Numerical Methods for Wave Phenomena

by

Fortino Garcia

B.A., Rice University, 2015

M.S., University of Colorado Boulder, 2018

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Applied Mathematics

2021

Committee Members:

Daniel Appelö, Chair

Prof. Adrianna Gillman

Prof. Stephen Becker

Prof. Olof Runborg

Dr. Anders Petersson

Garcia, Fortino (Ph.D., Applied Mathematics)

Numerical Methods for Wave Phenomena

Thesis directed by Prof. Daniel Appelö

This dissertation describes numerical methods for wave phenomena and is divided into two main sections. The first concerns a new time-domain approach to solving the Helmholtz equation. The second concerns numerical methods for the optimal control of closed quantum systems.

The efficient solution of the Helmholtz equation is an active area of research. Traditionally, many methods in the literature take the approach of solving the Helmholtz equation "directly". By directly we mean solving the Helmholtz equation in the frequency domain, whether by a direct discretization of the PDE via finite differences/elements or by integral equation methods. An alternative approach is to instead solve the Helmholtz equation by seeking time-harmonic solutions in the time-domain. In this thesis we present the WaveHoltz iteration, which is a fixed-point iteration for solving the Helmholtz equation by instead solving a sequence of wave equations. We demonstrate that WaveHoltz is amenable to acceleration via Krylov subspace methods. Moreover we show that WaveHoltz is simple to implement, inherits the memory-leanness and scalability of the underlying wave equation discretization, and that it is possible to remove time-discretization errors from the WaveHoltz solution.

The second part of the thesis introduces tools for devising optimal controls to realize logic gates in closed quantum systems. We motivate a novel approximation of control functions via B-spline wavelets with carrier waves that are specifically constructed to trigger the transition frequencies of a quantum system. Using the symplectic and time-reversible Störmer-Verlet scheme, we take a "discretize-then-optimize" approach to determine a corresponding adjoint partitioned Runge Kutta scheme. This allows the computation of **exact** discrete gradients for the quantum optimal control problem. Finally, we outline a submitted solution to the IBM SWAP Gate Challenge using these methods.

Dedication

To my mom and my sisters for their unconditional love and support.

Acknowledgements

First and foremost, I would like to thank my advisor Daniel Appelö for his support and encouragement. I hope to be just as patient and confident in every student I work with in the future as you have been with me. Thank you to both Olof Runborg and Anders Petersson for hosting me at KTH and Lawrence Livermore National Lab, respectively, as well as for continued mentorship and guidance from which I have learned an immense amount. I would also like to thank Adrianna Gillman, without whom I would not have heard of the applied math department at Boulder or the Helmholtz equation.

I don't believe I would be successful without the support of the many wonderful friends I have made in the applied math department and in Boulder. Thank you to Caleb, Allen, Alec, Minah, Mingyu, David, Mike, Ryan, Liam, Lyndsey, Lauren, and Roxie. A very special thanks to Caleb, Sara, and Lyndsey for the late game nights that kept me sane during the worst of the pandemic. I would also like to thank all of the graduate students in APPM for letting me be a part of such a kind and supportive environment during this time of my life.

To my friends from my time in Texas, including Taylor, Matthew, Neel, Nick, Luis, Sam, Eric, Michael, and all of the members of Steve Cox's Beard, thank you for your friendship all of these years.

Finally, I would like to thank the NSF for the support provided by NSF Grant DMS-1913076, as well as to STINT initiation grant IB2019–8154.

Contents

Chapter

1 Introduction 1

1.1 Outline of chapters 2

2 WaveHoltz: Iterative Solution of the Helmholtz Equation via the Wave Equation 4

2.1 WaveHoltz: A New Method for Designing Scalable Parallel Helmholtz Solvers 9

2.1.1 Iteration for the Energy Conserving Case 11

2.1.2 Analysis of the Discrete Iteration 18

2.1.3 Tunable Filters 22

2.1.4 Multiple Frequencies in One Solve 23

2.1.5 WaveHoltz Iteration for Impedance Boundary Conditions 23

2.2 Wave Equation Solvers 24

2.2.1 The Energy Based Discontinuous Galerkin Method 24

2.2.2 Finite Difference Discretizations 26

2.2.3 Time Discretization 27

2.3 Numerical Examples 28

2.3.1 Examples in One Dimension 28

2.3.2 Problems in Two Dimensions 34

2.3.3 Problems in Three Dimensions 43

2.4 Summary and Future Work 46

3 Analysis of an Iterative Solution of the Helmholtz Equation via the Wave Equation for Impedance Boundary Conditions 47

3.1 The General Iteration . . . 51

3.1.1 Iteration for the Energy Conserving Case for the General WaveHoltz Iteration 52

3.1.2 Convergence in the Non-Energy Conserving Case . . . 60

3.2 Damped Wave/Helmholtz Equation . . . 66

3.3 Analysis of Higher Order Time Stepping Schemes for the Discrete Iteration . . . 68

3.4 Wave Equation Solvers . . . 72

3.4.1 The Energy Based Discontinuous Galerkin Method . . . 72

3.4.2 Symmetric Interior Penalty Discontinuous Galerkin Method . . . 74

3.4.3 Finite Difference Discretizations . . . 74

3.4.4 Time Discretization . . . 75

3.5 Numerical Examples . . . 75

3.5.1 Examples in One Dimension . . . 75

3.5.2 Examples in Two Dimensions . . . 82

3.6 Summary and Future Work . . . 85

4 El WaveHoltz Method 87

4.1 Governing Equations . . . 92

4.1.1 The Time Harmonic Elastic Wave Equation . . . 92

4.1.2 The Elastic Wave Equation . . . 93

4.2 The El WaveHoltz Iteration . . . 94

4.3 Numerical Methods and Discrete Analysis . . . 96

4.3.1 El WaveHoltz by Finite Differences . . . 96

4.3.2 El WaveHoltz by Symmetric Interior Penalty Discontinuous Galerkin Method 99

4.3.3 Explicit Time-Corrected Scheme . . . 100

4.3.4 Implicit Time-Corrected Scheme . . . 101

4.3.5 Krylov Solution of the El WaveHoltz Iteration . . . 104
4.4 Numerical Experiments . . . 106
4.4.1 Accuracy of the Finite Difference Method . . . 106
4.4.2 Verification of Corrected Time-Steppers . . . 107
4.4.3 Accuracy of the Symmetric Interior Penalty Discontinuous Galerkin Method 109
4.4.4 Effects on Number of Iterations from Number of Periods and Accuracy . . . 109
4.4.5 Iteration Count as a Function of Frequency for Rectangles and Annular Sectors 112
4.4.6 Effect of Boundary Conditions in A Cube . . . 113
4.4.7 Iteration Count as a Function of Wave Speed Ratio . . . 114
4.4.8 Comparison of Explicit and Implicit El WaveHoltz with Direct Discretization of Elastic Helmholtz . . . 115
4.4.9 Materials with Spatially Varying Properties . . . 120
4.4.10 Vibrations of a Toroidal Shell . . . 121
4.5 Conclusion . . . 122

5 Optimal Control of Closed Quantum Systems via B-Splines with Carrier Waves 124
5.1 Hamiltonian model . . . 129
5.1.1 Rotating wave approximation . . . 130
5.1.2 Resonant frequencies . . . 131
5.2 Quadratic B-splines with carrier waves . . . 133
5.3 Real-valued formulation . . . 134
5.3.1 Time integration . . . 135
5.3.2 Time step restrictions for accuracy and stability . . . 136
5.4 Discretizing the objective function and its gradient . . . 138
5.4.1 Discretizing the objective function . . . 138
5.4.2 The discrete adjoint approach . . . 140
5.5 Numerical optimization . . . 143

5.5.1 A CNOT gate on a single qudit with guard levels . . . 144
5.5.2 The Hessian of the objective function . . . 148
5.5.3 Risk-neutral controls . . . 150
5.6 Comparing Juqbox with QuTiP/pulse_optim and Grape-TF . . . 152
5.6.1 Setup of simulation codes . . . 154
5.6.2 Numerical results . . . 155
5.7 Conclusions . . . 160

6 IBM Open Science Prize – SWAP Gate Challenge 162
6.1 Hamiltonian model . . . 163
6.2 Optimal control with Juqbox.jl and Quandary . . . 165
6.2.1 Open system optimal control . . . 167
6.3 Rabi pulse calibrations . . . 168
6.4 Gaussian square and DRAG pulses in Qiskit . . . 169
6.5 Converting Qiskit pulses to B-splines with carrier waves . . . 170
6.6 Reverse model calibration using X- and Cx-gates . . . 171
6.6.1 X-gates . . . 172
6.6.2 Calibrating a cross-talk model using the Cx gates . . . 174
6.7 Implementation of custom pulses in Qiskit . . . 176
6.8 Randomized benchmarking results . . . 178
6.9 Conclusion . . . 182

7 Conclusion 184

Bibliography 187

Appendix

.1 Proof of Lemma 2.1.1 . . . 194

.2 Verification of Discrete Solution . . . 196
.3 Proof of Lemma 2.1.5 . . . 196
.4 Proof of Lemma 3.1.1 . . . 198
.5 Wave Equation Extension . . . 201
.6 Verification of Discrete Solution . . . 204
.7 Well-definedness of modified frequencies . . . 205
.8 Error in discrete Helmholtz frequency . . . 207
.9 Verification of Discrete Solution . . . 208
.10 Time-step Restriction . . . 208
.11 Motivation of Conjecture 1 . . . 209
.12 Composite quantum systems and essential states . . . 212
.13 The Hamiltonian in a rotating frame of reference . . . 214
.14 Conditions for resonance . . . 215
.15 Derivation of the discrete adjoint scheme . . . 217
.16 Proof of Corollary 1 . . . 222
.17 Computing the gradient of the discrete objective function . . . 223

Tables

Table

2.1 Maximum error for various combinations of boundary conditions and methods. . . . 29

4.1 L_1, L_2 and L_∞ errors of the computed solution with corresponding estimated rates of convergence. 106

4.2 Estimated rates of convergence for the spatial discretization. 109

4.3 The table displays the number of iterations required and the efficiency of the longer times to reduce the relative residual by a factor 10^{-10} for the two cases (described in the text). 111

4.4 The effect on iteration count depending on different combinations of λ and μ. . . . 114

4.5 Comparison of the number of iterations for the three different methods. 115

4.6 The number of right hand side evaluations (estimated) for the three different methods. The top four rows display the actual number of right hand side evaluations and the rows below indicate how many times more the HH and IWH method evaluates the right hand side. An infinity sign indicates that the computation did not converge. 119

4.7 The table reports how many times longer a computation with the HH and IWH method takes compared to the explicit WH method. 120

5.1 The Frobenius norm of the symmetric and asymmetric parts of the approximate Hessian, L, for the case $\alpha_{max}/2\pi = 3.0$ MHz. 149

196 5.2 Gate duration, number of time steps (M) and total number of control parameters
197 (D) in the $|0\rangle \leftrightarrow |d\rangle$ SWAP gate simulations. The number of time steps and control
198 parameters are the same for pulse_optim and Grape-TF. 156
199 5.3 QuTiP/pulse_optim results for $|0\rangle \leftrightarrow |d\rangle$ SWAP gates. Note the larger infidelity and
200 guard state population for $d = 6$. 157
201 5.4 Grape-TF results for $|0\rangle \leftrightarrow |d\rangle$ SWAP gates. Note the very large infidelity for $d = 6$.
202 These simulations used two NVIDIA P-100 GPUs to accelerate TensorFlow. 157
203 5.5 Juqbox results for $|0\rangle \leftrightarrow |d\rangle$ SWAP gates. 157

Figures

Figure

2.1 The filter transfer function β for $\omega = 10$. 14

2.2 Convergence of the residual for the plain WaveHoltz iteration and its accelerated versions using LSQR, QMR, CG and GMRES. The titles of the figures indicate the boundary conditions used to the left and right, e.g. D-N means Dirichlet on the left and Neumann on the right. 30

2.3 Left: Number of iterations divided by ω as a function of ω for different boundary conditions. Middle and right: Zoom in around a resonance for the Dirichlet problem when using Krylov acceleration (middle) and when using WHI (right). 31

2.4 Left: Convergence history of the near resonant frequency 4.1π for the WaveHoltz filter and a tunable filter, and that of the frequency 1.5π for reference. Middle: The error between successive WaveHoltz iterates with the usual WaveHoltz filter. Right: Convergence of the solution for the CG accelerated WaveHoltz iteration with a point forcing. 33

2.5 (Left) The usual WaveHoltz filter (in blue) and updated tunable filter (in red). (Right) Closeup of both the usual WaveHoltz filter and the updated tunable filter near the resonant frequency 4π. 33

2.6 Typical solutions computed with the GMRES accelerated WHI at $\omega = 77.5$. The thick lines indicate Dirichlet boundary conditions. 35

2.7 To the left: number of iterations as a function of frequency to reduce the relative residual below 10^{-7} for problems with no trapped waves. Middle: the same but for problems with trapped waves and for the interior problem. Both are with the GMRES accelerated WHI . To the right: Residuals for the GMRES accelerated WHI , the CG accelerated WHI and for GMRES solution of the directly discretized Helmholtz problem. 36

2.8 Left: the speed of sound (squared) used in example 2.3.2.2. Red indicates a rigid wall and black indicates open walls. Middle: Number of iterations as a function of frequency. Right: Compute time normalized by the frequency times the number of degrees of freedom. 38

2.9 The magnitude of the Helmholtz solution for, from left to right, $\omega = 25\pi, 50\pi$ and 100π. 38

2.10 The maximum error GMRES residuals as a function of number of iterations for four different mesh sizes. The rates of convergence agree with the order of the method. . 39

2.11 Displayed is the base 10 logarithm of the magnitude of the Helmholtz solution ($\log_{10}|u|$) caused by a point source near the surface. The results are, from top to botton, for $\omega = 800, 400$ and 200. 41

2.12 Zoom in of the base 10 logarithm of the magnitude of the Helmholtz solution ($\log_{10}|u|$) caused by a point source near the surface. The results are, from left to right, for $\omega = 800, 400$ and 200. 42

2.13 Computation of three Helmholtz problems by one solve. The frequencies are $\omega = 15, 30$ and 60. The material model is also displayed, red is $c^2 = 1$ and dark blue is $c^2 = 0.1$. 42

2.14 To the left: number of iterations as a function of frequency to reduce the relative residual below $5 \cdot 10^{-5}$ for problems with no trapped waves. Here WHI is accelerated by TFQMR. To the right: the same but for the interior problem. Here WHI is accelerated with either CG or TFQMR. 43

2.15 Displayed is the base 10 logarithm of the magnitude of the Helmholtz solution ($\log_{10}|u|$) caused by a point source near the surface for $\omega = 200$ (left) and $\omega = 300$ (right) at the slice $x = 0.1$. 45

3.1 (Left, Middle) The initial conditions v_0 and v_1 for a Helmholtz frequency of $\omega = 10\pi$. (Right) The estimate of the quantity $1 - \|\mathcal{S}\|$ with increasing Helmholtz frequency ω. 77

3.2 The norm of WaveHoltz iterates for increasing Helmholtz frequencies of $\omega = 10\pi, 40\pi,$ and 70π for the adversarial example of Figure 3.1. 78

3.3 The estimate of the quantity $1 - \|\mathcal{S}\|$ with increasing Helmholtz frequency ω for a radially symmetric initial condition. 79

3.4 Convergence of the discrete WaveHoltz solution to the true solution of the discrete Helmholtz problem with fixed spatial discretization. Solid lines indicate relative errors between discrete solutions. The blue and yellow solid lines indicate relative errors between the usual discrete WaveHoltz solution and the true solution, and the red and purple solid lines indicate relative errors for the frequency corrected solution. 80

3.5 Number of iterations as a function of ω for different boundary conditions and damping parameters. Left: $\eta = 1/2\omega$, Middle: $\eta = 1/2$, Right: $\eta = \omega/2$. In the above legends each entry is made up of a two letter string where the first letter indicates the boundary condition on the left at $x = -6$, and the second letter indicates the boundary condition on the right at $x = 6$. Here D indicates Dirichlet, N indicates Neumann, and I indicates impedance/Sommerfeld conditions. 82

3.6 Computational domain where the mesh in blue corresponds to a wavespeed of $c = 2100$, the mesh in green corresponds to a speed of $c = 1000$, and the magenta mesh with $c = 2900$. The solid black line is not physical and is meant to more easily distinguish between regions. 83

3.7 Number of iterations to reach a GMRES tolerance of 10^{-10} for the wedge problem in 2D with all Neumann or all impedance boundary conditions. 84

3.8 In the above we plot the $\log_{10}$ of the absolute value of the real part of the Helmholtz solution with frequency $\omega = 40\pi$ for (Left) damping parameter $\eta = 20\pi$ and (Right) no damping. 85

4.1 (Left) Convergence of the discrete WaveHoltz solution to the true solution of the discrete Helmholtz problem. (Right) Convergence of the discrete WaveHoltz solution to the true solution of the discrete Helmholtz problem for a manufactured solution. 108

4.2 From top to bottom: displacement magnitude, σ_{xx}, σ_{xy} and σ_{yy}. The domain is $[0, 8] \times [0, 1]$ and the color scales are $[0, 8]$, $[-50, 50]$, $[-15, 15]$ and $[-40, 40]$ respectively. 110

4.3 The number of iterations as a function of frequency to reach convergence for (Left) a rectangle, quarter circle and half circle, and (Right) the unit cube with Dirichlet or free surface conditions. 113

4.4 From top left to bottom right: displacement magnitude, σ_{xy}, σ_{xx} and σ_{yy}. 116

4.5 The $\log_{10}$ of the magnitude of the displacements for the CG accelerated solution of WH for the inclusion problem using sixth order polynomials within each element. (Left) Solution using a grid resolution of at least one element per wavelength, and (Right) two elements per wavelength. 121

4.6 The solution in the toroidal shell for (Left) $\omega = 5.1234$, and (Right) $\omega = 10.2468$. The projection onto the xy-plane is the magnitude of the displacement on the outermost free surface $r = 2$. In black we display the (scaled) displaced mesh for $r = 2$. 122

