{Q}\mathcal{F}, \Delta\mathcal{P}-\Delta\mathcal{Q}\mathcal{F}\rangle_\mu \\
&+ \int_{-y_{\max}}^{y_{\max}} |\Delta\mathcal{P}(y)-\Delta\mathcal{Q}(y)\mathcal{F}(y)|^2\mu(y)dy.
\end{aligned} \tag{3.27}$$

The last term in this expression cannot be properly expressed as $C(\Delta P, \Delta Q)$, since, as noted above, ΔQ does not contain a top degree-d term. If P, Q are chosen to ensure that $-\mathcal{P}+\mathcal{Q}\mathcal{F}$ is orthogonal to all h_n, then

$$C(P+\Delta P, Q+\Delta Q) = C(P,Q) + \int_{-y_{\max}}^{y_{\max}} |\Delta\mathcal{P}(y)-\Delta\mathcal{Q}(y)\mathcal{F}(y)|^2\mu(y)dy. \tag{3.28}$$

This confirms that the described characterization of P, Q indeed minimizes the cost function.

□

Remark 3.2.4. *One reaches the same conclusion of the last theorem via formal partial differentiation of the cost function* $C(p_0, \ldots, p_{d-1}, q_0, \ldots, q_{d-1})$ *with respect to the* p_j *and* q_j.

We have characterized the P, Q which solve the relevant linear minimization problem. To actually solve the problem, we follow a procedure similar to the described Gram-Schmidt procedure used to solve the simple least squares problem studied in **Subsection 3.2.1**. To describe this procedure, we introduce a new notation.

Remark 3.2.5. *Given a function of y, say $\mathcal{F}$, use $s\mathcal{F}$ to denote the function of y with values $(x+iy)\mathcal{F}(y)$. Likewise, use $s^j\mathcal{F}$ to denote the function of y with values $(x+iy)^j\mathcal{F}(y)$.*

With Theorem 2.2 in mind, we orthogonalize the $2d+1$ functions

$$\mathcal{F}, 1, s\mathcal{F}, s, \ldots, s^{d-1}\mathcal{F}, s^{d-1}, s^d\mathcal{F}. \tag{3.29}$$

The result is a family $\{g_n : n = 1, \ldots, 2d\}$ of orthogonal functions, with the last member g_{2d+1} the sought for solution $-\mathcal{P}+\mathcal{Q}\mathcal{F}$ from **Theorem 3.2.3**. The following is proved in [9]

Theorem 3.2.6. *The orthogonalization is iteratively computed as follows:*

$$g_n(y) = \begin{cases} \mathcal{F}(y) & \textit{for } n = 1 \\ 1 - c_{21}\mathcal{F}(y) & \textit{for } n = 2 \\ (x+\mathrm{i}y)g_{n-2}(y) - \sum_{j=1}^{\min[4,n-1]} c_{nj}g_{n-j}(y) & \textit{for } n = 3, \ldots, 2d+1, \end{cases} \tag{3.30}$$

where the real constants c_{nj} are given by

$$c_{nj} = \begin{cases} \frac{\langle 1,\mathcal{F}\rangle_k}{\langle \mathcal{F},\mathcal{F}\rangle_k} & \textit{for } n = 2 \textit{ and } j = 1 \\ \frac{\langle (sg_{n-2}),g_{n-j}\rangle_k}{\langle g_{n-j},g_{n-j}\rangle_k} & \textit{for } n = 3, \ldots, 2d+1 \textit{ and } j = 1, \ldots, \min\{4, n-1\}. \end{cases} \tag{3.31}$$

Here we use the notation $(sg_{n-2})(y)$ for the function $(x+iy)g_{n-2}(y)$, as mentioned above. Also, the inner product $\langle,\rangle_k$ is the inner product $\langle,\rangle_\mu$ with the kth weight $\mu_k(y)$.

3.2.3 Evaluation of P and Q and their derivatives

The algorithm for evaluation of $P(s), P'(s), Q(s)$, and $Q'(s)$ at an arbitrary complex point s is analogous to the one given in Algorithm 2 for evaluation of the optimal polynomial $p(\xi)$ and its derivative $p'(\xi)$ stemming from the least squares problem. Theorem 2.2.6 describes orthogonalization of the set (2.29) which results in the coefficients $\{c_{nj} : n = 1, \ldots, 2d+1$ and $j = 1, \cdots, \min\{4, n-1\}\}$ which define $g_{2d+1} = -\mathcal{P}+\mathcal{Q}\mathcal{F}$. To evaluate $P(s)$ and $Q(s)$, the recursion described in the theorem is performed with the precomputed c_{nj} on the set $\Gamma, \delta, s\Gamma, s\delta, \ldots, s^{d-1}\Gamma, s^{d-1}\delta, s^d\Gamma$. If $\Gamma = \mathcal{F}$ and $\delta = 1$, then this is the set (2.29). To generate $P(s)$, the recursion is performed with the fixed c_{nj} on the set with $\Gamma = 0$ and $\delta = -1$. To generate $Q(s)$, the recursion is performed on the set with $\Gamma = 1$ and $\delta = 0$. Generation of $P'(s)$ and $Q'(s)$ is similar.