5.1 The real part of a quadratic B-spline control function, with zero carrier frequency (dashed black). The solid colored lines are the individual B-spline wavelets. 134

5.2 Convergence of the IPOPT iteration for the CNOT gate with the parameter constraint $\|\boldsymbol{\alpha}\|_\infty \leq \alpha_{max}$. Here, $\alpha_{max}/2\pi = 4$ MHz (left) and $\alpha_{max}/2\pi = 3$ MHz (right). 146

5.3 The population of the "forbidden" state $|5\rangle$ as function of time for the four initial conditions of the CNOT gate. Here, $\alpha_{max}/2\pi = 3$ MHz. 146
5.4 The rotating frame control functions $p(t)$ (blue) and $q(t)$ (orange) for realizing a CNOT gate with $D_1 = 10$ basis function per carrier wave and three carrier wave frequencies. Here, $\alpha_{max}/2\pi = 3$ MHz. 147
5.5 The population of the states $|0\rangle$ (blue), $|1\rangle$ (orange), $|2\rangle$ (green) and $|3\rangle$ (purple), as function of time, for each initial condition of the CNOT gate. Here, $\alpha_{max}/2\pi = 3$ MHz. 148
5.6 The eigenvalues of the symmetric part of the approximate Hessian, $0.5(L+L^T)$, evaluated at the optima for the parameter thresholds $\alpha_{max}/2\pi = 4$ MHz (blue triangles) and $\alpha_{max}/2\pi = 3$ MHz (orange circles). The positive eigenvalues are shown on a log-scale on the left and the small eigenvalues are shown on a linear scale on the right. 150
5.7 Infidelity objective ($\mathcal{J}_1$) and guard level objective ($\mathcal{J}_2$) as function of ε in $H_s^u(\epsilon)$. Here 'NF' and 'RN' correspond to the "Noise-Free" and "Risk-Neutral" cases. 152
5.8 Control functions (without carrier waves) for the cases: "noise-free" (top), and "risk-neutral" (bottom). Here, $p_{k,n}(t)$ and $q_{k,n}(t)$ are defined in (5.27). 153
5.9 Magnitude of the Fourier spectrum of the laboratory frame control function for the $|0\rangle \leftrightarrow |5\rangle$ SWAP gate. 159

6.1 Results of the Rabi experiment on qubit 5 of the Casablanca hardware. 169
6.2 The pulse schedule for an X gate on Casablanca for qubit 5 (left), and qubit 6 (right). 173
6.3 The pulse schedule for a CNOT gate on Casablanca where qubit 5 is the control qubit and qubit 6 is the target. 175
6.4 The pulse schedule for the first interleaved RB circuit. 179
6.5 Simulated results. The observed ground state population as function of the length of the Clifford circuit. Here, the non-interleaved circuits are shown in blue and interleaved ones in red. The estimated error per Clifford (EPC) is 2.36%. 180

6.6 Classification results after 1000 shots in two of the randomized circuits with one interleaved SWAP gate. Ideally the $|00\rangle$ state should have 100% of the population. 181

6.7 Randomized benchmarking results on the Casablanca hardware using the InterleavedRBFitter. 181

1 Values of $\alpha = \cos(\omega\Delta t)(2 + \omega^2\Delta t^2)$ for values of $\omega\Delta t$ in the interval $[0, 2]$. The red lines indicate the desired bound on α, and the black line indicates the maximum allowable value of $\omega\Delta t$ at $r \approx 1.93$. 209

2 A plot of the discrete filter function using five time-steps $0 \leq r \leq 5/4$. On the left we plot the full range of values of r, and on the right we zoom in close to resonance, i.e. $r = 1$. 210

3 A bound on the gap from resonance that creates problematic modes. The blue curve is the true gap, $1 - r^*$, and the dotted red curve is a proposed bound. 211

Chapter 1

Introduction

Many problems of practical interest in diverse areas such as acoustics, seismics, and quantum mechanics are governed by equations in which the solution is defined by a wave or a superposition of waves. Waves exist at an incredibly wide range of scales, and their accurate numerical treatment is of great practical importance. In acoustic scattering problems, for instance, it is desirable to obtain time-harmonic solutions to the wave equation. These solutions satisfy the **Helmholtz equation**, the efficient and scalable solution of which is an active area of research (see the review articles [47, 50, 44]). On a much smaller scale, the wave-particle duality is a foundational principle of quantum mechanics which implies that all information about a particle is contained in its wave function. This wave function, which can be interpreted as a probability distribution, evolves according to the **Schrödinger equation** and is key to understanding how we may eventually exploit the nascent power of quantum computers.

In either case, the numerical treatment of wave propagation problems requires high-order accurate and efficient numerical methods. These methods must be able to scale well in many dimensions, and potentially simulate over long distances and/or times. This thesis will develop numerical methods for two different wave problems: (1) an iterative wave equation solution method for Helmholtz problems, and (2) optimizing control functions for realizing logical gates in closed quantum systems, where the evolution of the state vector is governed by the time dependent Schrödinger equation.

The chapters of this thesis are thus separated into two broad sections. The first section,

Chapters 2-4, concern time-domain methods for Helmholtz problems. The second section, **Chapters 5-6**, concern both theory and practice of optimal control methods for quantum systems. In each section, there is considerable overlap in notation but this notation is generally restated and redefined within each chapter so that they may be read independently. Each of **Chapters 2-5** has either been published [14], submitted for publication [94], or is to be submitted for publication [51, 13].

1.1 Outline of chapters

The first section of the thesis, **Chapters 2-4**, focus on time-domain methods for solving the Helmholtz equation. In **Chapter 2** we introduce the WaveHoltz iteration, which is a fixed-point iteration that filters the solution of the wave equation with time-periodic forcing and boundary data. We show that the WaveHoltz iteration can be recast as a positive definite linear system of equations which can be solved using Krylov subspace techniques. We additionally present a continuous and discrete analysis for energy-conserving problems.

In **Chapter 3**, we extend the analysis of the energy-conserving WaveHoltz iteration to problems with damping and/or impedance boundary conditions. Furthermore, we investigate higher order modified equation timestepping schemes and show that the WaveHoltz solution converges to the discrete Helmholtz solution to the order matching the order of the timestepping scheme. We then present a method to *completely* remove time discretization error from the WaveHoltz solution.

In **Chapter 4**, we apply the WaveHoltz iteration to the "elastic" Helmholtz equation (also known as the Navier equation) for energy-conserving problems with Dirichlet and/or free surface boundary conditions. We present a discrete analysis for an implicit timestepping scheme in which time discretization errors are removed.

In the second section of the thesis, **Chapters 5-6**, we consider optimal control methods for quantum systems. In **Chapter 5**, we describe an optimal control problem for closed quantum systems governed by Schrödinger's equation. We motivate and describe the novel use of B-splines with carrier waves to interpolate control functions, which allow the number of parameters to be

independent of the number of timesteps used in the simulation. The system is discretized with the Störmer-Verlet scheme, which is a symplectic partitioned Runge-Kutta scheme. Using a "discretize-then-optimize" approach, we derive a discrete timestepping scheme used to compute *exact* discrete gradients at the cost of solving two Schrödinger systems. The methods described in this chapter have been implemented in the Julia programming language, [1], and are made available as the open-source package Juqbox (available through GitHub at `https://github.com/LLNL/Juqbox.jl`).

In **Chapter 6**, we apply the methods of **Chapter 5** to the IBM SWAP Gate Challenge. We describe the approach taken for a submission to the SWAP Gate Challenge, and present the results of building a custom gate for a real-word noisy quantum system.

Finally in **Chapter 7**, we summarize the results of the thesis. Additionally, future research directions are discussed.

Chapter 2

WaveHoltz: Iterative Solution of the Helmholtz Equation via the Wave Equation

The defining feature of waves are their ability to propagate over large distances without changing their shape. It is this property that allows them to carry information which underpins all communication, be it through speech or electronic transmission of data. Waves can also be used to probe the interior of the earth, the human body or engineering structures like buildings or bridges. This probing can be turned into images of the interior by the means of solving inverse problems. Harnessing the nature of waves requires high-order accurate and efficient numerical methods that are able to simulate wave propagation in three dimensions and over long distances. For cutting edge problems in scientific and engineering research such simulations must be carried out on parallel high-performance computing platforms and thus the numerical methods must scale while being easy to implement and generally applicable.

In this chapter we focus on approximating solutions to the scalar wave equation in the frequency domain, i.e. the Helmholtz equation

$$\nabla \cdot (c^2(x)\nabla u) + \omega^2 u = f(x). \tag{2.1}$$

However, to to obtain such solutions we will use time domain discretizations of the wave equation. The motivation for developing high order accurate and scalable Helmholtz solvers comes from both mathematics and applications. On the mathematics side the recent results by Engquist and Zhao [42] give sharp lower bounds on the number of terms in a separated representation approximation of the Green's function of the Helmholtz equation as a function of the frequency (wavenumber).

These bounds limit the applicability of the state of the art sweeping preconditioners in the high frequency regime and, for example, for interior and wave guide problems. Motivation also comes from applications in seismology, optics and acoustics. For example in full waveform inversion the problems are very large and the robustness of the inversion process can be enhanced by combining frequency and time domain inversion in a multi-scale fashion to avoid getting trapped in local minima.

Designing efficient iterative solvers for the Helmholtz equation (2.1) is notoriously difficult and has been the subject of much research (for detailed reviews see Ernst and Gander, [47], Gander and Zhang [50], and Erlangga, [44]). The main two difficulties in solving the Helmholtz equation are the resolution requirements and the highly indefinite character of the discretized system of equations.

Assuming that (2.1) has been scaled so that the mean of $c(x)$ is about 1 then the typical wavelength is $\lambda = 2\pi/\omega$ and the typical wavenumber is $\omega/2\pi$. In order to numerically propagate solutions to the time dependent wave equation corresponding to (2.1) with small errors it is crucial to control the dispersion by using high order methods. The basic estimate by Kreiss and Oliger [75] shows that in order to propagate a wave over J wavelengths with a pth order finite difference method and with an error no greater than ϵ one must choose the number of points per wavelength $\mathrm{PPW}(J,p)$ as

$$\mathrm{PPW}(J,p) \geq C(p,\epsilon)J^{\frac{1}{p}}.$$

Here $C(p,\epsilon)$ depends on the tolerance ϵ but decreases with increasing order of accuracy p. Consequently, for a problem in d-dimensions and with fixed physical size the number of wavelengths in the domain will scale as ω^d and to maintain a fixed tolerance the total number of degrees of freedom needed, $N_p(\omega) = \mathcal{O}(\omega^{d(1+\frac{1}{p})})$, is very large for high frequencies.

The dependence on p and ω in $N_p(\omega)$ immediately reveals two fundamental criteria for designing high frequency Helmholtz solvers:

1. The solvers must be **parallel, memory lean** and they must **scale well**. In 3D the number

of degrees of freedom representing the solution cannot be stored on a single computer, and even on a parallel computer it is important to preserve the sparsity of the discrete version of (2.1).

2. The underlying discretizations must be **high order accurate**. At high frequencies and in 3D the extra penalty due to pollution / dispersion errors becomes prohibitive.

Further, the linear system matrix, A, resulting from direct discretization of (2.1) is indefinite so that the robust and easy to implement preconditioned conjugate gradient (PCG) method cannot be used. Instead the method of necessity becomes the preconditioned generalized minimal residual method (GMRES). To efficiently precondition GMRES one must exploit the intrinsic properties of the wave equation. The oscillatory nature of the Helmholtz Green's function and its discrete counterpart A^{-1} can only be well approximated if the (unconditioned) Krylov subspace is allowed to grow quite large (with "large" scaling adversely with the frequency ω , [47]). The slow growth of the "spanning power" of the Krylov vectors is due to the underlying local connectivity of the discretization, preventing information to propagate rapidly. Efficient preconditioners must thus accelerate the propagation of information or reduce the cost of each iteration. Without preconditioners the iteration typically stagnates.

Perhaps the first contribution that aimed to improve the propagation of information was the Analytic Incomplete LU preconditioner (AILU) by Gander and Nataf [49]. The AILU preconditioner finds an LDL^T factorization from an approximation of the same pseudodifferential operators that are used to construct non-reflecting boundary conditions [39, 6, 66] and sweeps forward then backward along one of the coordinate directions in a structured grid.

The pioneering works on sweeping preconditioners by Engquist and Ying [40, 41] were major breakthroughs in the solution of the Helmholtz equation. Similar to the AILU, the preconditioners in [40, 41] use a LDL^T decomposition but exploit the low rank properties of off-diagonal blocks together with perfectly matched layers to obtain solvers that converge in a small number of GMRES iterations. The papers [40, 41] were the two first instances of iterative Helmholtz solvers that

converge in a small number of iterations that is almost independent of frequency.

Once it had been established that low rank approximations, combined with clever use of sweeping and perfectly matched layers (PML), could be used to find Helmholtz solvers with linear scaling then many extensions and specializations were constructed. For example, in [100] Stolk introduced a domain decomposition method with transmission conditions based on the perfectly matched layer (PML) that is able to achieve near linear scaling. Chen and Xiang, [34], and Vion and Geuzaine, [105], also considered sweeping domain decomposition method combined with PML and showed that their methods could be used as efficient preconditioners for the Helmholtz equation. The method of polarized traces by Zepeda-Núñez, Demanet and co-authors, [113, 112, 111], is a two step sweeping preconditioner that compresses the traces of the Greens function in an offline computation and utilizes incomplete Green's formulas to propagate the interface data. See also the recent review by Gander and Zhang [50] for connections between sweeping methods.

Alongside iterative methods there are some attractive direct and multigrid methods. Examples from the class of direct methods are the Hierarchically Semi-Separable (HSS) parallel multifrontal sparse solver by deHoop and co-authors, [107], the spectral collocation solver by Gillman, Barnett and Martinsson, [57], and the p-FEM approach of Bériot, Prinn and Gabard, [23], which utilizes an *a priori* error indicator to choose the polynomial order of each element . Notable examples of multigrid methods are the Wave-ray method by Brandt and Livshits [28, 81] and the shifted Laplacian preconditioner with multigrid by Erlangga et al. [43].

As mentioned previously, the invention of sweeping preconditioners was a breakthrough and it is likely that they will have lasting and continuing impacts for the solution of the Helmholtz equation in various settings. There are, however, some limitations. First, in the recent paper [42], Engquist and Zhao provide precise lower bounds on how the number of terms that are needed to approximate the Helmholtz Green's function depends on the frequency. In particular, for the high frequency regime they show that for interior problems and waveguides the rank of the off-diagonal elements grows fast, rendering sweeping preconditioners less efficient. They also show that the situation is, in general, worse in 3D than in 2D. This lack of compressibility may, in cases of

practical importance, increase the cost of both the factorization and compression as well as the application of the compressed preconditioner. We note that this loss of compressibility at high frequency will also prevent direct methods such as [107, 57, 23] from reaching their most efficient regimes. An additional drawback of direct methods is their memory consumption for 3D problems.

Another potential drawback with the sweeping methods is the long setup times before the solve. Of course all of the algorithms above do not suffer from this deficiency but many of them do. This may not be problematic when considering a background velocity that does not change but this is not the case, for example, when inverting for material parameters. In this case the velocity model will change constantly, necessitating a costly factorization in each update.

Finally, the two criterions 1.) and 2.) above are not so easy to meet for sweeping preconditioners. The sweep itself is intrinsically sequential and although there have been at least partially successful attempts to parallelize the sweeping methods it is hard to say that they are easy to parallelize in a scalable way. In a similar vein most of the methods use (and some rely on) low order discretizations. Although it is possible to use higher order accurate discretizations together with sweeping preconditioners, their scarcity in the literature is noticeable.

Another approach that is somewhat popular in the engineering literature is to simply run the wave equation for a long time to get a Helmholtz solution, see e.g. [70]. The theoretical underpinning of this approach is the *limiting amplitude principle* which says that every solution to the wave equation with an oscillatory forcing, in the exterior of a domain with reflecting boundary conditions tends to the Helmholtz solution. However, since the limiting amplitude principle only holds for exterior problems this approach does not work for interior problems and becomes very slow for problems with trapping waves. See e.g. the articles by Ladyzhenskaya [76], Morawetz [88] and Vainberg [104].

An alternative approach, the so called Controllability Method (CM), was originally proposed by Bristeau et al. [29]. In the CM the solution to the Helmholtz equation is found by solving a convex constrained least-squares minimization problem where the deviation from time-periodicity is minimized in the classic wave equation energy. The basic ingredients in an iteration step in CM

are: a.) the solution of a forward wave and a backward wave equation over one time-period, and b.) the solution of a symmetric coercive elliptic (and wave number independent) problem.

In [29] and the later spectral element implementations of CM by Heikkola et al. [69, 68] only sound-soft scatterers were considered. For more general boundary conditions the minimizer of the cost functional of [29] is not unique but alternative cost functionals that does guarantee uniqueness (and thus convergence to the Helmholtz solution) were recently proposed by Grote and Tang in [63]. We also note that if the wave equation is formulated as a first order system it is possible to avoid solving the elliptic problem [58, 61].

In what follows we will present an alternative to the controllability method. Our method, which we call the WaveHoltz Iteration method (WHI), only requires a single forward wave equation solve and no elliptic solves but produces a positive definite (and sometimes symmetric) iteration that can be accelerated by, e.g. the conjugate gradient method or other Krylov subspace methods. As the WaveHoltz iteration is built from a time domain wave equation solver we claim and hope to demonstrate that it meets both criterion 1. and 2. above.

The rest of the chapter is organized as follows. In Section 2 we present and analyze our method and its extensions, in Section 3 we briefly outline the numerical methods we use to solve the wave equation, in Section 4 we present numerical experiments, and in Section 5 we summarize and conclude.

Before proceeding we would like to acknowledge that although our method is distinct from the controllability method, it was the work by Grote and Tang, [63], that introduced us to CM and inspired us to derive the method discussed below.

2.1 WaveHoltz: A New Method for Designing Scalable Parallel Helmholtz Solvers

We consider the Helmholtz equation in a bounded open smooth domain Ω,

$$\nabla \cdot (c^2(x)\nabla u) + \omega^2 u = f(x), \qquad x \in \Omega, \tag{2.2}$$

with boundary condtions of the type

$$i\alpha\omega u + \beta(c^2(x)\vec{n}\cdot\nabla u) = 0, \quad \alpha^2+\beta^2 = 1, \qquad x\in\partial\Omega. \tag{2.3}$$

We assume $f \in L^2(\Omega)$ and that $c \in L^\infty(\Omega)$ with the bounds $0 < c_{\min} \le c(x) \le c_{\max} < \infty$ a.e. in Ω. Away from resonances, this ensures that there is a unique weak solution $u \in H^1(\Omega)$ to (2.2). Due to the boundary conditions u is in general complex valued.

We first note that the function $w(t,x) := u(x)\exp(i\omega t)$ is a $T = 2\pi/\omega$-periodic (in time) solution to the forced scalar wave equation

$$\begin{aligned} &w_{tt} = \nabla\cdot(c^2(x)\nabla w) - f(x)e^{i\omega t}, \quad x\in\Omega, \quad 0\le t\le T,\\ &w(0,x) = v_0(x), \quad w_t(0,x) = v_1(x),\\ &\alpha w_t + \beta(c^2(x)\vec{n}\cdot\nabla w) = 0, \quad x\in\partial\Omega, \end{aligned} \tag{2.4}$$

where $v_0 = u$ and $v_1 = i\omega u$. Based on this observation, our approach is to find this w instead of u. We could thus look for initial data v_0 and v_1 such that w is a T-periodic solution to (2.4). However, there may be several such w, see [63], and we therefore impose the alternative constraint that a certain time-average of w should equal the initial data. More precisely, we introduce the following operator acting on the initial data $v_0 \in H^1(\Omega)$, $v_1 \in L^2(\Omega)$,

$$\Pi\begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right)\begin{bmatrix} w(t,x) \\ w_t(t,x) \end{bmatrix} dt, \quad T = \frac{2\pi}{\omega},$$

where $w(t,x)$ and its time derivative $w_t(t,x)$ satisfies the wave equation (2.4) with initial data v_0 and v_1. The result of $\Pi[v_0,\ v_1]^T$ can thus be seen as a filtering in time of $w(\cdot,x)$ around the ω-frequency. We will further motivate the choice of time averaging in the analysis below. By construction, the solution u of Helmholtz now satisfies the equation

$$\begin{bmatrix} u \\ i\omega u \end{bmatrix} = \Pi\begin{bmatrix} u \\ i\omega u \end{bmatrix}. \tag{2.5}$$

The WaveHoltz method then amounts to solving this equation with the fixed point iteration

$$\boxed{\begin{bmatrix} v \\ v' \end{bmatrix}^{(n+1)} = \Pi \begin{bmatrix} v \\ v' \end{bmatrix}^{(n)}, \qquad \begin{bmatrix} v \\ v' \end{bmatrix}^{(0)} \equiv 0.} \tag{2.6}$$

Provided this iteration converges and the solution to (2.5) is unique, we obtain the Helmholtz solution as $u = \lim_{n\to\infty} v^n$.

Remark 2.1.1. *Note that each iteration is inexpensive and that T is reduced by the reciprocal of ω as ω grows. If we assume that the number of degrees of freedom in each dimension scales with ω and that we evolve the wave equation with an explicit method this means that the number of timesteps per iteration is independent of ω. Also note that the iteration is trivial to implement (in parallel or serial) if there is already a time domain wave equation solver in place. The integral in the filtering is carried out independently for each degree of freedom and simply amounts to adding up a weighted sum (e.g. a trapezoidal sum) of the solution one timestep at a time. Finally, note that WHI allows all the advanced techniques that have been developed for wave equations (e.g. local timestepping, non-conforming discontinuous Galerkin finite elements h- and p-adaptivity etc.) can be transferred to the Helmholtz equation and other time harmonic problems.*

2.1.1 Iteration for the Energy Conserving Case

Here we consider boundary conditions of either Dirichlet ($\beta = 0$) or Neumann ($\alpha = 0$) type. This is typically the most difficult case for iterative Helmholtz solvers when Ω is bounded. The wave energy is preserved in time and certain ω-frequencies in Helmholtz are resonant, meaning they equal an eigenvalue of the operator $-\nabla \cdot (c^2(x)\nabla)$. Moreover, the limiting amplitude principle does not hold, and one can thus not obtain the Helmholtz solution by solving the wave equation over a long time interval.

We start by introducing a simplified iteration for this case. With the given boundary conditions the solution to Helmholtz will be real valued, since f is a real valued function. Without loss of generality, we may then take $w_t(0, x) = 0$ and $w(t, x) = u(x)\cos(\omega t)$, since for a T-periodic real

valued solution there is a time when $w_t(0,x) = 0$. We choose that time as the initial time so that (2.4) becomes

$$\begin{aligned} &w_{tt} = \nabla \cdot (c(x)^2 \nabla w) - f(x)\cos(\omega t), \quad x \in \Omega, \quad 0 \le t \le T, \\ &w(0,x) = v(x), \quad w_t(0,x) \equiv 0, \\ &\alpha w_t + \beta(c^2(x)\vec{n} \cdot \nabla w) = 0, \quad x \in \partial\Omega. \end{aligned} \tag{2.7}$$

The simplified iteration is then defined as

$$\boxed{v^{n+1} = \Pi v^n, \qquad v^0 \equiv 0,} \tag{2.8}$$

where

$$\Pi v = \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right) w(t,x)dt, \qquad T = \frac{2\pi}{\omega}, \tag{2.9}$$

with $w(t,x)$ solving the wave equation (2.7) with initial data $v = v^n \in H^1(\Omega)$. We now analyze this iteration.

By the choice of boundary conditions the operator $-\nabla \cdot (c^2(x)\nabla)$ has a point spectrum with non-negative eigenvalues with corresponding eigenfunctions that form an orthonormal basis of $L^2(\Omega)$. Denote those eigenmodes $(\lambda_j^2, \phi_j(x))$, with $\|\phi_j\|_{L^2(\Omega)} = 1$. We assume that the angular frequency ω is not a resonance, i.e. $\omega^2 \neq \lambda_j^2$ for all j. The Helmholtz equation (2.2) is then wellposed.

We recall that for any $q \in L^2(\Omega)$ we can expand

$$q(x) = \sum_{j=0}^{\infty} \hat{q}_j \phi_j(x),$$

for some coefficients $\hat{q}_j$ and

$$||q||^2_{L^2(\Omega)} = \sum_{j=0}^{\infty} |\hat{q}_j|^2, \qquad c^2_{\min}||\nabla q||^2_{L^2(\Omega)} \le \sum_{j=0}^{\infty} \lambda_j^2 |\hat{q}_j|^2 \le c^2_{\max}||\nabla q||^2_{L^2(\Omega)}.$$

We start by expanding the Helmholtz solution u, the initial data v to the wave equation (2.7), and the forcing f in this way,

$$u(x) = \sum_{j=0}^{\infty} \hat{u}_j \phi_j(x), \qquad v(x) = \sum_{j=0}^{\infty} \hat{v}_j \phi_j(x), \qquad f(x) = \sum_{j=0}^{\infty} \hat{f}_j \phi_j(x).$$

Then,

$$-\lambda_j^2 \hat{u}_j + \omega^2 \hat{u}_j = \hat{f}_j \quad \Rightarrow \quad \hat{u}_j = \frac{\hat{f}_j}{\omega^2 - \lambda_j^2}.$$

For the wave equation solution $w(t,x)$ with initial data $w = v$ and $w_t = 0$ we have

$$w(t,x) = \sum_{j=0}^{\infty} \hat{w}_j(t)\phi_j(x), \qquad \hat{w}_j(t) = \hat{u}_j\Big(\cos(\omega t) - \cos(\lambda_j t)\Big) + \hat{v}_j \cos(\lambda_j t). \tag{2.10}$$

The filtering step (2.9) then gives

$$\Pi v = \sum_{j=0}^{\infty} \bar{v}_j \phi_j(x), \qquad \bar{v}_j = \hat{u}_j(1 - \beta(\lambda_j)) + \hat{v}_j \beta(\lambda_j),$$

where

$$\beta(\lambda) := \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right)\cos(\lambda t)dt.$$

We introduce the linear operator $\mathcal{S} : L^2(\Omega) \to L^2(\Omega)$,

$$\mathcal{S}\sum_{j=0}^{\infty} \hat{u}_j \phi_j(x) := \sum_{j=0}^{\infty} \beta(\lambda_j)\hat{u}_j\phi_j(x), \tag{2.11}$$

which gives the filtered solution of the wave equation with $f = 0$, when applied to the initial data v. We can then write the iteration as

$$v^{n+1} = \Pi v^n = \mathcal{S}(v^n - u) + u. \tag{2.12}$$

The operator $\mathcal{S}$ is self-adjoint and has the same eigenfunctions $\phi_j(x)$ as $-\nabla\cdot(c^2(x)\nabla)$ but with the (real) eigenvalues $\beta(\lambda_j)$. The convergence properties of the iteration depend on these eigenvalues and it is therefore of interest to study the range of the filter transfer function β. Figure 2.1 shows a plot of β which indicates that the eigenvalues of $\mathcal{S}$ are inside the unit interval, with a few of them being close to 1 (when $\lambda_j \approx \omega$), and most of them being close to zero (when $\lambda_j \gg \omega$). In the appendix we show the following lemma about β.

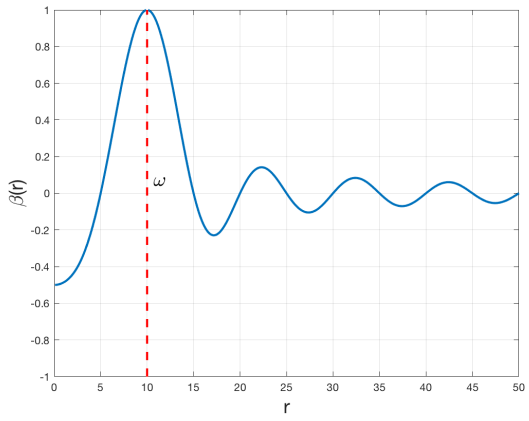


Figure 2.1: The filter transfer function β for $\omega = 10$.

Lemma 2.1.1. *The filter transfer function β satisfies $\beta(\omega) = 1$ and*

$$0 \le \beta(\lambda) \le 1 - \frac{1}{2}\left(\frac{\lambda-\omega}{\omega}\right)^2, \qquad \text{when } \left|\frac{\lambda-\omega}{\omega}\right| \le \frac{1}{2},$$
$$|\beta(\lambda)| \le \frac{1}{2}, \qquad \text{when } \left|\frac{\lambda-\omega}{\omega}\right| \ge \frac{1}{2},$$
$$|\beta(\lambda)| \le b_0 \frac{\omega}{\lambda-\omega}, \qquad \text{when } \lambda > \omega.$$

where $b_0 = \frac{3}{4\pi}$. Moreover, close to ω we have the local expansion

$$\beta(\omega + r) = 1 - b_1\left(\frac{r}{\omega}\right)^2 + R(r/\omega)\left(\frac{r}{\omega}\right)^3, \qquad b_1 = \frac{2\pi^2}{3} - \frac{1}{4} \approx 6.33, \quad ||R||_\infty \le \frac{5\,\pi^3}{6}. \tag{2.13}$$

Remark 2.1.2. *It is easy to see that $\beta(\omega) = 1$ for any constant besides $1/4$. The particular choice $1/4$ is made to ensure that $\beta'(\omega) = 0$, which is necessary to keep $\beta \le 1$ in a neighborhood of ω. We explore other possibilities in Section 2.1.3.*

From this lemma we can derive some results for the operator $\mathcal{S}$. To do this we first quantify the non-resonance condition. We let

$$\delta_j = \frac{\lambda_j - \omega}{\omega},$$

be the relative size of the gap between λ_j and the Helmholtz frequency, and then denote the smallest gap (in magnitude) by δ,

$$\delta = \delta_{j^*}, \qquad j^* = \text{argmin}_j |\delta_j|.$$

611 Then we have

612 **Lemma 2.1.2.** *Suppose $\delta > 0$. The spectral radius ρ of $\mathcal{S}$ is strictly less than one, and for small*
613 *δ,*

$$\rho = 1 - b_1\delta^2 + O(\delta^3), \tag{2.14}$$

614 *with b_1 as in Lemma 2.1.1. Moreover, $\mathcal{S}$ is a bounded linear map from $L^2(\Omega)$ to $H^1(\Omega)$.*

Proof. From Lemma 2.1.1 we get

$$\rho = \sup_j |\beta(\lambda_j)| \leq \sup_j \max\left(1 - \frac{1}{2}\delta_j^2,\ \frac{1}{2}\right) \leq \max\left(1 - \frac{1}{2}\delta^2,\ \frac{1}{2}\right) < 1.$$

For the more precise estimate when δ is small we will use (2.13). Since $1 > \rho \geq \beta(\omega + \omega\delta) \to 1$ as $\delta \to 0$, we can assume that $\rho > 1 - \eta^2/2$, with $\eta := b_1/2||R||_\infty$, for small enough δ. Then, since $|\beta(\omega + \omega\delta_j)| \leq 1 - \eta^2/2$ for $|\delta_j| > \eta$ by Lemma 2.1.1, we have

$$\rho = \sup_{|\delta_j| \leq \eta} \beta(\omega + \omega\delta_j) = \beta(\omega + \omega\delta_{k^*}),$$

for some k^* with $|\delta_{k^*}| \leq \eta$. If $\delta_{k^*} = \delta_{j^*}$ (where $\delta = |\delta_{j^*}|$) then (2.13) gives (2.14). If not, we have $\eta \geq |\delta_{k^*}| \geq \delta$ and by Lemma 2.1.1

$$0 \leq \beta(\omega+\omega\delta_{k^*}) - \beta(\omega+\omega\delta_{j^*}) = -b_1(\delta_{k^*}^2 - \delta^2) + R(\delta_{k^*})\delta_{k^*}^3 - R(\delta_{j^*})\delta_{j^*}^3 \leq -b_1(\delta_{k^*}^2 - \delta^2) + \frac{b_1}{2}(\delta_{k^*}^2 + \delta^2),$$

which implies that $\delta_{k^*}^2 \leq 3\delta^2$ and that

$$0 \leq b_1(\delta_{k^*}^2 - \delta^2) \leq R(\delta_{k^*})\delta_{k^*}^3 - R(\delta_{j^*})\delta_{j^*}^3 \leq ||R||_\infty(1 + 3\sqrt{3})\delta^3.$$

Therefore,

$$\rho = 1 - b_1\delta_{k^*}^2 + O(\delta_{k^*}^3) = 1 - b_1\delta^2 + b_1(\delta^2 - \delta_{k^*}^2) + O(\delta_{k^*}^3) = 1 - b_1\delta^2 + O(\delta_{k^*}^3 + \delta^3) = 1 - b_1\delta^2 + O(\delta^3).$$

615 This shows (2.14). For the second statement, we note first that by Lemma

Lemma 1. *filterlemma,*

$$|\lambda_j\beta(\lambda_j)| \leq \omega \begin{cases} 1, & \lambda_j \leq \omega, \\ \frac{b_0\lambda_j}{\lambda_j - \omega}, & \lambda_j > \omega, \end{cases} = \omega \begin{cases} 1, & \lambda_j \leq \omega, \\ b_0(1 + 1/\delta_j), & \lambda_j > \omega, \end{cases} \leq \omega \min(1, b_0(1 + 1/|\delta|)) =: D.$$

Suppose now that $g \in L^2(\Omega)$ *and*

$$g(x) = \sum_{j=0}^{\infty} \hat{g}_j \phi_j(x).$$

Then

$$||\mathcal{S}g||^2_{H^1(\Omega)} \leq \sum_{j=0}^{\infty} |\beta(\lambda_j)|^2 |\hat{g}_j|^2 + \sum_{j=0}^{\infty} \frac{\lambda_j^2}{c_{\min}^2} |\beta(\lambda_j)|^2 |\hat{g}_j|^2 \leq \left(1 + \frac{D^2}{c_{\min}^2}\right) \sum_{j=0}^{\infty} |\hat{g}_j|^2 = \left(1 + \frac{D^2}{c_{\min}^2}\right) ||g||^2_{L^2(\Omega)}.$$

This proves the lemma. □

Letting $e^n := u - v^n$ we can rearrange (2.12) and obtain

$$e^{n+1} = \mathcal{S}e^n \quad \Rightarrow \quad ||e^{n+1}||_{L^2(\Omega)} \leq \rho ||e^n||_{L^2(\Omega)} \quad \Rightarrow \quad ||e^n||_{L^2(\Omega)} \leq \rho^n ||e^0||_{L^2(\Omega)} \to 0,$$

which shows that v^n converges to u in L^2. By Lemma 2.1.2 all iterates $v^n \in H^1(\Omega)$ since $v^0 = 0$. We can therefore also get convergence in H^1. Let

$$e^n(x) = \sum_{j=0}^{\infty} \hat{e}_j^n \phi_j(x),$$

and consider similarly

$$\sum_{j=0}^{\infty} |\hat{e}_j^{n+1}|^2 \lambda_j^2 = \sum_{j=0}^{\infty} \beta(\lambda_j)^2 |\hat{e}_j^n|^2 \lambda_j^2 \leq \rho^2 \sum_{j=0}^{\infty} |\hat{e}_j^n|^2 \lambda_j^2 \quad \Rightarrow$$

$$||\nabla e^n||^2_{L^2(\Omega)} \leq \frac{1}{c_{\min}^2} \sum_{j=0}^{\infty} |\hat{e}_j^n|^2 \lambda_j^2 \leq \frac{\rho^{2n}}{c_{\min}^2} \sum_{j=0}^{\infty} |\hat{e}_j^0|^2 \lambda_j^2 \leq \rho^{2n} \frac{c_{\max}^2}{c_{\min}^2} ||\nabla e^0||^2_{L^2(\Omega)} \to 0.$$

We conclude that the iteration converges in H^1 with convergence rate ρ. By Lemma 2.1.1 we have $\rho \sim 1 - 6.33\delta^2$ and, not surprisingly, the smallest gap, δ, determines the convergence factor. We have thus showed

Theorem 2.1.3. *The iteration in (2.8) and (2.9) converges in* $H^1(\Omega)$ *for the Dirichlet and Neumann problems away from resonances to the solution of the Helmholtz equation (2.2). The convergence rate is* $1 - O(\delta^2)$*, where* δ *is the minimum gap between* ω *and the eigenvalues of* $-\nabla \cdot (c^2(x)\nabla)$*.*

As discussed in the introduction, the dependence of the convergence rate on ω is often of interest. For the energy conserving case, however, this question is ambiguous as the problem is not well-defined for all ω. As soon as $\omega = \lambda_j$ there are either no or an infinite number of solutions.