Algorithm 3 Evaluation of P, Q and their derivatives

```
Data: Coefficients $C_{nj} : n = 1, \cdots, 2d+1$ and $j = 1, 2, 3, 4$ and complex $s$
Result: Evaluation of $P(s)$ and $P'(s)$
Initialization for $P$ and $P'$ :
$A_{-1} = 0 \qquad dA_{-1} = 0$
$A_0 = 0 \qquad dA_0 = 0$
$A_1 = 0 \qquad dA_1 = 0$
$A_2 = -1 \qquad dA_2 = 0$
for $n = 3$ to $2d+1$ do
    $A_n = sA_{n-2} - \sum_{j=1}^{4} C_{nj} A_{n-j}$
    $dA_n = A_{n-2} + sdA_{n-2} - \sum_{j=1}^{4} C_{nj} dA_{n-j}$
end for
$Q(s) = A_{2d+1}, P'(s) = dA_{2d+1}$
Result: Evaluation of $Q(s)$ and $Q'(s)$
Initialization for $Q$ and $Q'$ :
$B_{-1} = 0 \qquad dB_{-1} = 0$
$B_0 = 0 \qquad dB_0 = 0$
$B_1 = 1 \qquad dB_1 = 0$
$B_2 = -C_{21} \qquad dB_2 = 0$
for $n = 3$ to $2d+1$ do
    $B_n = sB_{n-2} - \sum_{j=1}^{4} C_{nj} B_{n-j}$
    $dB_n = B_{n-2} + sdB_{n-2} - \sum_{j=1}^{4} C_{nj} dB_{n-j}$
end for
$Q(s) = B_{2d+1}, Q'(s) = dB_{2d+1}$
```

Newton's methods can then be used to find the roots $\{\beta_j : j = 1, \ldots, n\}$ of $Q(s)$, and then we have the residues $\alpha_j = \left[(s - \beta_j) \, P(s) / \, Q(s) \right] \Big|_{s=\beta_j}$. Then

$$\frac{P(s)}{Q(s)} = \sum_{j=1}^{d} \frac{\alpha_j}{s - \beta_j}.$$

However, due to special form of $Q(s)$ and $P(s)$ with real coefficients, this pole sum will have the form

$$\frac{P(s)}{Q(s)} = \sum_{j=1}^{d_{\text{pair}}} \left(\frac{m_j}{s - k_j} + \frac{\bar{m}_j}{s - \bar{k}_j} \right) + \sum_{j=1}^{d_{\text{sing}}} \frac{\mu_j}{s - \kappa_j},$$

where $d = 2d_{\text{pair}} + d_{\text{sing}}$ and the $\mu_j, \kappa_j \in \mathbb{R}$. We can confirm that such a pole sum will have the parity property. For example,

$$\begin{aligned}
\frac{m_j}{s - k_j} &= \frac{m_j^R + i m_j^I}{\left(x - k_j^R\right) + i \left(y - k_j^I\right)} \\
&= \frac{\left[m_j^R \left(x - k_j^R\right) + m_j^I \left(y - k_j^I\right)\right] + i \left[m_j^I \left(x - k_j^R\right) - m_j^R \left(y - k_j^I\right)\right]}{\left(x - k_j^R\right)^2 + \left(y - k_j^I\right)^2} \\
\frac{\bar{m}_j}{s - \bar{k}_j} &= \frac{m_j^R - i m_j^I}{\left(x - k_j^R\right) + i \left(y + k_j^I\right)} \\
&= \frac{\left[m_j^R \left(x - k_j^R\right) - m_j^I \left(y + k_j^I\right)\right] - i \left[m_j^I \left(x - k_j^R\right) + m_j^R \left(y + k_j^I\right)\right]}{\left(x - k_j^R\right)^2 + \left(y + k_j^I\right)^2}
\end{aligned}$$

Then real part

$$\frac{m_j^R \left(x - k_j^R\right) + m_j^I \left(y - k_j^I\right)}{\left(x - k_j^R\right)^2 + \left(y - k_j^I\right)^2} + \frac{m_j^R \left(x - k_j^R\right) - m_j^I \left(y + k_j^I\right)}{\left(x - k_j^R\right)^2 + \left(y + k_j^I\right)^2}$$

of sum is even, and the imaginary part

$$\frac{m_j^I \left(x - k_j^R\right) - m_j^{R_2} \left(y - k_j^I\right)}{\left(x - k_j^R\right)^2 + \left(y - k_j^I\right)^2} - \frac{m_j^I \left(x - k_j^R\right) + m_j^R \left(y + k_j^I\right)}{\left(x - k_j^R\right)^2 + \left(y + k_j^I\right)^2}$$

of the sum is odd. Also $\mu_j / \left((x - \kappa_j) + iy\right)$ has the parity property. Note if

$$\frac{1}{\sqrt{s^2 + 1}} \simeq \sum_{j=1}^{d_{\text{pair}}} \left(\frac{m_j}{s - k_j} + \frac{\bar{m}_j}{s - \bar{k}_j} \right) + \sum_{j=1}^{d_{\text{sing}}} \frac{\mu_j}{s - \kappa_j},$$

then

$$J_0(t) \simeq \sum_{j=1}^{d_{\text{pair}}} \left(m_j e^{k_j t} + \bar{m}_j e^{\bar{k}_j t} \right) + \sum_{j=1}^{d_{\text{sing}}} \mu_j e^{\kappa_j t},$$

which can also be expressed in terms of sine, cosine, and purely decaying exponentials.

Chapter 4

Numerical Examples

The numerical investigations in this chapter rely on a FORTRAN implementation of the AGH alogorithm, rather than the Matlab implementation. The FORTRAN implementation incorporates the following improvements.

1. Calculations in quadrupole precision.

2. Adaptivity in the numerical quadrature used in the Gram-Schmidt orthogonalizaton.

3. Re-orthgonalization in the Gram-Schmidt process.

Regarding 1, while the numerical approximation of a profile function $f(s)$ as a rational function $P(s)/Q(s)$ is performed in extended precision, we mainly report and work with a double-precision format for the table of pole locations $\{\beta_j\}_{j=1}^{n}$ and residues $\{\alpha_j\}_{j=1}^{n}$ for a pole sum $P(s)/Q(s) \simeq \sum_{j=1}^{d} \alpha_j / (s - \beta_j)$. Such a table is computed "once for all time", and so the cost of its construction is not so important.

Regarding 2, we start with an approximation interval $T^{-1} + i\,[-y_{\max}, y_{\max}]$, which can be halved to $T^{-1} + i[0, y_{max}]$ because of the parity property discussed earlier. For $T = \infty$ this interval is a portion of the imaginary axis in the complex $s = x + iy$ plane. For simplicity in this description, let us just work with the interval $s \in iJ = i\,[0, y_{\max}]$. Next, we test whether $f(s)$ is slowly varying on iJ, for the time-being postponing what "slowly varying" means. If the answer is YES, then we accept iJ as the approximation interval. If the answer is NO, then iJ is divided into two subintervals $i\left[0, \frac{1}{2}y_{\max}\right]$ and $i\left[\frac{1}{2}y_{\max}, y_{\max}\right]$, and $f(s)$ is now tested for slow variation on each subinterval, with subsequent halving if NO is encountered. Continuation of this process yields a binary tree of subintervals, and on the leaves of the tree $f(s)$ is slowly varying. The quadrature in the approximation is based on the introduction of quadrature nodes and weights on the leaves of the binary tree. Although we will not describe the details, in words the test for whether $f(s)$ is slowing varying on a particular interval is as follows. We expand $f(s(y)) \simeq \sum_{k=0}^{K} c_k T_k(\xi(y))$ as a Chebyshev series over the interval, where K is typically 20 to 30 . If $c_K / \left(\sum_{k=0}^{K-1} |c_k| \right)$ is small (relative to some chosen tolerance), meaning the coefficients have decayed sufficiently, then we declare $f(s)$ to be slowly varying. Otherwise, $f(s)$ is not slowly varying and an interval division is necessary.

Regarding 3, while the Gram-Schmidt process produces exactly orthogonal vectors in exact arithmetic, numerically orthogonality can be lost. Numerically, improved orthogonality results from re-orthgonalization (essentially subsequent passes through the Gram-Schmidt process). The FORTRAN code implements re-orthogonalization, and it has a profound effect on the final approximations.

We consider two separate errors associated with approximation of a function $f(s)$:

$$\begin{aligned} \mathrm{err}_{P,Q}(f) &= \sup_{s \in T^{-1} + \mathrm{i}[-y_{\max}, y_{\max}]} \left| \frac{f(s) - P(s)/Q(s)}{f(s)} \right| \\ \mathrm{err}_{\alpha,\beta}(f) &= \sup_{s \in T^{-1} + \mathrm{i}[-y_{\max}, y_{\max}]} \left| \frac{f(s) - \sum_{k=1}^{d} \alpha_k / (s - \beta_k)}{f(s)} \right| . \end{aligned} \tag{4.1}$$

In exact arithmetic

$$P(s)/Q(s) = \sum_{k=1}^{d} \frac{\alpha_k}{s - \beta_k},$$

and so the two errors should be the same. However, the two behave differently in finite precision arithmetic and number storage. Note the evaluation of $P(s)/Q(s)$ relies on the coefficients $\{c_{nj} \,:\, n = 2, \cdots, 2d+1 \text{ and } j = 1, \ldots, \min\{4, n-1\}\}$ described in Theorem 2.2.6, while evaluation of $\sum_{k=1}^{d} \frac{\alpha_k}{s-\beta_k}$ of course relies on the pole locations $\{\beta_k\}_{k=1}^{d}$ and residues $\{\alpha_k\}_{k=1}^{d}$. Note that computation of the pole sum relies on Newton's method to first find the roots of $Q(s)$. Therefore, computation of the (α_k, β_k) relies on the c_{nj}.

In our experiments, both the c_{nj} and the (α_k, β_k) are computed in quadruple precision arithmetic. However, we experiment with storage of the these arrays in both quadruple precision and double precision formats. Our aim is to study the accuracy of the double precision format. Each of the errors (4.1) can then be evaluated with the approximation corresponding to either double precision or quadruple precision formats. For example,

$$\mathrm{err}^{\mathrm{quad}}_{\alpha,\beta}(f) = \sup_{s \in T^{-1} + \mathrm{i}[-y_{\max}, y_{\max}]} \left| \frac{f(s) - \sum_{k=1}^{d} \alpha_k^{\mathrm{quad}} \Big/ \left(s - \beta_k^{\mathrm{quad}}\right)}{f(s)} \right| \tag{4.2}$$

if the (α_k, β_k) are stored in quadruple precision format. Therefore, we consider four errors: $\mathrm{err}^{\mathrm{quad}}_{P,Q}(f), \mathrm{err}^{\mathrm{dble}}_{P,Q}(f), \mathrm{err}^{\mathrm{quad}}_{\alpha,\beta}(f), \mathrm{err}^{\mathrm{dble}}_{\alpha,\beta}(f)$.