In higher dimensions, the eigenvalues λ_j get denser as j increases, meaning that in general the problem will be closer and closer to resonance as ω grows. Therefore, solving the interior undamped Helmholtz equation for high frequencies, with pure Dirichlet or Neumann boundary conditions, may not be of great practical interest.

Nevertheless, we can make the following analysis. By the work of Weyl [109] we know that the eigenvalues grow asymptotically as $\lambda_j \sim j^{1/d}$ in d dimensions. The average minimum gap δ when $\omega \approx \lambda_j$ is then

$$\begin{aligned}\delta \approx \frac{1}{\lambda_{j+1}-\lambda_j}\int_{\lambda_j}^{\lambda_{j+1}} \frac{\min(\lambda-\lambda_j, \lambda_{j+1}-\lambda)}{\omega} d\lambda = \frac{\lambda_{j+1}-\lambda_j}{4\omega} \sim \frac{(j+1)^{1/d}-j^{1/d}}{\omega} \approx \frac{j^{1/d-1}}{d\omega} \\ \sim \frac{\omega^{1-d}}{\omega} \\ \sim \omega^{-d}.\end{aligned}$$

When the convergence rate is $1 - O(\delta^2)$, the number iterations to achieve a fixed accuracy grows as $O(1/\delta^2)$. This shows that the number of iterations would grow at the unacceptable rate ω^{2d} for the iteration.

Fortunately, one can accelerate the convergence by using the conjugate gradient method in the energy conserving case and with any other Krylov method in the general case. The linear system that we actually want to solve is

$$(I - \mathcal{S})v =: \mathcal{A}v = b := \Pi 0.$$

Moreover, with $b = \Pi 0$ pre-computed we can easily evaluate the action of $\mathcal{A}$ at the cost of a single wave solve. Precisely, since $\mathcal{A}v = v - \Pi v + b$ we simply carry out the evaluation of $\mathcal{A}v$ by evolving the wave equation for one period in time with v as the initial data and then subtract the filtered solution from the sum of the initial data and the right hand side b.

The operator $\mathcal{A}$ is self adjoint and positive, since $-1/2 < \beta(\lambda_j) < 1$, which implies that the eigenvalues of $\mathcal{A}$ lie in the interval $(0, 3/2)$. The condition number of $\mathcal{A}$ is of the same order as $1 - \rho$, where ρ is the spectral radius of $\mathcal{S}$, i.e. by the simple analysis above, $\text{cond}(\mathcal{A}) \sim \omega^{2d}$. If this system is solved using the (**unconditioned**) conjugate gradient method the convergence rate

is $1 - 1/\sqrt{\text{cond}(\mathcal{A})} \sim 1 - 1/\omega^d$, [25]. Thus, then the method just requires $\sim \omega^d$ iterations for fixed accuracy.

Remark 2.1.3. *The operator $\mathcal{A}$ is self-adjoint and coercive when $\delta > 0$ since*

$$\langle \mathcal{A}u, u\rangle = \langle (I - \mathcal{S})u, u\rangle = \sum_{j=0}^{\infty} (1 - \beta(\lambda_j))|\hat{u}_j|^2 \geq (1 - \rho)||u||^2.$$

This should be contrasted with the original indefinite Helmholtz problem, which is not coercive. In fact, the eigenvalues satisfy the simple relation $\lambda_{\text{WHI}} = 1 - \beta(\lambda_{\text{Helmholtz}} + \omega)$ ¿ 0. The two formulations are however mathematically equivalent for the interior Dirichlet and Neumann problems away from resonances, as the analysis above shows.

The coercivity also implies that the solution to (2.5) for the simplified iteration is unique since $w = \Pi w$ is equivalent to $\mathcal{A}(w - u) = 0$.

Remark 2.1.4. *A discretization would have approximately a fixed number of grid points per wavelength, leading to a (sparse) matrix of size $N \times N$ with $N \sim \omega^d$. Hence, the number of iterations for WHI would be $O(N^2)$ and the total cost $O(N^3)$ since each iteration costs $O(N)$. This should be compared with a direct solution method which is better than $O(N^3)$ when the matrix is sparse.*

Remark 2.1.5. *In the Krylov accelerated case this analysis suggests that the number of iterations would now be $O(N)$ and the total cost $O(N^2)$. However, in the experiments below we observe slightly better complexity for interior problems and significantly better complexity for open problems. In fact, for the open problems we find that, in both two and three dimensions, the number of iterations scale as $\sim \omega$ which is the required number of iterations for the information to travel through the domain.*

2.1.2 Analysis of the Discrete Iteration

To better understand the effects of discretizations we consider the following discrete version of the algorithm for the energy conserving case described above in Section 2.1.1. We introduce the temporal grid points $t_n = n\Delta t$ and a spatial grid with N points together with the vector $w^n \in \mathbb{R}^N$ containing the grid function values of the approximation at $t = t_n$. We also let $f \in \mathbb{R}^N$ hold the

corresponding values of the right hand side. The discretization of the continuous spatial operator $-\nabla \cdot (c^2(x)\nabla)$, including the boundary conditions, is denoted L_h and it can be represented as an $N \times N$ matrix. The values $-\nabla \cdot (c^2(x)\nabla w)$ are then approximated by $L_h w^n$. As in the continuous case, we assume L_h has the eigenmodes (λ_j^2, ϕ_j), such that $L_h \phi_j = \lambda_j^2 \phi_j$ for $j = 1, \ldots, N$, where all λ_j are strictly positive and ordered as $0 \leq \lambda_1 \leq \ldots \leq \lambda_N$.

We let the Helmholtz solution u be given

$$-L_h u + \omega^2 u = f.$$

The numerical approximation of the iteration operator is denoted Π_h, and it is implemented as follows. Given $v \in \mathbb{R}^N$, we use the leap frog method to solve the wave equation as

$$w^{n+1} = 2w^n - w^{n-1} - \Delta t^2 L_h w^n - \Delta t^2 f \cos(\omega t_n), \tag{2.15}$$

with time step $\Delta t = T/M$ for some integer M, and initial data

$$w^0 = v, \qquad w^{-1} = v - \frac{\Delta t^2}{2}(L_h v + f).$$

The trapezoidal rule is then used to compute $\Pi_h v$,

$$\Pi_h v = \frac{2\Delta t}{T} \sum_{n=0}^{M} \eta_n \left(\cos(\omega t_n) - \frac{1}{4} \right) w^n, \qquad \eta_n = \begin{cases} \frac{1}{2}, & n = 0 \text{ or } n = M, \\ 1, & 0 < n < M. \end{cases} \tag{2.16}$$

With these definitions we can prove

Theorem 2.1.4. *Suppose there are no resonances, such that $\delta_h = \min_j |\lambda_j - \omega|/\omega > 0$. Moreover, assume that Δt satisfies the stability and accuracy requirements*

$$\Delta t < \frac{2}{\lambda_N + 2\omega/\pi}, \qquad \Delta t \omega \leq \min(\delta_h, 1). \tag{2.17}$$

Then the fixed point iteration $v^{(k+1)} = \Pi_h v^{(k)}$ with $v^{(0)} = 0$ converges to v^∞ which is a solution to the discretized Helmholtz equation with the modified frequency $\tilde{\omega}$,

$$-L_h v^\infty + \tilde{\omega}^2 v^\infty = f, \qquad \tilde{\omega} = 2\frac{\sin(\Delta t \omega/2)}{\Delta t}.$$

The convergence rate is at least $\rho_h = \max(1 - 0.3\delta_h^2, 0.6)$.

Proof. We expand all functions in eigenmodes of L_h,

$$w^n = \sum_{j=1}^{N} \hat{w}_j^n \phi_j, \qquad f = \sum_{j=1}^{N} \hat{f}_j \phi_j, \qquad u = \sum_{j=1}^{N} \hat{u}_j \phi_j, \qquad v = \sum_{j=1}^{N} \hat{v}_j \phi_j, \qquad v^\infty = \sum_{j=1}^{N} \hat{v}_j^\infty \phi_j.$$

Then the Helmholtz eigenmodes of u and v^∞ satisfy

$$\hat{u}_j = \frac{\hat{f}_j}{\omega^2 - \lambda_j^2}, \qquad \hat{v}_j^\infty = \frac{\hat{f}_j}{\tilde{\omega}^2 - \lambda_j^2}.$$

We note that $\tilde{\omega}$ is not resonant and $\hat{v}_j^\infty$ is well-defined for all j, since by (5) and (3.25)

$$|\tilde{\omega} - \lambda_j| \geq |\omega - \lambda_j| - |\tilde{\omega} - \omega| \geq \omega\delta_h - \frac{\Delta t^2 \omega^3}{24} \geq \omega \left(\delta_h - \frac{1}{24} \min(\delta_h, 1)^2 \right) > 0.$$

675 The wave solution eigenmodes are given by the difference equation

$$\hat{w}_j^{n+1} - 2\hat{w}_j^n + \hat{w}_j^{n-1} + \Delta t^2 \lambda_j^2 \hat{w}_j^n = -\Delta t^2 \hat{f}_j \cos(\omega t_n). \tag{2.18}$$

with initial data

$$\hat{w}_j^0 = \hat{v}_j, \qquad \hat{w}_j^{-1} = \hat{v}_j \left(1 - \frac{1}{2}\Delta t^2 \lambda_j^2 \right) - \frac{1}{2}\Delta t^2 \hat{f}_j.$$

By (3.25)

$$|2 - \Delta t^2 \lambda_j^2| < 2,$$

676 and the characteristic polynomial for the equation, $r^2 + (\Delta t^2 \lambda_j^2 - 2)r + 1$, then has two roots on
677 the boundary of the unit circle. The solution is therefore stable and is given by (the verification
678 of which is found in Appendix .2)

$$\hat{w}_j^n = (\hat{v}_j - \hat{v}_j^\infty) \cos(\tilde{\lambda}_j t_n) + \hat{v}_j^\infty \cos(\omega t_n), \tag{2.19}$$

where $\tilde{\lambda}_j$ is well-defined by the relation

$$2\frac{\sin(\Delta t \tilde{\lambda}_j / 2)}{\Delta t} = \lambda_j.$$

Now, let

$$\Pi_h v = \sum_{j=1}^{\infty} \bar{v}_j \phi_j.$$

Then the numerical integration gives

$$\bar{v}_j = \frac{2\Delta t}{T}\sum_{n=0}^{M}\eta_n\left(\cos(\omega t_n)-\frac{1}{4}\right)\left((\hat{v}_j-\hat{v}_j^\infty)\cos(\tilde{\lambda}_j t_n)+\hat{v}_j^\infty\cos(\omega t_n)\right)$$
$$= (\hat{v}_j-\hat{v}_j^\infty)\beta_h(\tilde{\lambda}_j)+\hat{v}_j^\infty\beta_h(\omega) = \hat{v}_j\beta_h(\tilde{\lambda}_j)+(1-\beta_h(\tilde{\lambda}_j))\hat{v}_j^\infty,$$

where

$$\beta_h(\lambda) = \frac{2\Delta t}{T}\sum_{n=0}^{M}\eta_n\cos(\lambda t_n)\left(\cos(\omega t_n)-\frac{1}{4}\right),$$

and we used the fact that the trapezoidal rule is exact, and equal to one, when $\lambda = \omega$. (Recall that for periodic functions the trapezoidal rule is exact for all pure trigonometric functions of order less than the number of grid points.) Hence, if $|\beta_h(\tilde{\lambda}_j)| < 1$ the j-th mode in the fixed point iteration converges to $\hat{v}_j^\infty$. This is ensured by the following lemma, the proof of which is found in Appendix .3.

Lemma 2.1.5. *Under the assumptions of Theorem 2.1.4,*

$$\max_{1\le j\le N}|\beta_h(\tilde{\lambda}_j)| \le \rho_h =: \max(1-0.3\delta_h^2, 0.63). \tag{2.20}$$

Since the bound $|\beta_h(\tilde{\lambda}_j)| \le \rho_h < 1$ in the lemma is uniform for all j the convergence $v^{(k)} \to v^\infty$ with rate at least ρ_h follows. This concludes the proof of the theorem.

□

Remark 2.1.6. *The discretization above is used as an example to illustrate the impact of going from the continuous to the discrete iteration. For a particular discretization we can improve the iteration further by using the knowledge of how it approximates ω and the eigenvalues of the continuous operator. Indeed, for the discretization above, let us define $\bar{\omega}$ by the relation*

$$\omega = 2\frac{\sin(\Delta t\bar{\omega}/2)}{\Delta t}.$$

Then if we use $f\cos(\bar{\omega}t_n)$ instead of $f\cos(\omega t_n)$ in the time stepping (2.15), the limit will be precisely the Helmholtz solution, $v^\infty = u$. Furthermore, the condition $\Delta t\omega \le \min(\delta_h, 1)$ can be quite restrictive for problems close to resonance. It is only important to ensure convergence of the iterations. Another way to do that is to slightly change the discrete filter by replacing the constant $1/4$

in (3.24) by a Δt-dependent number such that $|\beta_h(\lambda)| < 1$ for $\lambda \neq \omega$. Another option is to use a higher order quadrature rule, which would mitigate the restriction on Δt.

2.1.3 Tunable Filters

In Lemma 2.1.1 we saw that the filter transfer function satisfies $\beta(\omega) = 1$ and $-1/2 < \beta(r) < 1$ when $r \neq \omega$ and that these conditions guaranteed convergence of the WaveHoltz iteration. To improve convergence when $r \approx \omega$ we now consider a more general filter transfer function

$$\bar{\beta}(\lambda) = \frac{2}{T}\int_0^T (\cos(\omega t) + \alpha(t))\cos(\lambda t)\, dt, \quad \alpha(t) = a_0 + \sum_{n=1}^{\infty} a_n \sin(n\omega t), \tag{2.21}$$

where we refer to $\alpha(t)$ as a *time-dependent shift*. . As before , necessary conditions for convergence are $\bar{\beta}(\omega) = 1$, $\bar{\beta}'(\omega) = 0$. Straightforward calculations reveal that these conditions require that the two first coefficients must satisfy

$$a_1 = \frac{1}{2\pi}(1 + 4a_0).$$

The remaining terms in the sum are orthogonal to $\cos(\lambda t)$ when $\lambda = \omega$. Carrying out the integration in full for each term yields the general form

$$\bar{\beta}(\lambda) = \frac{\lambda\omega\sin(\lambda T)}{\pi(\lambda^2 - \omega^2)} + a_0\frac{\omega\sin(\lambda T)}{\pi\lambda} + \sum_{n=1}^{\infty} a_n \frac{n\omega^2}{\pi(\lambda^2 - n^2\omega^2)}\left(\cos(\lambda T) - 1\right),$$

from which it follows that another necessary condition is $|a_0| < 1/2$ since $|\bar{\beta}(r)| < 1$ and

$$\bar{\beta}(0) = a_0 \lim_{\lambda\to 0} \frac{\omega\sin(2\pi\lambda/\omega)}{\pi\lambda} = 2a_0.$$

We note that the standard filter, where $a_0 = -1/4$ and $a_1 = 0$, satisfies the necessary conditions.

Remark 2.1.7. *For the remaining coefficients a_n we only need to ensure that $|\bar{\beta}(r)| < 1$ which leaves large freedom to design $\bar{\beta}$. For example we may try to maximize $|\bar{\beta}''(\omega)|$ (minimize $\bar{\beta}''(\omega)$) so that $\bar{\beta}(r)$ is sharply peaked around $r = \omega$. We do not pursue a systematic study of this here but illustrate the utility of the added flexibility of (2.21) with numerical experiments below in Section 2.3.*

2.1.4 Multiple Frequencies in One Solve

We can use the WaveHoltz algorithm to solve for multiple frequencies at once. Suppose we look for the solutions u_i of

$$\nabla \cdot (c^2(x)\nabla u_i) + \omega_i^2 u_i = f_i(x), \qquad i = 1, \ldots, N,$$

with the same c and boundary condition for all i. To find those solutions we include all frequencies in the wave equation part of the iteration (2.4), and solve

$$w_{tt} = \nabla \cdot (c(x)^2 \nabla w) - \sum_{i=1}^{N} f_i(x)\cos(\omega_i t). \tag{2.22}$$

We then seek a decomposition

$$w(x,t) \equiv \sum_{i=1}^{N} u_i(x)\cos(\omega_i t), \tag{2.23}$$

of the solution. The filtering part of the WaveHoltz iteration is also updated to reflect the multiple frequencies

$$v_{n+1} = \frac{2}{T}\int_0^T \left(\sum_{i=1}^{N} \cos(\omega_i t) - \frac{1}{4}\right) w(x,t)\, dt.$$

As before we take $v_0 = 0$ when we deal with energy conserving boundary conditions. To this end we assume that the frequencies are related by an integer multiple in a way so that the period T can be chosen based on the lowest frequency.

The different $u_i(x)$ in (2.23) can be found as follows. Once we have found the time periodic solution to (2.22) evolve one more period and sample $w(x,t)$ at N distinct times t_j, $j = 1, \ldots, N$. We then have

$$u_i(x) = \sum_{j=1}^{N} \beta_{ij} w(x, t_j),$$

where the coefficients β_{ij} are the elements of A^{-1} with the elements of A being $a_{ij} = \cos(\omega_j t_i)$.