4.1 Bessel function $J_1(t)/t$

As an example of the described rational approximation scheme, this subsection constructs approximations of $f(s) = \sqrt{s^2+1} - s$ along $s = x + \mathrm{i}y$ for $x = 1/T > 0$. Note $x > 0$ avoids branch singularities in $f(s)$.

Remark 4.1.1. *$\sqrt{s^2+1} - s$ is the Laplace transform of $J_1(t)/t$. Note, $J_1(t)/t \to 1/2$ as $t \to 0^+$, as shown in [7].*

Owing to the remark, our sum-of-poles approximations for $f(s)$ will correspond to sum-of-exponential approximations of $J_1(t)/t$. This subsection also presents results on these time-domain approximations.

Table 4.1 presents results for a number of sum-of-pole approximations $\widehat{A}(s) = \sum_{k=1}^{d} \alpha_k/(s - \beta_k)$ to $f(s) = \sqrt{s^2+1} - s$. Each block corresponds to a particular choice of T; again the approximations $\widehat{A}(s)$ are constructed along $s = T^{-1} + \mathrm{i}\,[-y_{\max}, y_{\max}]$. All table entries correspond to the choice $y_{\max} = 100$. The $T = 10^8, d = 73$ table entry involved a binary

$T = 10^4$	
d	tol
30	10^{-6}
40	10^{-8}
50	10^{-10}
60	10^{-12}

$T = 10^6$	
d	tol
43	10^{-6}
57	10^{-8}
71	10^{-10}

$T = 10^8$	
d	tol
54	10^{-6}
73	10^{-8}
91	10^{-10}

Table 4.1: Number of poles requires to achieve $\mathrm{err}_{\alpha,\beta}(f) < \mathrm{tol}$

tree of subintervals with 33 levels, with levels added around $y = \pm 1$ in order to resolve the nearby singularities of $f(s)$. The tables list the number of poles needed to achieve

$$\mathrm{err}_{\alpha,\beta}(f) < \mathrm{tol}, \tag{4.3}$$

in terms of the error measure listed in (3.1). For these approximations, the pole locations and residues can be stored in double precision format without affecting the table results. In fact, whether we use $\left(\alpha_k^{\mathrm{dble}}, \beta_k^{\mathrm{dble}}\right)$ or $\left(\alpha_k^{\mathrm{quad}}, \beta_k^{\mathrm{quad}}\right)$ in computing the errors, the results are the same.

Figure 3.1 shows the pole locations $\{\beta_k\}_{k=1}^{d}$ for one approximation, where $d = 54$ and $T = 10^8$. Notice that the location appear to lie on a semicircle of radius 1 , and they cluster near the singularities at $s = \pm \mathrm{i}$; however, in fact all poles lie in the proper left-half s-plane (each location has a strictly negative real part). Our final studies focus on the sum-of-exponential approximation

$$A(t) = \sum_{k=1}^{d} \alpha_k e^{\beta_k t}$$

corresponding to our rational approximation $\widehat{A}(s)$ of $f(s) = \sqrt{s^2+1} - s$. As mentioned, the time domain function $A(t)$ approximates $J_1(t)/t$. Figure 4.2 corresponds to the choices $T = 10^4, d = 60$. The top pane shows that over the t-interval $[0, 300]$ both $f(t)$ and $A(t)$ are identical to the eye. The corresponding pointwise error is shown in the bottom pane. It shows that $A(t)$ agrees with $f(t)$ to nearly double precision accuracy. Figure 3.3 examines the long-time behavior of our approximations. Here the approximations correspond to the $T = 10^8$ results in Table 1, and we represent the approximations with $\left(\alpha_k^{\text{dble}}, \beta_k^{\text{dble}}\right)$ in double-precision format. The figure depicts the pointwise error $|A(t) - J_1(t)/t|$ masked by the function $max(1, \sqrt{\pi t/2})$. Enhancement of the error by this factor incorporates the asymptotic decay of $J_1(t)$. If instead of the adopted mask, we were to use $max(1, \sqrt{\pi t^3/2})$, then the scaled errors grow linearly with time. Note that $t^{-3/2}$ captures the full decay of the envelope for $J_1(t)/t$.