2.1.5 WaveHoltz Iteration for Impedance Boundary Conditions

For impedance and other boundary conditions that leads to a decreasing energy for the wave equation we cannot make the simplifying assumption in (2.4) that $w_t(0,x) = 0$ but we must seek

both $v_0(x)$ and $v_1(x)$ in (2.4). To do so we define an extended iteration (2.8) where we apply Π to both the displacement and the velocity:

$$\begin{bmatrix} v \\ v' \end{bmatrix}^{(n+1)} = \tilde{\Pi} \begin{bmatrix} v \\ v' \end{bmatrix}^{(n)}, \qquad \begin{bmatrix} v \\ v' \end{bmatrix}^{(0)} \equiv 0, \tag{2.24}$$

where

$$\tilde{\Pi} \begin{bmatrix} v \\ v' \end{bmatrix} = \frac{2}{T} \int_0^T \left(\cos(\omega t) - \frac{1}{4} \right) \begin{bmatrix} w(t,x) \\ w_t(t,x) \end{bmatrix} dt, \quad T = \frac{2\pi}{\omega}. \tag{2.25}$$

Here $w(t,x)$ and its time derivative $w_t(t,x)$ satisfies the wave equation (2.4) with initial data $v_0(x) \equiv v^{(n)}$ and $v_1(x) \equiv v'^{(n)}$.

2.2 Wave Equation Solvers

In this section we briefly outline the numerical methods we use in the experimental section below. We consider both discontinuous Galerkin finite element solvers and finite difference solvers. In all the experiments we always use the trapezoidal rule to compute the integral in the WaveHoltz iteration.

2.2.1 The Energy Based Discontinuous Galerkin Method

Our spatial discretization is a direct application of the formulation described for general second order wave equations in [9, 10]. Here we outline the spatial discretization for the special case of the scalar wave equation in one dimension and refer the reader to [9] for the general case.

The energy of the scalar wave equation is

$$H(t) = \int_D \frac{v^2}{2} + G(x, w_x) dx,$$

where

$$G(x, w_x) = \frac{c^2(x) w_x^2}{2},$$

is the potential energy density, v is the velocity (not to be confused with the iterates v^n above) or the time derivative of the displacement, $v = w_t$. The wave equation, written as a second order

equation in space and first order in time then takes the form

$$\begin{aligned} w_t &= v, \\ v_t &= -\delta G - f(x)\cos(\omega t), \end{aligned}$$

where δG is the variational derivative of the potential energy

$$\delta G = -(G_{w_x})_x = -(c^2(x)w_x)_x.$$

For the continuous problem the change in energy is

$$\frac{dH(t)}{dt} = \int_D v v_t + w_t (c^2(x) w_x)_x \, dx = -\int_D v f(x)\cos(\omega t) dx + [w_t (c^2(x) w_x)]_{\partial D}, \tag{2.26}$$

where the last equality follows from integration by parts together with the wave equation. Now, a variational formulation that mimics the above energy identity can be obtained if the equation $v - w_t = 0$ is tested with the variational derivative of the potential energy. Let Ω_j be an element and $\Pi^s(\Omega_j)$ be the space of polynomials of degree s, then the variational formulation on that element is:

Problem 1. *Find $v^h \in \Pi^s(\Omega_j)$, $w^h \in \Pi^r(\Omega_j)$ such that for all $\psi \in \Pi^s(\Omega_j)$, $\phi \in \Pi^r(\Omega_j)$*

$$\int_{\Omega_j} c^2 \phi_x \left(\frac{\partial w_x^h}{\partial t} - v_x^h \right) dx = [c^2 \phi_x \cdot n \left(v^* - v^h \right)]_{\partial \Omega_j}, \tag{2.27}$$

$$\int_{\Omega_j} \psi \frac{\partial v^h}{\partial t} + c^2 \psi_x \cdot w_x^h + \psi f(x)\cos(\omega t)\, dx = [\psi \, (c^2 \, w_x)^*]_{\partial \Omega_j}. \tag{2.28}$$

Let $[[\zeta]]$ and $\{\zeta\}$ denote the jump and average of a quantity ζ at the interface between two elements, then, choosing the numerical fluxes as

$$\begin{aligned} v^* &= \{v\} - \tau_1 [[c^2 \, w_x]] \\ (c^2 \, w_x)^* &= \{c^2 \, w_x\} - \tau_2 [[v]], \end{aligned}$$

will yields a contribution $-\tau_1([[c^2 \, w_x]])^2 - \tau_2([[v]])^2$ from each element face. To this end we choose $\tau_i > 0$ (so called upwind or Sommerfeld fluxes) which together with the choice that the approximation spaces be of the same degree $r = s$ result in methods that are $r + 1$ order accurate in space

and measured in the L_2 norm. We note that even in the case of energy conserving numerical fluxes the formulation does not lead to a **symmetric** matrix for the WaveHoltz iteration (it is of course positive definite though).

Physical boundary conditions can also be handled by appropriate specification of the numerical fluxes, see [9] for details. The above variational formulation and choice of numerical fluxes results in an energy identity similar to (2.26). However, as the energy is invariant to certain transformations the variational problem does not fully determine the time derivatives of w^h on each element and independent equations must be introduced. In this case there is one invariant and an independent equation is $\int_{\Omega_j} \left(\frac{\partial w^h}{\partial t} - v^h \right) = 0$.

Denoting the degrees of freedom on element Ω_j by v_j and w_j the semi-discretization according to (2.27)-(2.28) on element Ω_j can be written

$$S(\frac{\partial w_j}{\partial t} - v_j) = L_1(v_{j-1}, v_j, v_{j+1}, w_{j-1}, w_j, w_{j+1}), \tag{2.29}$$

$$M\frac{\partial v_j}{\partial t} + Sw_j + f_j \cos(\omega t) = L_2(v_{j-1}, v_j, v_{j+1}, w_{j-1}, w_j, w_{j+1}), \tag{2.30}$$

where the elements of the element matrices M and S are $M_{kl} = \int_{\Omega_j} \phi_x \phi_l dx$ and $S_{kl} = \int_{\Omega_j} c^2 (\phi_k)_x (\phi_l)_x dx$ respectively and the lift operators L_1 and L_2 represents the numerical fluxes. Note that a convenient way to directly enforce the independent equation is to compute the time derivatives of w_j according to

$$\frac{\partial w_j}{\partial t} = v_j = S^\dagger L_1(v_{j-1}, v_j, v_{j+1}, w_{j-1}, w_j, w_{j+1}),$$

where $S^\dagger$ is the pseudo inverse of S.

2.2.2 Finite Difference Discretizations

For the finite difference examples we exclusively consider Cartesian domains $(x, y, z) \in [L_x, R_x] \times [L_y, R_y] \times [L_z, R_z]$ discretized by uniform grids $(x_i, y_j, z_k) = (L_x + ih_x, L_y + jh_y, L_z + kh_z)$, with $i = 0, \ldots, n_x$ and $h_x = (R_x - L_x)/n_x$, etc.

When we have impedance boundary conditions on the form $w_t \pm \vec{n} \cdot \nabla w = 0$ we evolve the wave equation as a first order system in time according to the semi-discrete approximation

$$\frac{dv_{ijk}(t)}{dt} = (D_+^x D_-^x + D_+^y D_-^y + D_+^z D_-^z) w_{ijk}, \tag{2.31}$$

$$\frac{dw_{ijk}(t)}{dt} = v_{ijk}, \tag{2.32}$$

for all grid points that do not correspond to Dirichlet boundary conditions. On boundaries with impedance conditions we find the ghost point values by enforcing (here illustrated on the top of the domain)

$$v_{ijn_z} - D_0^z w_{ijn_z} = 0. \tag{2.33}$$

Here we have used the standard forward, backward and centered finite difference operators, for example $h_x D_+^x w_{i,j,k} = w_{i+1,j,k} - w_{i,j,k}$ etc. For problems with variable coefficients the above discretization is generalized as in [12].

We note that in some of the examples where we require high order accuracy we use the summation by parts discretization for variable coefficients developed by Mattson in [86] and described in detail there.

2.2.3 Time Discretization

In most of the numerical examples we use either an explicit second order accurate centered discretization of w_{tt} (for finite differences with energy conserving boundary conditions we eliminate v and time discretize w_{tt} directly as in the analysis in Section 2.1.2) or the classic fourth order accurate explicit Runge-Kutta method.

For some of the DG discretizations we employ Taylor series time-stepping in order to match the order of accuracy in space and time. Assuming that all the degrees of freedom have been assembled into a vector $\mathbf{w}$ we can write the semi-discrete method as $\mathbf{w}_t = Q\mathbf{w}$ with Q being a matrix representing the spatial discretization. Assuming we know the discrete solution at the time

t_n we can advance it to the next time step $t_{n+1} = t_n + \Delta t$ by the simple formula

$$\begin{aligned} \mathbf{w}(t_n + \Delta t) &= \mathbf{w}(t_n) + \Delta t \mathbf{w}_t(t_n) + \frac{(\Delta t)^2}{2!}\mathbf{w}_{tt}(t_n)\dots \\ &= \mathbf{w}(t_n) + \Delta t Q\mathbf{w}(t_n) + \frac{(\Delta t)^2}{2!}Q^2\mathbf{w}(t_n)\dots \end{aligned}$$

The stability domain of the Taylor series which truncates at time derivative number N_{T} includes part of the imaginary axis if $\mathrm{mod}(N_{\mathrm{T}}, 4) = 3$ or $\mathrm{mod}(N_{\mathrm{T}}, 4) = 0$ (see e.g. [72]) . However as we use a slightly dissipative spatial discretization the spectrum of our discrete operator will be contained in the stability domain of all sufficiently large choices of N_{T} (i.e. the N_{T} should not be smaller than the spatial order of approximation). Note also that the stability domain grows linearly with the number of terms.

2.3 Numerical Examples

In this section we illustrate the properties of the proposed iteration and its Krylov accelerated version by a sequence of numerical experiments in one, two and three dimensions.

2.3.1 Examples in One Dimension

We begin by presenting some very basic numerical experiments in one dimension.

2.3.1.1 Convergence of Different Iterations / Solvers at a Fixed Frequency

We start by repeating the example described in Section 3.5 in [61]. This example is used in [61] to illustrate that the original cost functional from [29] (denoted J in [61]) does not yield the correct solution due to the existence of multiple minimizers.

The example solves the Helmholtz equation with $c = 1$ and with the exact solution

$$u(x) = 16x^2(x-1)^2, \quad 0 \le x \le 1.$$

Here both u (and w_t for the time-dependent problem) and u_x vanish at the endpoints so any boundary condition of the form

$$\alpha w_t + \beta(\vec{n} \cdot w_x) = 0, \quad \alpha^2 + \beta^2 = 1,$$

will be satisfied. Dirichlet boundary conditions correspond to $\alpha = 1$ and Neumann boundary conditions correspond to $\alpha = 0$, all other values will be an impedance boundary condition. Here, as in [61], we take the frequency to be $\omega = \pi/4$.

We discretize using the energy based DG method discussed above and use upwind fluxes which adds a small amount of dissipation. For this experiment we use 5 elements with degree $q = 7$ polynomials and we use an 8th order accurate Taylor series method in time. We set Δt so that $n_t \Delta t = T = 2\pi/\omega$ while making the inequality $\Delta t \leq C_{\text{CFL}} \Delta x/(q+1)$ as sharp as possible (in this experiment we fix $C_{\text{CFL}} = 1/2$). With this resolution in space and time the truncation errors are negligible and we expect that the observed convergence properties should match those of the continuous analysis.

Method / b. c.	WHI	LSQR	QMR	CG	GMRES
D-D	94.5(-15)	76.1(-15)	75.9(-15)	151.5(-15)	97.9(-15)
N-N	49.2(-15)	142.8(-15)	144.4(-15)	158.5(-15)	144.1(-15)
D-N	28.3(-15)	55.4(-15)	81.9(-15)	272.3(-15)	67.0(-15)

Table 2.1: Maximum error for various combinations of boundary conditions and methods.

As mentioned above we expect that our method works best when combined with a classical iterative Krylov subspace method. The energy based DG method will produce a matrix A with real eigenvalues in $(0, 3/2)$ but it will not yield a symmetric matrix A. We present results for the WaveHoltz iteration (denoted WHI in figures and tables), and its acceleration with Matlab implementations of LSQR, QMR, CG and GMRES (we use the default unconditioned settings with a tolerance of 10^{-13}). In Figure 2.2 we display the convergence histories for various combinations of boundary conditions. The residuals for the Krylov accelerated iterations are the ones returned by the Matlab functions and the residual for the WaveHoltz iteration is simply the L_2 norm of the difference between two subsequent iterations. As can be seen the convergence behavior for QMR and GMRES are uniformly the fastest and appears to be insensitive to the type of boundary condition used. Note that the numerical method used here does not yield a symmetric matrix and

CG is not guaranteed to work. Evidence of this loss or stagnation of convergence can be found in the cases D-D and D-N in Figure 2.2.

The actual errors in the converged solutions can be found in Table 2.1, where it can be seen that the error for all of the iteration methods are close to the residual tolerance.

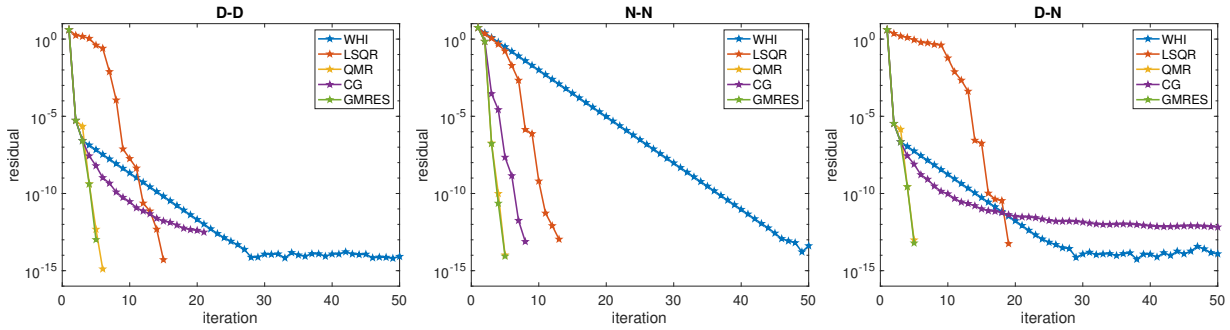


Figure 2.2: Convergence of the residual for the plain WaveHoltz iteration and its accelerated versions using LSQR, QMR, CG and GMRES. The titles of the figures indicate the boundary conditions used to the left and right, e.g. D-N means Dirichlet on the left and Neumann on the right.

2.3.1.2 Convergence with Increasing Frequency

To study how the number of iterations scale with the Helmholtz frequency ω we solve the wave equation on the domain $x \in [-6, 6]$ with constant wave speed $c^2(x) = 1$ and with a forcing

$$f(x) = \omega^2 e^{-(\omega x)^2},$$

that results in the solution being $\mathcal{O}(1)$ for all ω. The solver is the same as in the previous example. We keep the number of degrees of freedom per wave length fixed by letting the number of elements be $5\lceil\omega\rceil$. We always take the polynomial degree to be 7 and the number of Taylor series terms in the timestepping to be 8. As we now also consider impedance boundary conditions, with $\alpha = 1/2$, we use WHI accelerated by GMRES.

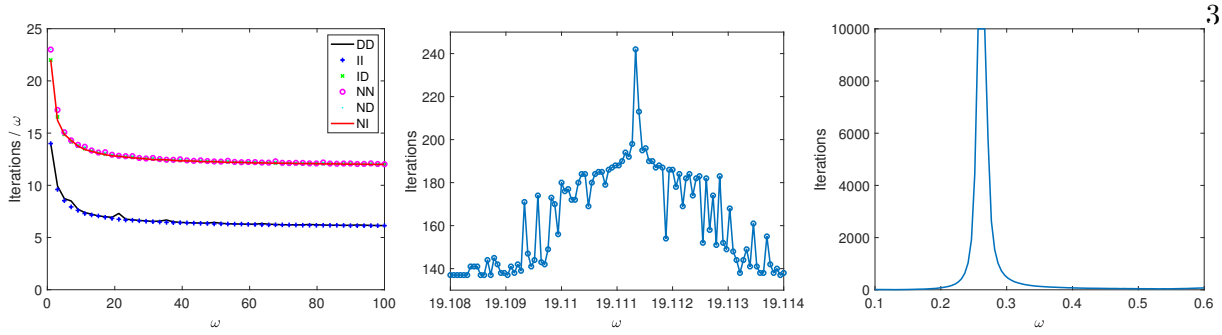


Figure 2.3: Left: Number of iterations divided by ω as a function of ω for different boundary conditions. Middle and right: Zoom in around a resonance for the Dirichlet problem when using Krylov acceleration (middle) and when using WHI (right).

We report the number of iterations it takes to reach a GMRES residual smaller than 10^{-10} for the six possible combinations of Dirichlet, Neumann and impedance boundary conditions for 50 frequencies distributed evenly between 1 and 100. The results are displayed to the left in Figure 2.3 where we plot the number of iterations divided by ω as a function of ω. It is clear that the asymptotic scaling is linear with growing frequency. Interestingly all the combinations of boundary conditions collapse to two different curves with the Dirichlet-Dirichlet and impedance-impedance conditions converging the fastest.

We know from the analysis in Section 2.1.1 that the rate of convergence of the WaveHoltz iteration deteriorates near resonant frequencies (for non-impedance problems) but from Figure 2.3 it appears that all frequencies converge more or less the same rate. To study the behavior of the accelerated algorithm for homogenous Dirichlet boundary conditions we zoom in around $\omega \approx 19.114$ where the continuous problem has a resonance. In the middle graph in Figure 2.3 we display the required number of iterations around the resonant frequency. As can be seen there is some deterioration but only in very narrow band around a frequency that is slightly less than 19.114 and probably is the modified resonant frequency discussed in Section 2.1.2. This behavior can be contrasted to the growth of the number of iterations for the WaveHoltz iteration without GMRES acceleration, see the right figure in Figure 2.3. Clearly the acceleration of the WaveHoltz iteration by GMRES improves the robustness of the method near resonances.

2.3.1.3 Multiple Frequencies in One Solve

Here we illustrate the technique described in Section 2.1.4 for finding solutions of multiple frequencies at once. We set $\omega = 1$ and $\omega_j = 2^{j-1}\omega$ for $j = 1, \ldots, 4$, and consider the domain $x \in [0, 1]$. We use the finite difference discretization discussed in Section 2.2.2 with Dirichlet boundary conditions . The time evolution is done by a second order centered discretization of w_{tt}, as was done in the discrete analysis in Section 2.1.2 . The problem is forced by a point source centered at $x = 1/2$ for $j = 1, \ldots, 4$, and we consider a constant wave speed $c^2(x) = 1$. We display the convergence with decreasing h in Figure 2.4 on the right, where it can be seen that each solution u_j converges at a rate of h^2.