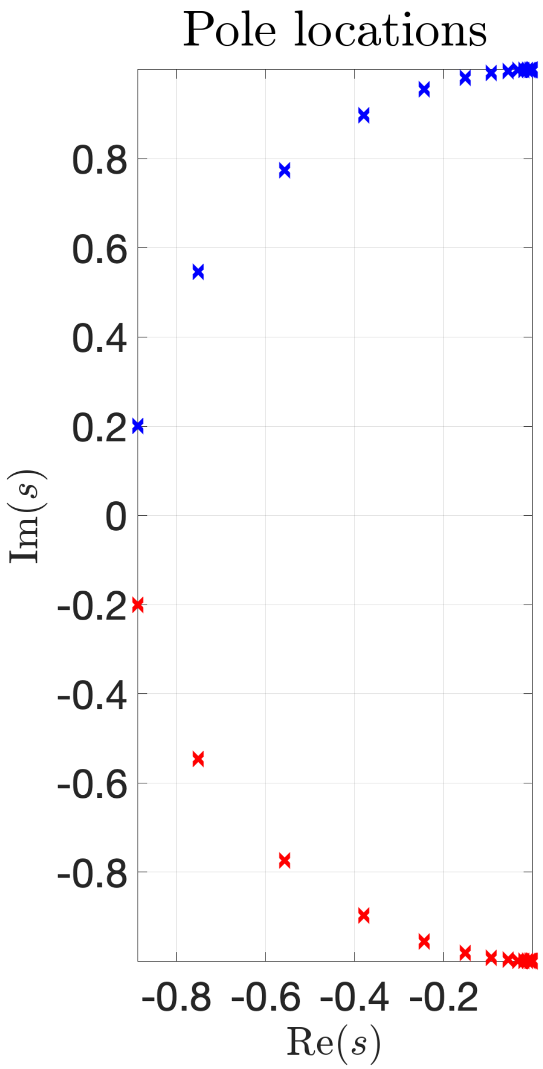

Figure 4.1: Plots the location $\{\beta_k\}_{k=1}^{d}$ for $d = 54$, $T = 10^8$ approximation of $\sqrt{s^2+1} - s$. Notice that the locations accumulate near $s = \pm i$, which are the singularities of $f(s)$.

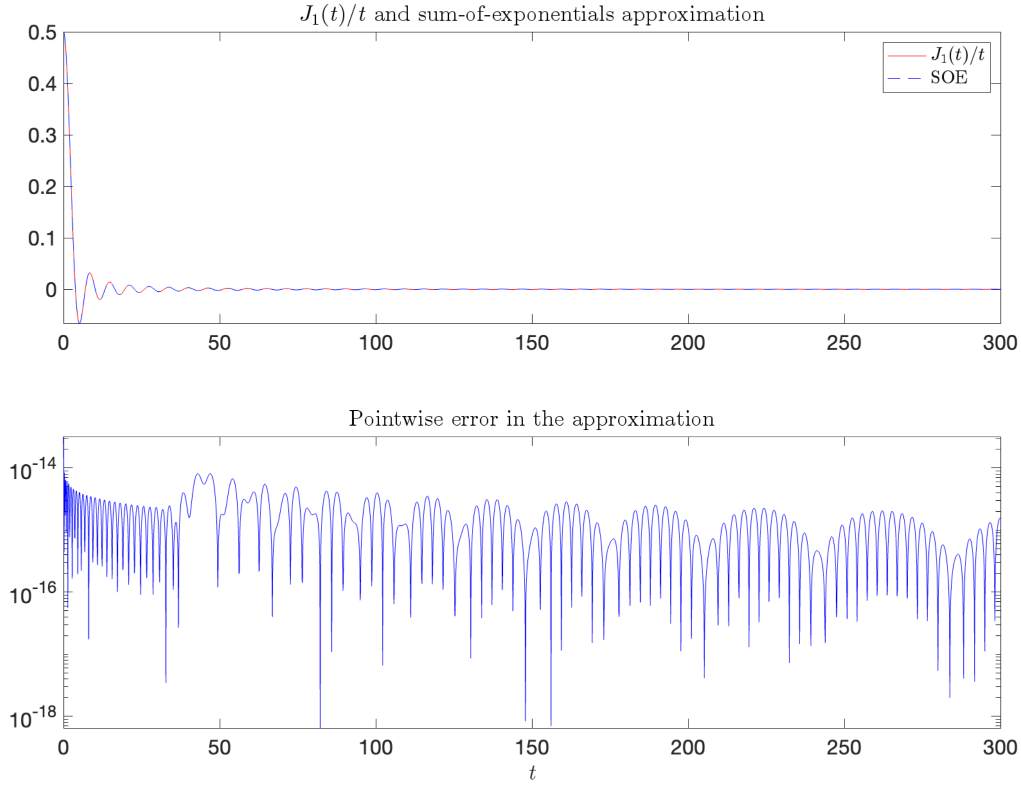

Figure 4.2: Corresponds to the choices $T = 10^4, d = 60$.

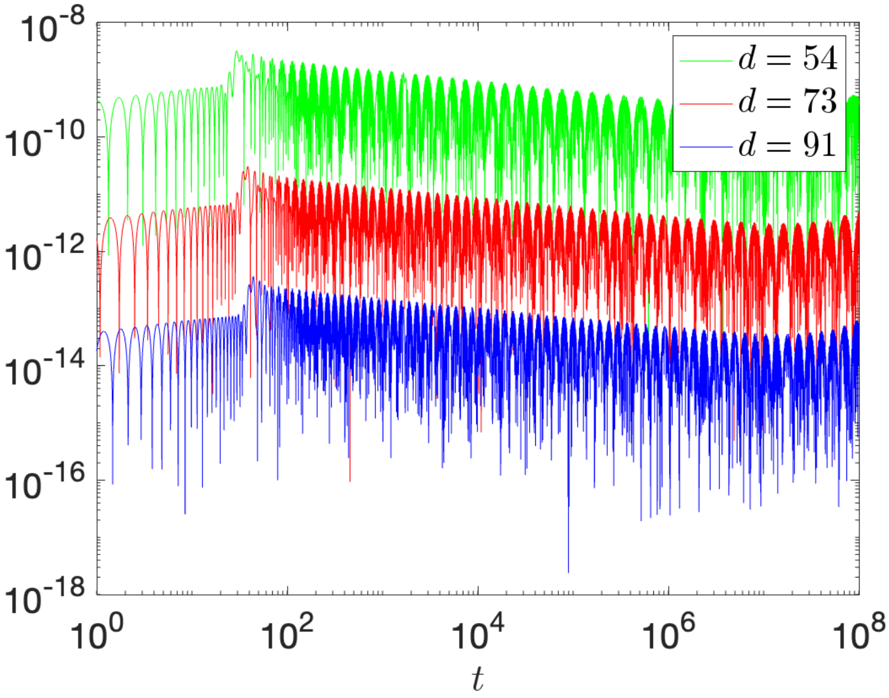

Figure 4.3: Error results long-time test.

4.2 Near-to-far propagation kernel

In this section, we will consider

$$f(s) = \widehat{\Phi}_l(s, r_1, r_2) = -1 + \frac{W_l\,(sr_2)}{W_l(sr_1)}, \tag{4.4}$$

for $l = 64$ and $r_2 = 240$. According to [4], let us consider $r_1 = 15, 30, 60, 120$. The exact kernel $\widehat{\Phi}_{64}(s, r_1, 240)$ is a sum of 64 poles, but it is known that in modulus the largest residue grows exponentially with l. For $l = 64$ the exact pole sum cannot be accurately used in double precision arithmetic. Here, we evaluate the kernel along $s = iy$ via an alternative method, and then approximate the kernel by a smaller pole sum.

We now describe our alternate approach for evaluation of the profiles $\mathrm{Re}\,\widehat{\Phi}_l\,(\mathrm{i}y, r_1, r_2)$ and $\mathrm{Im}\,\widehat{\Phi}_l\,(\mathrm{i}y, r_1, r_2)$ for $y \in \mathbb{R}$. As mentioned, Ref. [3] has described stable evaluation of the time-domain kernel $\Phi_l\left(t, r_1, r_2\right)$.

Our approach is based on the following expression for an NtF kernel:

$$\widehat{\Phi}_l\left(s, r_1, r_2\right) = -1 + \exp\left[\int_{r_1}^{r_2} \frac{\widehat{\Omega}_l(s,\eta)}{\eta} d\eta\right], \tag{4.5}$$

and

$$\exp\left[\int_{r_1}^{r_2} \frac{\widehat{\Omega}_l(s,\eta)}{\eta} d\eta\right] = W_l\left(sr_2\right)/W_l\left(sr_1\right), \tag{4.6}$$

where we have introduced an auxiliary function

$$\widehat{\Omega}_l(s,r) \equiv sr\frac{W_l'(sr)}{W_l(sr)} = \sum_{k=1}^{l} \frac{b_{lk}/r}{s - b_{lk}/r}, \tag{4.7}$$

with the prime indicating differentiation in argument. With Steed's algorithm [9] the kernel $\widehat{\Omega}_l(s,r)$ is accurately computed via the known continued fraction expansion

$$z\frac{W_l'(z)}{W_l(z)} = -\frac{l(l+1)}{2(z+1)+}\frac{(l-1)(l+2)}{2(z+2)+}\cdots\frac{2(2l-1)}{2(z+l-1)+}\frac{2l}{2(z+l)}. \tag{4.8}$$

This formula follows from recurrence relations obeyed by MacDonald functions [10]. Given the ability to compute $\widehat{\Omega}_l(\mathrm{i}y, r)$, computation of (4.5) can be carried out using numerical quadrature. Due to the structure of $\mathrm{Re}\,\widehat{\Omega}_l(\mathrm{i}y, r)$ and $\mathrm{Im}\,\widehat{\Omega}_l(\mathrm{i}y, r)$, the numerical integrations involve no cancellation errors (i.e. the sums involve only positive or negative values at each fixed y_j grid point).

We could instead work with the equivalent kernel $\Phi_{64}(s, 1, 16)$, but focus on $r_1 = 15$ and $r_2 = 240$ as this kernel appears in [4]. In fact, $\widehat{\Phi}_{64}(s, 15, 240)$ is exactly a sum of 64 poles, but its known that the largest residue (in modulus) of $\widehat{\Phi}_l\left(s, r_1, r_2\right)$ grows exponentially with l. Already for $l = 64$, in modulus the maximum residue is about 10^{15} whereas the smallest one is about 20; see [4]. Double precision storage of the pole locations and residues results in loss of precision. Nonetheless, here we approximate $\widehat{\Phi}_{64}(s, 15, 240)$ as a smaller pole sum, with the goal of finding the best double-precision storage pole-sum approximation.

For a given r_1, say $r_1 = 15$, we shall consider sum-of-poles approximations to $\widehat{\Phi}_{64}(s, r_1, 240)$, for sequence of d values, starting around $d = 30$ and increasing. These sum-of-poles approximations stem from *double precision* formatted tables (the Matlab-compatible tables outputted from the Fortran code are in this format). The key question is what is the optimal value of d for the best approximation to the profiles. If the computer performed arithmetic exactly, the error would go down as d was increases up to $d = 64$, at which point the error would be zero. This does not happen numerically. For $d = 46$ the pole sum error is at a minimum. This is the best practical sum-of-poles approximations to the profiles (accurately generated with (4.5)).