2.3.1.4 Tunable Filters

Here we consider solving a Helmholtz problem in the domain $x \in [0, 1]$ with Dirichlet boundary conditions and constant wave speed $c^2 = 1$. The discretization is the same as in the previous experiment and we use a point source centered at $x = 1/2$. A straightforward calculation shows that the resonant frequencies of the problem are integer multiples of π and we specifically consider solving the Helmholtz problem with frequency $\omega = 4.1\pi$, which has a minimum relative gap to resonance of $\delta = 1/41 \approx 0.024$. As discussed previously, we expect that the convergence rate of the WaveHoltz iteration will stagnate since ω is close to resonance. We compare the convergence against the problem with frequency $\omega = 1.5\pi$ which has a minimum relative gap to resonance of $\delta = 1/3$. The iteration history is displayed in Figure 2.4 on the left .

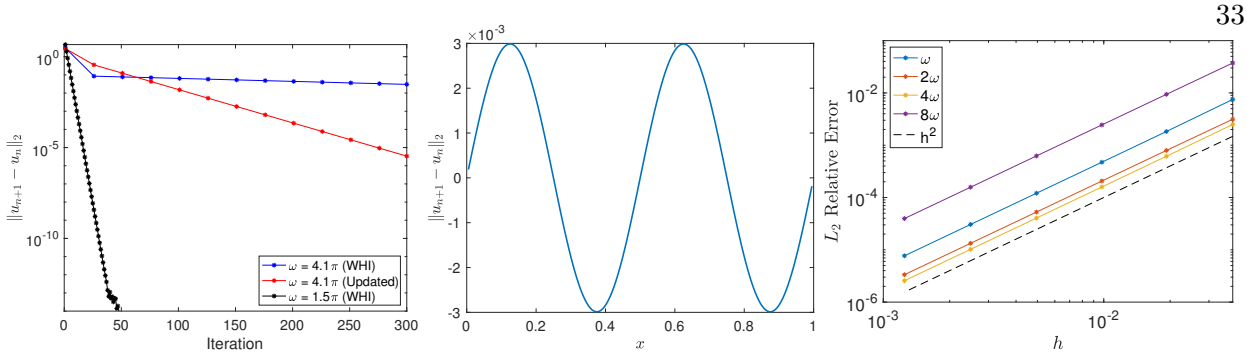


Figure 2.4: Left: Convergence history of the near resonant frequency 4.1π for the WaveHoltz filter and a tunable filter, and that of the frequency 1.5π for reference. Middle: The error between successive WaveHoltz iterates with the usual WaveHoltz filter. Right: Convergence of the solution for the CG accelerated WaveHoltz iteration with a point forcing.

It can be seen that the usual WaveHoltz iteration converges rapidly for the frequency 1.5π but that of 4.1π stagnates considerably. In the middle of Figure 2.4 we display the difference between successive WaveHoltz iterates for the Helmholtz problem with frequency ω, from which it is clear that the residual is a scaling of the resonant mode $\sin(4\pi x)$.

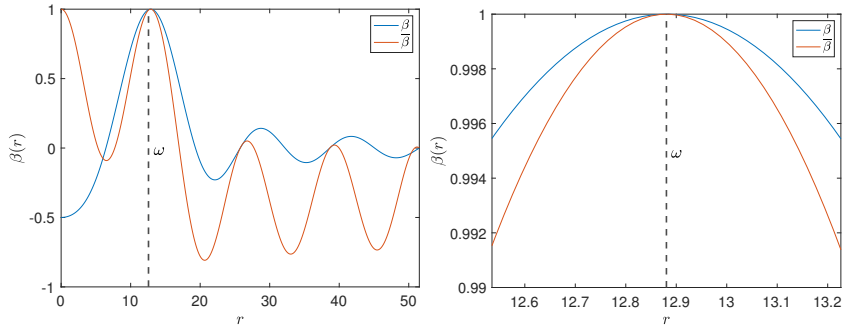


Figure 2.5: (Left) The usual WaveHoltz filter (in blue) and updated tunable filter (in red). (Right) Closeup of both the usual WaveHoltz filter and the updated tunable filter near the resonant frequency 4π.

To improve the rate of convergence close to resonance we leverage a tunable filter as mentioned in Section 2.1.3. To obtain this filter, we consider the filter transfer function (2.21) and truncate

the sin expansion of the time-dependent shift $\alpha(t)$ to 12 terms such that $a_n = 0$ for $n > 11$. In this example we take the usual choice of $a_0 = -1/4$ for the constant term in the filter transfer function (2.21) which, as discussed in Section 2.1.3, requires $a_1 = 0$. We then perform a minimization over a discrete set of 3000 equispaced points $r_j \in [0, 16\pi]$ of the empirically constructed functional

$$J(a_2, a_3, \ldots, a_{11}) = 10.6\bar{\beta}''(\omega) + 0.1 \sum_{|r_j - \omega| > 0.1} |\bar{\beta}(r_j)|^{20}, \tag{2.34}$$

via 100 steepest descent iterations. The first term in the functional (2.34) minimizes the second derivative at the peak $\omega = 4.1\pi$, while the second weakly enforces that $|\beta(r)| \leq 1$ for all $r > 0$ to ensure convergence of the fixed point iteration.

In Figure 2.5 on the right we see that the updated filter is steeper near $\omega = 4.1\pi$ so that repeated application of the updated filter will more quickly remove the resonant mode with frequency 4π and we thus expect faster convergence. This is confirmed in the resulting iteration history of the updated filter, shown in Figure 2.4 on the left . The cost of improving convergence behavior near resonance, however, is a larger value of $\bar{\beta}$ for many other modes as shown in Figure 2.5 on the left . A more careful investigation of optimized filters is left for the future.

2.3.2 Problems in Two Dimensions

In this section we present experiments in two space dimensions.

2.3.2.1 Convergence in Different Geometries

In this example we solve the Helmholtz equation with a constant wave speed, $c^2 = 1$, in the domain $(x, y) \in [-1, 1]^2$ and with forcing

$$f(x, y) = -\omega^2 e^{-\sigma[(x-0.01)^2 + (y-0.015)^2]},$$

where $\sigma = \max(36, \omega^2)$. We vary the frequency according to $\omega = 1/2 + k$, $k = 1, \ldots, 100$, and keep the number of points per wavelength roughly constant by choosing $n_x = n_y = 8\lceil\omega\rceil$. Here we use the finite difference method outlined in Section 2.2.2 combined with the classic fourth order Runge-Kutta method in time with a timestep $\Delta t = h_x/c$.

For each frequency we solve six different problems consisting of combinations of Dirichlet and impedance boundary conditions with zero to four open sides and with the two open boundary case forking into two cases: (1) the open boundaries are opposite each other, or (2) next to each other forming a corner. In Figure 2.6 we display the real part of the solution for the frequency $\omega = 77.5$ for the six different problems.

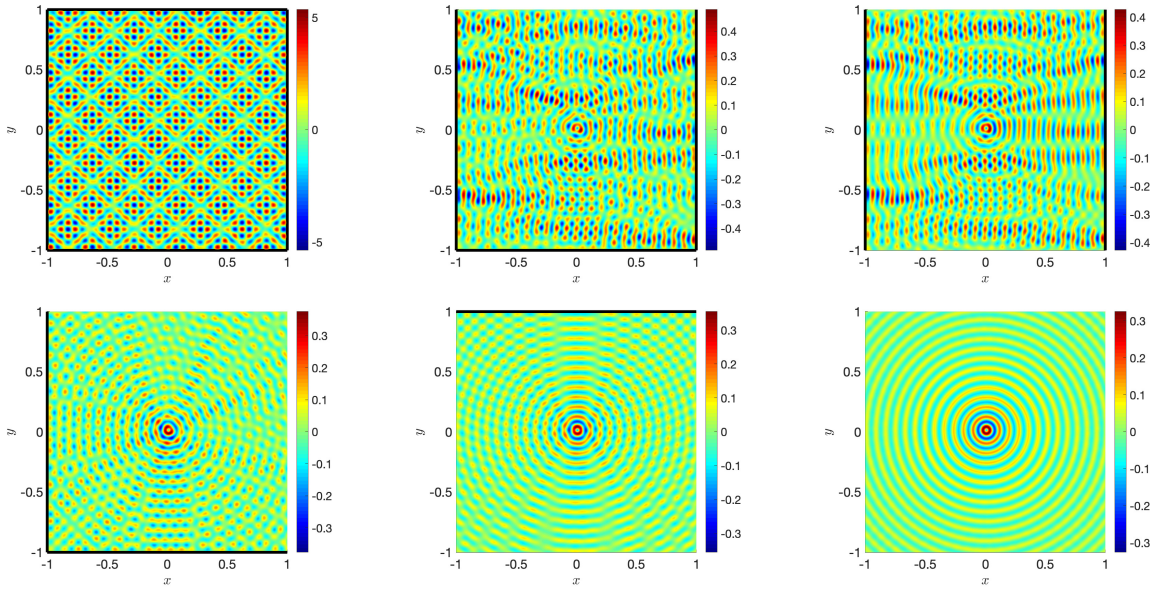


Figure 2.6: Typical solutions computed with the GMRES accelerated WHI at $\omega = 77.5$. The thick lines indicate Dirichlet boundary conditions.

In this example, the WaveHoltz iteration is accelerated by GMRES without restarts. Given that the storage requirement for GMRES grows with the number of iterations, it is often beneficial (especially for high frequency problems) to integrate and average over several periods to allow further propagation of information within the domain while mitigating the rapid growth of the Krylov subspace. For this example we thus choose to perform the WaveHoltz iteration with an integration time of 10 periods (i.e. we choose $T = 10\frac{2\pi}{\omega}$). In Figure 2.7 we report the number of iterations needed to reduce the relative residual below 10^{-7}. It is clear from the results that the geometries where the waves can get trapped are considerably more difficult and requires more

iterations. The computational results appear to indicate that the number of iterations to reach the tolerance scale as $\omega^{1.55}$ for the inner Dirichlet problem and similarly for the waveguide and the case with three Dirichlet boundary conditions. As the frequency increases and the distance between resonant frequencies decreases the iteration is not able to reduce the relative residual below the tolerance 10^{-7} within the prescribed maximum 1000 iterations.

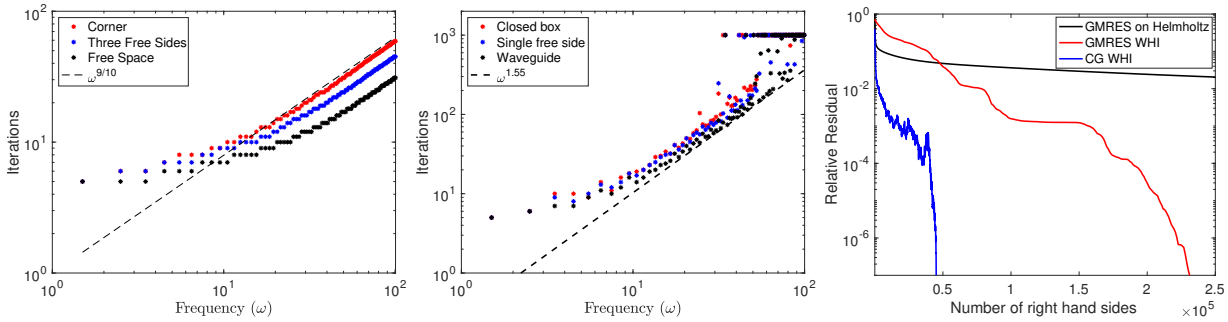


Figure 2.7: To the left: number of iterations as a function of frequency to reduce the relative residual below 10^{-7} for problems with no trapped waves. Middle: the same but for problems with trapped waves and for the interior problem. Both are with the GMRES accelerated WHI . To the right: Residuals for the GMRES accelerated WHI , the CG accelerated WHI and for GMRES solution of the directly discretized Helmholtz problem.

On the other hand for geometries with no trapped waves we see faster convergence (see the left figure in Figure 2.7) with the number of iterations scaling roughly as $\omega^{9/10}$.

To the right in Figure 2.7 we display the residual as a function of the number of right hand side evaluations (for the wave equation this is equivalent to taking a timestep and for the direct discretization of Helmholtz this is equivalent to one application of the sparse system matrix, the cost of these are roughly equivalent) when $\omega = 51.5$ for the pure Dirichlet boundary condition problem. The three different results are for: 1. the WaveHoltz acceleration, 2. the WaveHoltz iteration accelerated with conjugate gradient and based on the same spatial discretization but with a second order accurate centered discretization of w_{tt} using $\Delta t = 0.7h_x$ and, 3. a direct discretization of the Helmholtz equation (using the spatial discretization described in Section 2.2.2) combined with

GMRES for solving the resulting system of equations. Precisely we use GMRES with restart every 100 iterations. For space reasons we only display this for one frequency but note that although the results may differ a bit between frequencies the trend is similar in the problems we have investigated.

It is clear from the residuals that both the GMRES and conjugate gradient accelerated WaveHoltz iterations are radically faster than applying GMRES to the direct discretization of Helmholtz. As all the methods use the same spatial discretization this is an indication of the importance of changing the problem from an indefinite system of equations to a positive definite and to a symmetric positive definite system.

Remark 2.3.1. *We note that the problems considered in this experiment can be naturally solved with integral equation techniques since the the wave speed is constant. In addition as the problem is posed in two dimensions and can be stored in memory a good sparse solver will also be a very good alternative. What we want to demonstrate is: 1. The positive definiteness of the accelerated WHI makes it faster than standard iterative techniques for the direct discretization of Helmholtz, 2. The complexity is different for open and closed problems as predicted by the theory in [42].*

2.3.2.2 Smoothly Varying Wave Speed in an Open Domain

In this example we consider a smoothly varying medium in a box $(x, y) \in [-1, 1]^2$. The wave speed is

$$c^2(x, y) = 1 - 0.4e^{-\left(\frac{x^2+y^2}{0.25^2}\right)^4},$$

and is also depicted in Figure 2.8.

Here we use the energy based DG solver and impose a right going plane wave $e^{i\omega(t-x)}$ through impedance boundary conditions on the left, bottom and top faces of the domain. On the right boundary we impose a zero Dirichlet condition.

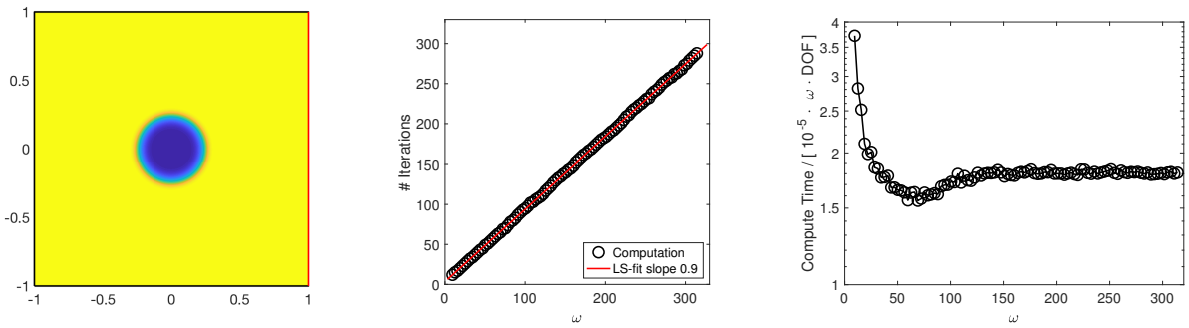


Figure 2.8: Left: the speed of sound (squared) used in example 2.3.2.2. Red indicates a rigid wall and black indicates open walls. Middle: Number of iterations as a function of frequency. Right: Compute time normalized by the frequency times the number of degrees of freedom.

In all the computations we use degree 5 polynomials and a 6th order Taylor series method. The elements used form a Cartesian structured grid and we scale the number of elements so that we have 8 degrees of freedom per wavelength. The WHI is applied with an integration time of 5 periods and is accelerated by GMRES with a termination tolerance 10^{-7} on the relative residual.

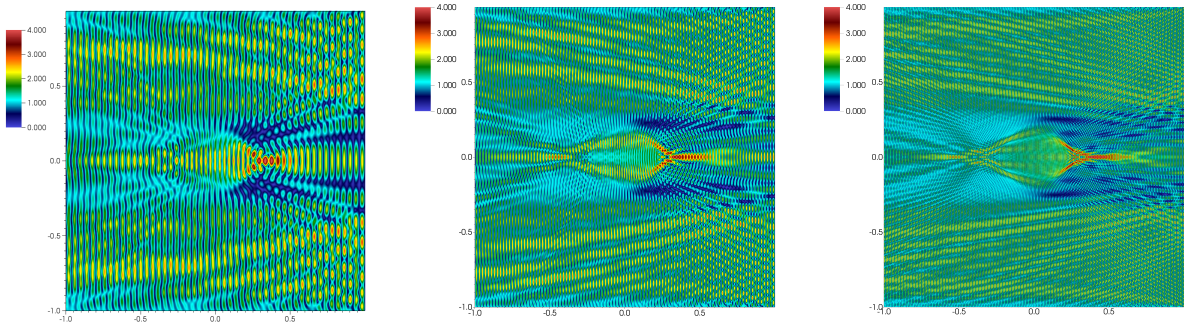


Figure 2.9: The magnitude of the Helmholtz solution for, from left to right, $\omega = 25\pi, 50\pi$ and 100π.

We solve the Helmholtz problem with $\omega = k\pi, k = 3, 4, \ldots, 100$ and measure the total time from start to time of solution and we also measure the number of iterations needed to converge. The results, displayed in Figure 2.8, again show that for this type of open problem the iteration

appears to require $N_{\text{iter}} \sim \mathcal{O}(\omega^{0.9})$ iterations to converge to a fixed tolerance. In terms of total computational time we observe $T_{\text{Total}} \sim \mathcal{O}(\omega N_{\text{DOF}})$ which is slightly higher than what would be expected from the $\mathcal{O}(\omega^{0.9})$ behavior.

However, as the distance traveled by the wave solution is proportional to $cT = 2\pi c/\omega$ and the information must travel through the domain at least once the time to solution is as good as can be expected. To reduce the computational complexity further we would need to propagate the solution faster than the speed of sound by applying a preconditioner or some type of multi-level strategy. Although we believe this is possible we leave such attempts to future work.

The magnitude of the solutions with $\omega = 25\pi, 50\pi$ and 100π are plotted in Figure 2.9.

2.3.2.3 Convergence of the Approximation Error and the Residual

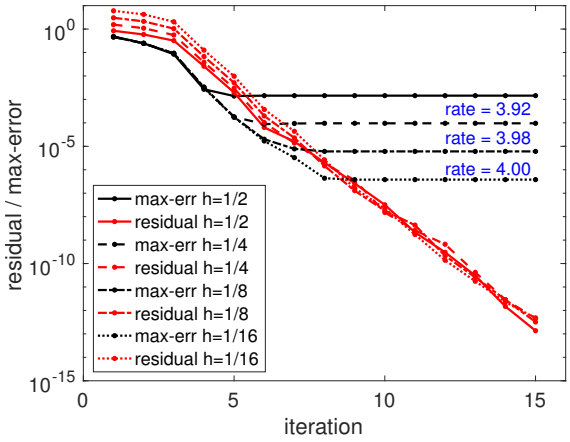


Figure 2.10: The maximum error GMRES residuals as a function of number of iterations for four different mesh sizes. The rates of convergence agree with the order of the method.

As our iteration leads to a linear system of equations (and consequently a different residual) we should check that the residual is still a suitable proxy for the discretization error. Although we have no reason to believe this would not be the case we note that we have not yet performed a detailed analysis and resort to checking this numerically. We consider the same computational domain and method as above but with speed of sound $c = 1$ and with zero Dirichlet boundary

conditions. We set $\omega = 2$ and choose the forcing so that the solution is

$$u = -(x^2 - 1)^2(y^2 - 1)^2,$$

and compute the solution using polynomials of degree three in the energy DG method and a fourth order accurate Taylor time stepper. In Figure 2.10 we display the maximum errors in u and the residuals for each GMRES iteration for Cartesian grids with grid spacings $1/2, 1/4, 1/8$ and $1/16$. As can be seen the residuals and the errors track well until the errors saturate. To the right in the figure we also indicate the rates of convergence based on the subsequent grid refinements. As expected they are very close to four.

2.3.2.4 The Marmousi2 Model

In the last two examples in this section we use the sixth order summation-by-parts finite difference operators developed by Mattson in [86]. Here we use the classic fourth order Runge-Kutta method for timestepping. In this example we simulate the solution caused by a point source placed in a material model where the speed of sound is taken from P-wave velocity in the Marmousi2 model[1] . We discretize the full model which consists of 13601×2801 grid points and covers a domain that is roughly 17×3.5 kilometers. On the top surface we prescribe a zero Dirichlet condition and on the remaining three sides we add a 50 grid point wide supergrid layer (see [8]) that is terminated by zero Dirichlet boundary conditions. We accelerate the WHI by the transpose free quasi minimal residual (TFQMR) method and terminate the iteration when the relative residual is below 10^{-5}. We perform each iteration over 8 periods and take 500 timesteps per iteration. The time periodic point forcing is applied near the surface in grid point $(6750, 2600)$ and we perform computations with $\omega = 200, 400$ and 800.

[1] `http://www.agl.uh.edu/downloads/downloads.htm`

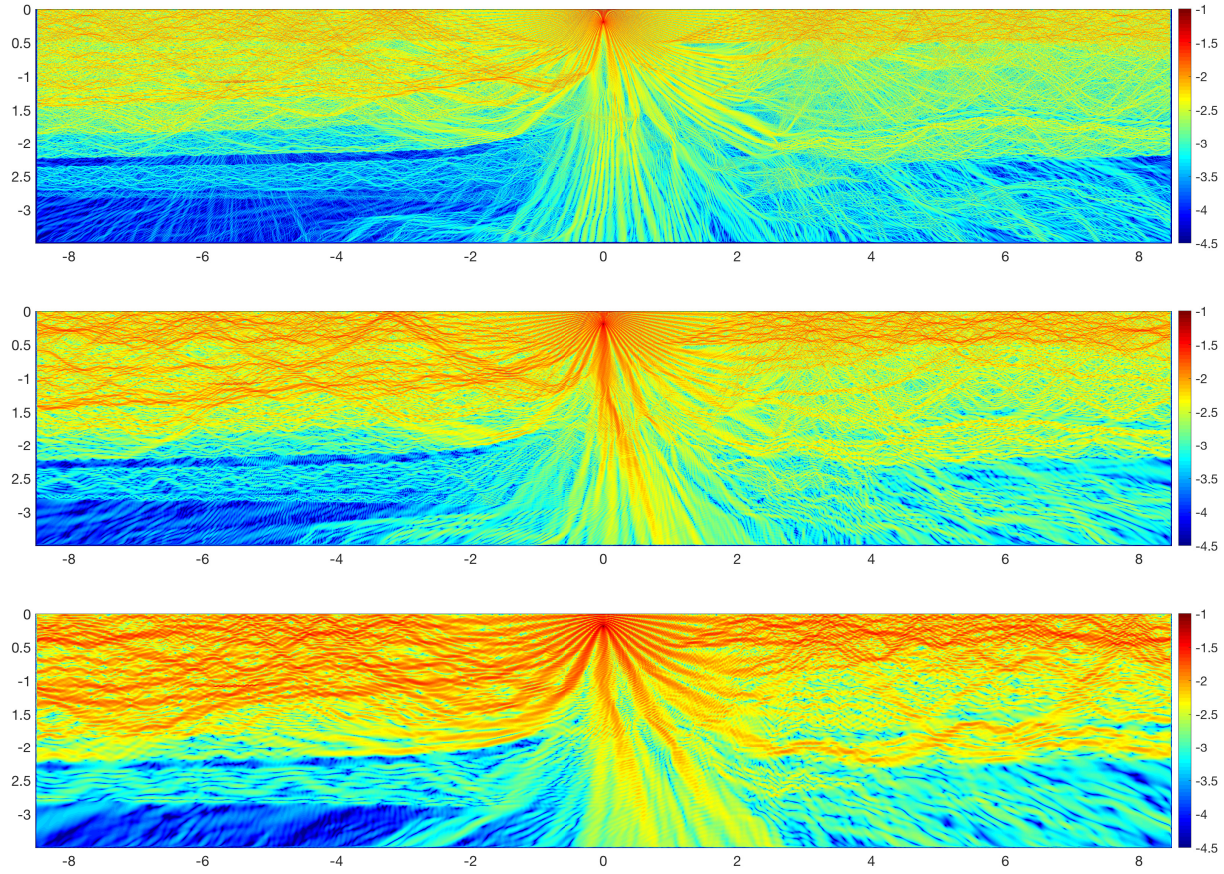


Figure 2.11: Displayed is the base 10 logarithm of the magnitude of the Helmholtz solution ($\log_{10}|u|$) caused by a point source near the surface. The results are, from top to botton, for $\omega = 800, 400$ and 200.

As the number of unknowns is relatively large, $\sim 76 \cdot 10^6$, we parallelize the finite difference solver by a straightforward domain decomposition with the communication handled by MPI. The simulations were carried out on Maneframe II at the Center for Scientific Computation at Southern Methodist University using 60 dual Intel Xeon E5-2695v4 2.1 GHz 18-core Broadwell processors with 45 MB of cache each and 256 GB of DDR4-2400 memory. The results displayed in Figure 2.11 and 2.12 illustrate the ability of the method to find solutions to large problems and at high frequencies.

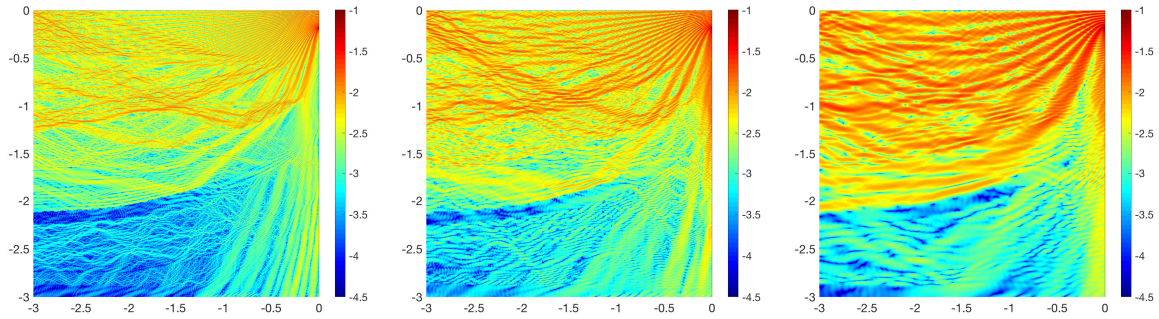


Figure 2.12: Zoom in of the base 10 logarithm of the magnitude of the Helmholtz solution ($\log_{10}|u|$) caused by a point source near the surface. The results are, from left to right, for $\omega = 800, 400$ and 200.

2.3.2.5 Multiple Frequencies

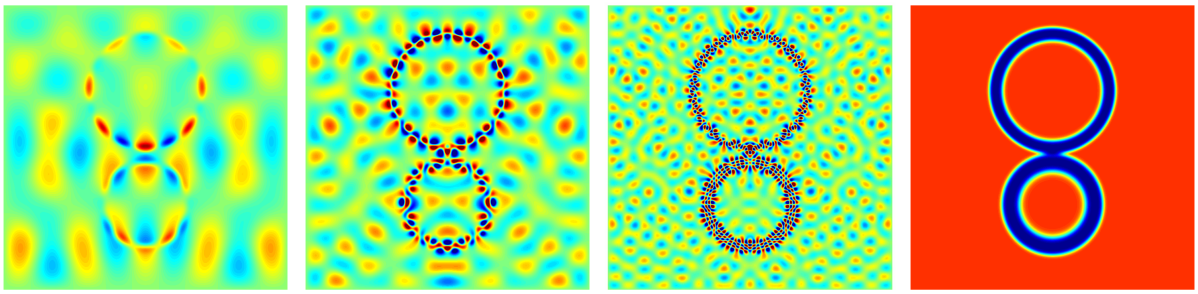


Figure 2.13: Computation of three Helmholtz problems by one solve. The frequencies are $\omega = 15, 30$ and 60. The material model is also displayed, red is $c^2 = 1$ and dark blue is $c^2 = 0.1$.

In this final example in two dimensions we again use the sixth order accurate summation-by-parts discretization from [86] with homogenous Dirichlet boundary conditions on the domain $(x, y) \in [-1, 1]^2$. The spatial discretization size is the same in both coordinates and is taken to be $2/300$. The velocity model is taken to be smoothly varying. Precisely we have that

$$c^2(x,y) = 1 - 0.9\left(e^{-\left(\frac{(x^2+(y-0.4)^2-0.4^2)}{0.2^2}\right)^4} + e^{-\left(\frac{(x^2+(y+0.4)^2-0.3^2)}{0.2^2}\right)^4}\right),$$

see also Figure 2.13. We consider three frequencies, $\omega = 15, 30, 60$, and use the same forcing in Helmholtz for all frequencies,

$$f(x,y) = \frac{\sigma}{\pi} e^{-\sigma(x^2+y^2)}, \quad \sigma = (4\omega)^2.$$

Here we use the WaveHoltz iteration over three periods of the lowest frequency, accelerated by GMRES (with tolerance 10^{-8}). We time step using a centered second order approximation to w_{tt} with a timestep $\Delta t = 1/600$. Since we solve for three frequencies at once we adjust the filter as described in Section 2.1.3 and extract all three solutions at once. Those solutions along with the material model are displayed in Figure 2.13.

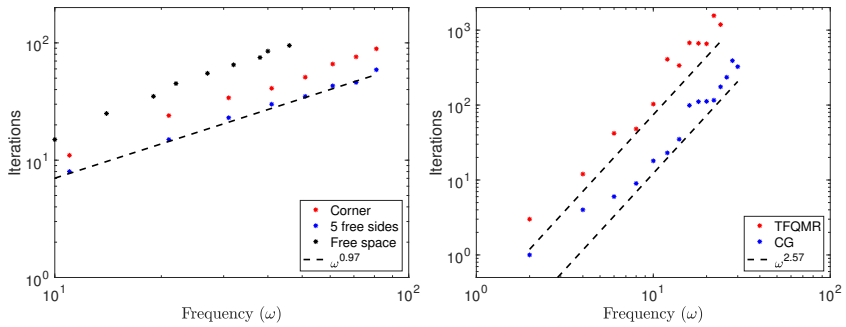


Figure 2.14: To the left: number of iterations as a function of frequency to reduce the relative residual below $5 \cdot 10^{-5}$ for problems with no trapped waves. Here WHI is accelerated by TFQMR. To the right: the same but for the interior problem. Here WHI is accelerated with either CG or TFQMR.

2.3.3 Problems in Three Dimensions

In this section we present experiments in three dimensions.

2.3.3.1 Convergence in Different Geometries

We solve the wave equation in a box $(x,y,z) \in [-1,1]^3$ with the smoothly varying medium

$$c^2(x,y,z) = 1 + \frac{1}{10} e^{-(x^2+y^2+z^2)}.$$

We use a uniform grid $(x_i, y_j, z_k) = (-1 + ih, -1 + jh, -1 + kh)$ with grid spacing $h = 1/n$ and choose $n = \max(\lceil 10\,\omega \rceil, 20)$ to keep the resolution fixed. The Helmholtz problem is forced by

$$F(x, y, z) = \omega^3 e^{-36\omega^2\left((x-x_0)^2+(y-y_0)^2+(z-z_0)^2\right)}, \tag{2.35}$$

where $x_0 = 1/100$, $y_0 = 3/250$, and $z_0 = 1/200$. We impose a mixture of boundary conditions consisting of homogenous Dirichlet and/or impedance boundary conditions: (1) impedance on all sides, (2) Dirichlet at $z = 1$ and impedance on all other sides, (3) Dirichlet at $z = -1$, $y = 1$, and $x = 1$ with impedance on all other sides, and (4) Dirichlet on all sides. We solve the equations in first order form in time and use the semi-discrete approximation described in Section 2.2.2.

In this example the WaveHoltz iteration is performed over 5 periods, numerically integrated in time with the classic Runge Kutta method of order four, and accelerated by TFQMR method. For the pure Dirichlet problem we also use CG but note that although the spatial discretization leads to a symmetric WHI matrix when combined with a centered finite difference approximation in time the matrix is only close to symmetric when combined with the slightly dissipative Runge Kutta method. The experiments indicate that this slight non-symmetry does not destroy the convergence iteration of CG in this case.

In Figure 2.14 we report the number of iterations needed to reduce the relative residual below $5 \cdot 10^{-5}$. As was seen before in the 2D case, the fully Dirichlet case is notably more difficult and requires more iterations than the other problems considered. The computational results indicate that the number of iterations to reach the tolerance scale as $\omega^{2.57}$ for the inner Dirichlet problem with either CG or TFQMR, with the former taking fewer overall iterations than the latter. By comparison, the set of problems with boundary conditions (1)-(3) listed above appear to converge in a number of iterations that scales as $\omega^{0.97}$, i.e. close to linear in the frequency ω.

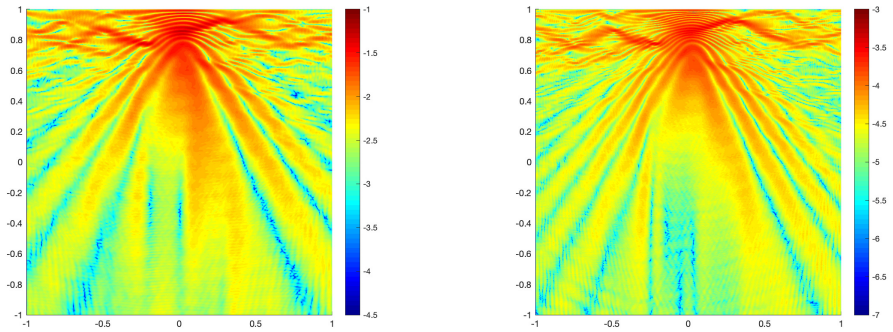


Figure 2.15: Displayed is the base 10 logarithm of the magnitude of the Helmholtz solution ($\log_{10}|u|$) caused by a point source near the surface for $\omega = 200$ (left) and $\omega = 300$ (right) at the slice $x = 0.1$.

2.3.3.2 Scattering from a Plate

For our final example, we again consider the box $(x, y, z) \in [-1, 1]^3$ with smoothly varying medium

$$c^2(x, y, z) = \frac{1}{2}\left[3 + \sin(16\pi z)\sin(4\pi(x + y))\right],$$

and impose Dirichlet boundary conditions at $z = 1$ and impedance boundary conditions on all other sides. We use a uniform grid $(x_i, y_j, z_k) = (-1 + ih, -1 + jh, -1 + kh)$ with grid spacing $h = 1/n$ where n is the number of gridpoints along a single dimension. The discretization in space and in time is exactly as in the previous example and the problem is forced by $F(x, y, z)$ as in (2.35) with $x_0 = 1/100$, $y_0 = 3/250$, and $z_0 = 4/5$. As in the Marmousi example in the previous section, we parallelize the finite difference solver by a straightforward domain decomposition with the communication handled by MPI. This simulation was carried out on Maneframe II at the Center for Scientific Computation at Southern Methodist University using 64 dual Intel Xeon E5-2695v4 2.1 GHz 18-core Broadwell processors with 45 MB of cache each and 256 GB of DDR4-2400 memory. The magnitude of the solution with $\omega = 200$ and $\omega = 300$ is plotted in Figure 2.15. We use $n = 1000$ for a total of 10^9 gridpoints in the first case, and $n = 1500$ for a total of $3.375 \cdot 10^9$ gridpoints in the second for roughly 15-16 points per wavelength.