Figure 3.4 depicts the kernel $\hat{\Phi}_{64}(s, 15, 240)$.

d	$\mathrm{err}^{\mathrm{quad}}_{P,Q}$	$\mathrm{err}^{\mathrm{dble}}_{P,Q}$	$\mathrm{err}^{\mathrm{quad}}_{\alpha,\beta}$	$\mathrm{err}^{\mathrm{dble}}_{\alpha,\beta}$
30	$1.5886\mathrm{E}-07$	$1.5886\mathrm{E}-07$	$1.5886\mathrm{E}-07$	$1.5888\mathrm{E}-07$
38	$1.3196\mathrm{E}-09$	$1.3195\mathrm{E}-09$	$1.3196\mathrm{E}-09$	$1.3236\mathrm{E}-09$
46	$3.3095\mathrm{E}-12$	$3.0465\mathrm{E}-12$	$3.3095\mathrm{E}-12$	$4.9992\mathrm{E}-10$
54	$1.9925\mathrm{E}-14$	$3.4432\mathrm{E}-13$	$2.0070\mathrm{E}-14$	$8.6415\mathrm{E}-10$
62	$3.6985\mathrm{E}-16$	$2.4559\mathrm{E}-12$	$2.1901\mathrm{E}-15$	$2.1468\mathrm{E}-09$
64	$4.3222\mathrm{E}-17$	$2.9317\mathrm{E}-12$	$7.3939\mathrm{E}-15$	$1.4739\mathrm{E}-08$

Table 4.2: Errors in rational approximations of $\widehat{\Phi}_{64}(s, 15, 240)$.

Table 4.2 shows the results; for these errors $T^{-1} = 0$. Notice that the fundamental $P(s)/Q(s)$ approximation, relying on quadruple representation of the coefficients c_{nj}, becomes more accurate as d is increased. However, the other representations degrade as d increases. The pole-sum approximation relying on double-precision representation of the poles locations β_k^{dble} and residues α_k^{dble} clearly degrades most rapidly.

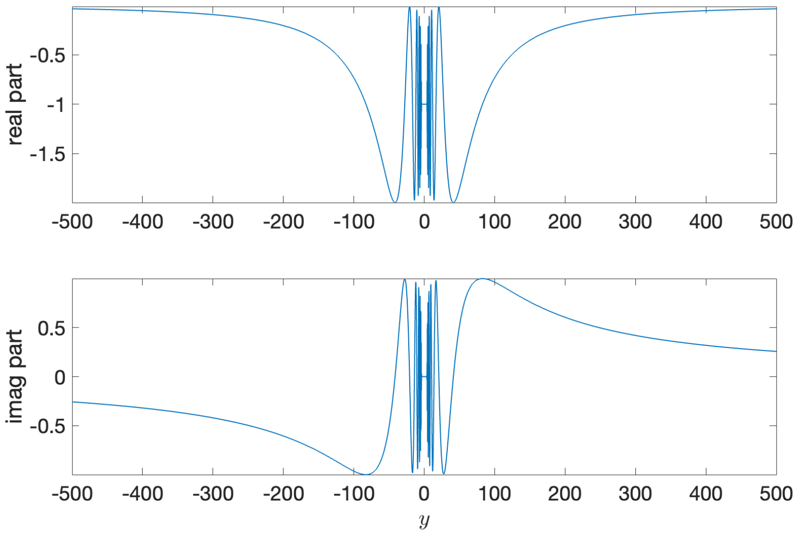

Figure 4.4: Depicts the real and imaginary profile of the kernel $\hat{\Phi}_{64}(s, 15, 240)$ along $s = \mathrm{i}y$.

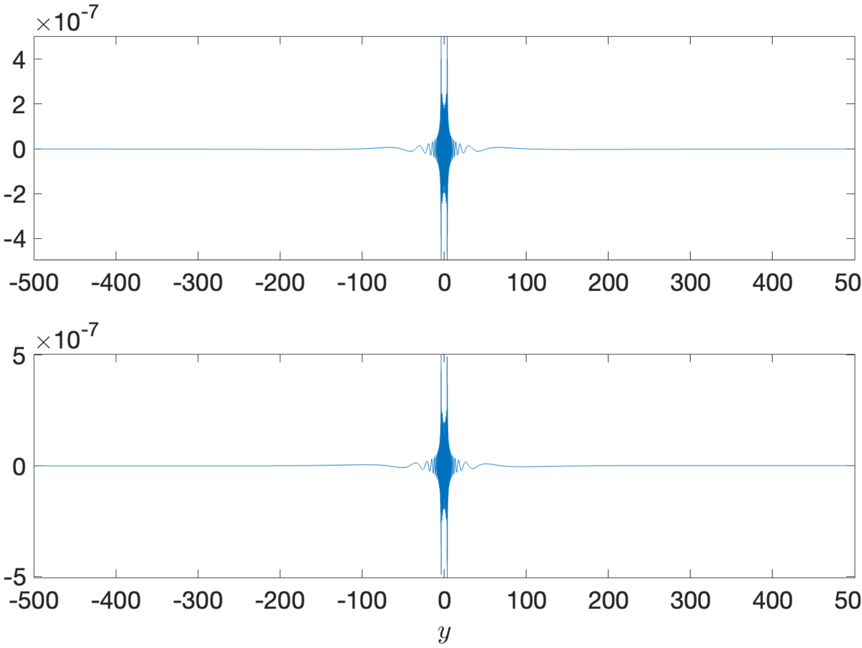

Figure 4.5: Shows the double-precision pole-sum for a $d = 28$ approximation.

Chapter 5

Conclusion

Lastly, we will wrap up by emphasizing a few of this thesis's most crucial points. Chapter 1 described multipole solutions to the radial wave equation (RWE) which arises from $3+1$ wave equation upon spherical harmonic transformation. In terms of these multipole solutions, the solution to the 3+1 wave equation is the expansion

$$\psi(t,x,y,z) = \psi(t, r\sin\theta\cos\phi, r\sin\theta\sin\phi, r\cos\theta) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l}\frac{1}{r}\Psi_{lm}(t,r)Y_{lm}(\theta,\phi).$$

By claim 2.1.1, lemma 2.1.2 and 2.1.3, we derived an outgoing multipole solution $\Psi_l(t,r)$ to the radial wave equation (RWE). The encountered formulas do not feature m, and so this index has been suppressed. We examined the structure of near-to-far propagation kernels used to map the solution $\Psi_\ell(t,r_1)$ at r_1 to the one $\Psi_\ell(t,r_2)$ at $r_2 > r_1$. These kernels have the form

$$\Phi_l(t_1,r_1,r_2) = \sum_{j=1}^{l} a_{lj}(r_1,r_2)e^{(b_{lj}/r_1)t},$$

where [3]

$$a_{ij}(r_1,r_2) = \frac{W_l(b_{lj}r_2r_1^{-1})}{r_1W_l'(b_{lj})}.$$

Here the $\{b_{lj} : j = 1,\dots,l\}$ are the zeros of the MacDonald function $K_{l+1/2}(z)$ which lie in the left-half plane.

Chapter 2 has considered the classical problem of polynomial approximation as a warm-up for rational approximation via the Alpert-Greengard-Hagstrom algorithm [5]. Chapter 2 considers approximation of a function $f(s)$, where $s = x + \mathrm{i}y$. For x fixed, we view $f(s)$ as a function of y, assuming that the real part of f is even in y and the imaginary part of f is odd in y (the "parity property"). The AGH algorithm described in Chapter 3 then allows for an approximation of the form

$$f(s) \simeq \frac{P(s)}{Q(s)} = \sum_{j=1}^{d} \frac{\alpha_j}{s-\beta_j} = \sum_{j=1}^{d_{\text{pair}}} \left(\frac{m_j}{s-k_j} + \frac{\bar{m}_j}{s-\bar{k}_j} \right) + \sum_{j=1}^{d_{\text{sing}}} \frac{\mu_j}{s-\kappa_j},$$

where $d = 2d_{\text{pair}} + d_{\text{sing}}$ and $\mu_j, \kappa_j \in \mathbb{R}$. We presented a Matlab code/implementation of the approximation scheme (collected in the Appendix).

Chapter 3 considered numerical examples. First, it studied approximation of $f(s) = \sqrt{s^2+1} - s$ which is the Laplace transform of $J_1(t)/t$. This example served to confirm that a Fortran implementation of the AGH algorithm works, and to understand the limits of this implementation. The second half of Chapter 3 has examined approximation of $\widehat{\Phi}_{64}(s, r_1, r_2)$ as a *smaller* pole sum of d terms, that is as

$$\widehat{\Phi}_{64}(s, r_1, r_2) \simeq \sum_{k=1}^{d} \frac{\alpha_k}{s-\beta_k},$$

where $d \leq 64$. Due to finite-precision effects, these smaller poles sum may in practice yield more accurate representations of $\hat{\Phi}_{64}(s, r_1, r_2)$ than is achievable by a full 64-term pole sum. The origin of this phenomena is that, as first shown in [3], the residues a_{lj} grow exponentially with l. Future work should address the following question. For a desired accuracy, say single-precision accuracy, what is the largest possible l-value such that $\hat{\Phi}_l(s, r_1, r_2)$ admits a d-pole approximation stored in double-precision format. This largest value of l is beyond 64, but we believe not much beyond.

References

[1] C. Wilcox, "The initial-boundary value problem for the wave equation in an exterior domain with spherical boundary," *Notices Amer. Math. Soc*, vol. 6, pp. 869–870, 1959.

[2] T. Tokita, "Exponential decay of solutions for the wave equation in the exterior domain with spherical boundary," *Journal of Mathematics of Kyoto University*, vol. 12, no. 2, pp. 413–430, 1972.

[3] L. Greengard, T. Hagstrom, and S. Jiang, "The solution of the scalar wave equation in the exterior of a sphere," *Journal of Computational Physics*, vol. 274, pp. 191–207, 2014.

[4] S. E. Field and S. R. Lau, "Fast evaluation of far-field signals for time-domain wave propagation," *Journal of Scientific Computing*, vol. 64, no. 3, pp. 647–669, 2015.

[5] B. Alpert, L. Greengard, and T. Hagstrom, "Rapid evaluation of nonreflecting boundary kernels for time-domain wave propagation," *SIAM Journal on Numerical Analysis*, vol. 37, no. 4, pp. 1138–1164, 2000.

[6] A. G. Benedict, S. E. Field, and S. R. Lau, "Fast evaluation of asymptotic waveforms from gravitational perturbations," *Classical and Quantum Gravity*, vol. 30, no. 5, p. 055015, 2013.

[7] M. Abramowitz and I. Stegun, "Handbook of mathematical functions with formulas, graphs, and mathematical tables" edited by dover publications," *Inc., New York, Ninth Printing*, 1970.

[8] A. Ahmed, "Near-to-far field signal propagation for the wave and maxwell equations, master's thesis, university of new mexico," 2019.

[9] I. J. Thompson and A. R. Barnett, "Coulomb and bessel functions of complex arguments and order," *Journal of Computational Physics*, vol. 64, no. 2, pp. 490–509, 1986.

[10] G. N. Watson, *A treatise on the theory of Bessel functions*, vol. 2. Second edition 1944.