2.4 Summary and Future Work

We have presented and analyzed the WaveHoltz iteration, a new iterative method for solving the Helmholtz equation. The iteration results in positive definite and sometimes symmetric matrices that are more amenable for iterative solution by Krylov subspace methods. In choosing a Krylov subspace method we note that CG is the most efficient and memory lean choice when the resulting system is symmetric positive definite, otherwise GMRES generally outperforms other methods such as QMR, LSQR, and TFQMR. As the iteration is based on solving the wave equation it naturally parallelizes and can exploit techniques and spatial discretizations that have been developed for the time dependent problem. Numerical experiments indicate that our iteration appears to converge significantly faster than when the Helmholtz equation is discretized directly and solved iteratively with GMRES.

We believe that the numerical and theoretical results above are promising and note that there are many possible avenues for future exploration. For example we have exclusively used unconditioned Krylov solvers here but the spectral properties of the operator $\mathcal{S}$ indicate that preconditioning should be possible. Further, we have not tried to exploit adaptivity in space or time or any type of sweeping ideas here and we have only briefly touched on the possibilities for more advanced filter design. We hope to study both the numerical and theoretical properties of these in the future.

Finally, here we only analyzed the energy conserving problem. In the following chapter we will analyze problems with energy loss, via either damping or impedance boundary conditions.

Chapter 3

Analysis of an Iterative Solution of the Helmholtz Equation via the Wave Equation for Impedance Boundary Conditions

In this chapter, we continue analyzing time-domain methods for the numerical solution of the Helmholtz equation

$$\nabla \cdot (c^2(x)\nabla u) + \omega^2 u = f(x), \qquad x \in \Omega, \tag{3.1}$$

for a domain Ω, frequency ω, and sound speed $c^2(x)$. The Helmholtz equation (both acoustic and elastic) is useful for seismic, acoustic, and optics applications. The numerical solution of the Helmholtz equation is especially difficult due to the resolution requirements and the indefinite nature of the Helmholtz operator for large frequencies.

In the previous chapter, we introduced a time-domain approach for solving the Helmholtz equation (3.1). Given the Helmholtz solution, $u(x)$, the time-harmonic wave field $\mathrm{Re}\{u(x)e^{-i\omega t}\}$ satisfies the wave equation

$$\begin{aligned} &w_{tt} = \nabla \cdot (c^2(x)\nabla w) - f(x)\cos(\omega t), \quad x \in \Omega, \quad 0 \le t \le T, \\ &w(0,x) = v_0(x), \quad w_t(0,x) = v_1(x), \end{aligned}$$

where $v_0 = \mathrm{Re}\{u(x)\}$ and $v_1 = \omega \mathrm{Im}\{u(x)\}$. In Chapter 2, we introduced an integral operator that time-averaged the wave solution resulting from initial data v_0^n, v_1^n. The time-averaging generates new iterates v_0^{n+1}, v_1^{n+1} leading to a fixed-point iteration we named the WaveHoltz iteration. The convergence of the fixed-point iteration for interior problems with Dirichlet/Neumann boundary conditions (i.e. energy conserving problems) was proven in the continuous and discrete settings.

For such problems, the WaveHoltz iteration can be reformulated as a symmetric and positive-definite system which can be accelerated with Krylov subspace methods such as the conjugate gradient method and GMRES. Numerical experiments using the WaveHoltz iteration indicated promising scaling with frequency for problems with outflow boundary conditions common in seismic applications, though no theoretical proof was given for the convergence of the method in that case.

In this chapter, we extend the analysis of the previous chapter to problems with impedance boundary conditions. In addition, we analyze the WaveHoltz iteration when applied to the damped Helmholtz equation and prove that the iteration convergences for problems with damping or impedance conditions. Numerical results verify that for a sufficiently large damping, the number of iterations for the WaveHoltz iteration to reach convergence for damped Helmholtz equations is independent of frequency. We thus can guarantee convergence of the method to the Helmholtz solution via impedance conditions and/or damping.

We also investigate the effect of choice of timestepper used for the WaveHoltz iteration. In the previous chapter, we noted that in the discrete case the WaveHoltz iteration converged to the solution of a discrete Helmholtz problem with modified frequency. We provided the modification for a centered second-order timestepping scheme which would recover the original discrete Helmholtz solution. Here we consider higher order modified equation (ME) timestepping schemes and show that the fixed-point of the discrete WaveHoltz iteration converges to the discrete Helmholtz solution with the order of the timestepper chosen. We additionally show that, as in the case for EM-WaveHoltz [93], it is possible to *completely remove* time discretization error from the WaveHoltz solution through careful analysis of the discrete iteration and updated quadrature formulas.

The efficient solution of the Helmholtz equation (3.1) via iterative methods is incredibly difficult, especially for high-frequency problems of practical interest, and has been the subject of much research. We refer to the paper [14] for a more in-depth overview of the literature on techniques for solving the Helmholtz equation, as well as the review articles [47, 50, 44]. We focus on the literature that is closely related to the methods and approach used here.

The theoretical justification for working in the time-domain comes from the *limiting amplitude*

principle (see [88, 76, 104]) which states that every solution to the wave equation with a time-harmonic forcing in the exterior of a domain with reflecting boundary conditions tends to the Helmholtz solution.

Rather than evolving a wave equation forward in time to reach a steady state by appealing to the limiting amplitude principle, it is possible to cast the problem as a constrained convex least-squares minimization problem. This approach, originally proposed by Bristeau et al. [29], is the so-called Controllability Method (CM) which seeks to accelerate the convergence to the steady-state limit by minimizing the deviation from time-periodicity of the time-domain solution in second-order form.

In the original CM, along with later work by Heikkola et al. [68, 69], only sound-soft scatterers were considered as the original cost functional of [29] did not generally yield unique minimizers for other types of boundary conditions. An alternative functional, J_∞, proposed by Bardos and Rauch in [18], however, did yield uniqueness of the minimizer at the cost of requiring the storage of the entire history of the computed solution to the wave equation which could be prohibitive for large problems.

For the wave equation in second-order form, the initial condition lies in $H^1 \times L^2$, requiring the solution of a coercive elliptic problem to find a Riesz representative for gradient calculations. Glowinski and Rossi [58] presented an update to the CM by considering the wave equation in first-order form, allowing the initial conditions to lie in a reflexive space and thus removing the need for an elliptic solve each iteration. The discretization chosen in this case, however, had the drawback of requiring inversion of a mass-matrix at each timestep.

In more recent work by Grote and Tang, [63], the use of an alternative functional (or post-processing via a compatibility condition) restored uniqueness of the minimizer of CM. In a follow-up paper, [85], Grote et al. proposed a HDG discretization of the first-order form wave equation which allowed the scheme to be fully explicit and therefore fully parallel. Moreover, they extend CM to general boundary conditions for the first-order formulation and additionally proposed a filtering procedure which allows the original energy functional to be used regardless of the boundary

condition.

The above work has inspired other time-domain methods outside of CM and WaveHoltz. Work by Stolk [101] leverages time-domain approaches as a preconditioner for a GMRES accelerated preconditioner for direct Helmholtz discretizations yielding a hybrid time-frequency domain method. Arnold et al. [15] propose a time-domain method for scattering problems which leverages the compact support of incident field plane wavelets together with a front-tracking adaptive meshing algorithm to reduce the cost of computing a Fourier transform of the wave solution to obtain Helmholtz solutions.

Another important class of methods for solving the Helmholtz equation are the so-called shifted Laplacian preconditioners. The use of the Laplacian as a preconditioner for Helmholtz problems emerged with the initial work of Bayliss et al. [20]. In [20], the normal equations of the discrete Helmholtz equation were iteratively solved using conjugate gradient, with a Symmetric Successive Over-Relaxation (SSOR) sweep of the discrete Laplacian as a preconditioner. Giles and Laird then extended the previous preconditioner to instead solve the Helmholtz system with a flipped sign in front of the Helmholtz term using multigrid [77]. Erlangga, Vuik and Osterlee [46, 43] further generalized the previous work to use a complex-valued shift of the Laplacian leading to the shifted Laplacian preconditioner. For a review of the class of shifted Laplacian preconditioners we refer the reader to the review article by Erlangga [44].

The rest of this chapter is organized as follows. In Section 2 we present analysis for the general WaveHoltz iteration and prove convergence in the case for impedance boundary conditions. In Section 3 we present a brief analysis for the case in which damping is present. Section 4 outlines a discrete analysis of higher order modified equation (ME) schemes, and we additionally present a method to *completely* remove time discretization error from the discrete WaveHoltz solution. Finally, in Section 5 we describe our numerical methods, Section 6 present our numerical examples, and summarize the chapter in Section 7.

3.1 The General Iteration

We consider the Helmholtz equation in a bounded open smooth domain Ω,

$$\nabla \cdot (c^2(x)\nabla u) + \omega^2 u = f(x), \qquad x \in \Omega, \tag{3.2}$$

with boundary conditions of the type

$$i\alpha\omega u + \beta(c(x)\vec{n} \cdot \nabla u) = 0, \quad \alpha^2 + \beta^2 = 1, \qquad x \in \partial\Omega. \tag{3.3}$$

We assume $f \in L^2(\Omega)$ and that $c \in L^\infty(\Omega)$ with the bounds $0 < c_{\min} \leq c(x) \leq c_{\max} < \infty$ a.e. in Ω. Away from resonances, this ensures that there is a unique weak solution $u \in H^1(\Omega)$ to (3.2). Due to the boundary conditions u is in general complex-valued.

We first note that the function $w(t,x) := \mathrm{Re}\{u(x)\exp(-i\omega t)\}$ is a $T = 2\pi/\omega$-periodic (in time) solution to the real-valued forced scalar wave equation

$$\begin{aligned} &w_{tt} = \nabla \cdot (c^2(x)\nabla w) - \mathrm{Re}\{f(x)e^{-i\omega t}\}, \quad x \in \Omega, \quad 0 \leq t \leq T, \\ &w(0,x) = v_0(x), \quad w_t(0,x) = v_1(x), \\ &\alpha w_t + \beta(c(x)\vec{n} \cdot \nabla w) = 0, \quad x \in \partial\Omega, \end{aligned} \tag{3.4}$$

where $v_0 = \mathrm{Re}\{u\}$ and $v_1 = \omega\mathrm{Im}\{u\}$. Based on this observation, our approach is to find this w instead of u. We could thus look for initial data v_0 and v_1 such that w is a T-periodic solution to (3.4). However, there may be several such w, see [63], and we therefore impose the alternative constraint that a certain time-average of w should equal the initial data. More precisely, we introduce the following operator acting on the initial data $v_0 \in H^1(\Omega)$, $v_1 \in L^2(\Omega)$,

$$\Pi \begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right) \begin{bmatrix} w(t,x) \\ w_t(t,x) \end{bmatrix} dt, \quad T = \frac{2\pi}{\omega}, \tag{3.5}$$

where $w(t,x)$ and its time derivative $w_t(t,x)$ satisfies the wave equation (3.4) with initial data v_0 and v_1. The result of $\Pi[v_0,\ v_1]^T$ can thus be seen as a filtering in time of $w(\cdot,x)$ around the ω-frequency. By construction, the solution u of Helmholtz now satisfies the system of equations

$$\begin{bmatrix} \mathrm{Re}\{u\} \\ \omega\mathrm{Im}\{u\} \end{bmatrix} = \Pi \begin{bmatrix} \mathrm{Re}\{u\} \\ \omega\mathrm{Im}\{u\} \end{bmatrix}. \tag{3.6}$$

The WaveHoltz method then amounts to solving this system of equations with the fixed point iteration

$$\boxed{\begin{bmatrix} v \\ v' \end{bmatrix}^{(n+1)} = \Pi \begin{bmatrix} v \\ v' \end{bmatrix}^{(n)}, \qquad \begin{bmatrix} v \\ v' \end{bmatrix}^{(0)} \equiv 0.} \tag{3.7}$$

Provided this iteration converges and the solution to is unique, we obtain the Helmholtz solution as $u = \lim_{n\to\infty} v^n$.

3.1.1 Iteration for the Energy Conserving Case for the General WaveHoltz Iteration

Here we consider boundary conditions of either Dirichlet ($\beta = 0$) or Neumann ($\alpha = 0$) type in (3.4). This is typically the most difficult case for iterative Helmholtz solvers when Ω is bounded. The wave energy is preserved in time and certain ω-frequencies in Helmholtz are resonant, meaning they equal an eigenvalue of the operator $-\nabla\cdot(c^2(x)\nabla)$. Moreover, the limiting amplitude principle does not hold, and one can thus not obtain the Helmholtz solution by solving the wave equation over a long time interval.

By the choice of boundary conditions the operator $-\nabla\cdot(c^2(x)\nabla)$ has a point spectrum with non-negative eigenvalues. Denote those eigenmodes $(\lambda_j^2, \phi_j(x))$. We assume that the angular frequency ω is not a resonance, i.e. $\omega^2 \neq \lambda_j^2$ for all j. The Helmholtz equation (3.1) is then wellposed.

We recall that for any $q \in L^2(\Omega)$ we can expand

$$q(x) = \sum_{j=0}^{\infty} \hat{q}_j \phi_j(x),$$

for some coefficients $\hat{q}_j$ and

$$||q||^2_{L^2(\Omega)} = \sum_{j=0}^{\infty} |\hat{q}_j|^2, \qquad c^2_{\min}||\nabla q||^2_{L^2(\Omega)} \leq \sum_{j=0}^{\infty} \lambda_j^2 |\hat{q}_j|^2 \leq c^2_{\max}||\nabla q||^2_{L^2(\Omega)}.$$

We start by expanding the Helmholtz solution $u = u^R + iu^I$, the initial data v_0, v_1 to the wave

equation (3.4), and the forcing $f = f^R + if^I$ in this way,

$$u^R(x) = \sum_{j=0}^{\infty} \hat{u}_j^R \phi_j(x), \quad v_0(x) = \sum_{j=0}^{\infty} \hat{v}_{0,j} \phi_j(x), \quad v_1(x) = \sum_{j=0}^{\infty} \hat{v}_{1,j} \phi_j(x), \quad f^R(x) = \sum_{j=0}^{\infty} \hat{f}_j^R \phi_j(x),$$

with analogous expansions for the imaginary parts of u and f, u^I and f^I, respectively. Then,

$$-\lambda_j^2 \hat{u}_j^R + \omega^2 \hat{u}_j^R = \hat{f}_j^R \quad \Rightarrow \quad \hat{u}_j^R = \frac{\hat{f}_j^R}{\omega^2 - \lambda_j^2},$$

and similarly for the imaginary parts $\hat{u}_j^I$ and $\hat{f}_j^I$. For the wave equation solution $w(t,x)$ with initial data $w = v_0$ and $w_t = v_1$ we have

$$w(t,x) = \sum_{j=0}^{\infty} \hat{w}_j(t) \phi_j(x),$$

$$\hat{w}_j(t) = \hat{u}_j^R \left[\cos(\omega t) - \cos(\lambda_j t)\right] + \hat{u}_j^I \left[\sin(\omega t) - \frac{\omega}{\lambda_j} \sin(\lambda_j t)\right] + \hat{v}_{0,j} \cos(\lambda_j t) + \frac{\hat{v}_{1,j}}{\lambda_j} \sin(\lambda_j t),$$

with

$$\hat{w}_0^N(t) = \hat{u}_0^R \left[\cos(\omega t) - 1\right] + \hat{u}_0^I \left[\sin(\omega t) - \omega t\right] + \hat{v}_{0,0} + \hat{v}_{1,0} t,$$

if $\lambda_0 = 0$, as is the case for Neumann boundary conditions (a special case which we denote via the superscript N in the following analysis). The filtering step then gives

$$\Pi \begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = \sum_{j=0}^{\infty} \begin{bmatrix} \bar{v}_j \\ \bar{v}_j' \end{bmatrix} \phi_j(x),$$

where

$$\bar{v}_j = \hat{u}_j^R \left(1 - \beta(\lambda_j)\right) - \hat{u}_j^I \frac{\omega}{\lambda_j} \gamma(\lambda_j) + \hat{v}_{0,j} \beta(\lambda_j) + \frac{\hat{v}_{1,j}}{\lambda_j} \gamma(\lambda_j),$$

$$\bar{v}_j' = \hat{u}_j^R \lambda_j \gamma(\lambda_j) + \omega \hat{u}_j^I \left(1 - \beta(\lambda_j)\right) - \hat{v}_{0,j} \lambda_j \gamma(\lambda_j) + \hat{v}_{1,j} \beta(\lambda_j),$$

and

$$\beta(\lambda) := \frac{2}{T} \int_0^T \left(\cos(\omega t) - \frac{1}{4}\right) \cos(\lambda t) dt, \quad \gamma(\lambda) := \frac{2}{T} \int_0^T \left(\cos(\omega t) - \frac{1}{4}\right) \sin(\lambda t) dt.$$

By definition we have

$$\left|\frac{\gamma(\lambda_j)}{\lambda_j}\right| \leq \frac{2}{T} \int_0^T \left|\left(\cos(\omega t) - \frac{1}{4}\right)\right| \left|t \frac{\sin(\lambda t)}{\lambda t}\right| dt \leq \frac{2}{T} \int_0^T \frac{5}{4} t dt = \frac{5\pi}{2\omega}, \tag{3.8}$$

since $|\sin(x)/x| \leq 1$, which ensures the boundedness of the coefficients $\bar{v}_j, \bar{v}'_j$ for small eigenvalues λ_j.

Letting $v_{0,j}, v_{1,j}$ denote the coefficients of v_0, v_1 in the eigenbasis of the Laplacian, we can write the iteration as

$$\begin{bmatrix} v_{0,j}^{n+1} \\ v_{1,j}^{n+1} \end{bmatrix} = \left(\Pi \begin{bmatrix} v_0^n \\ v_1^n \end{bmatrix} \right)_j = (I - B_j) \begin{bmatrix} u_j^R \\ \omega u_j^I \end{bmatrix} + B_j \begin{bmatrix} v_{0,j}^n \\ v_{1,j}^n \end{bmatrix}, \tag{3.9}$$

where if we define $\beta_j = \beta(\lambda_j)$ and $\gamma_j = \gamma(\lambda_j)$ then

$$B_j = \begin{pmatrix} \beta_j & \gamma_j/\lambda_j \\ -\lambda_j \gamma_j & \beta_j \end{pmatrix}, \quad B_0^N = \begin{pmatrix} -1/2 & -\pi/2\omega \\ 0 & -1/2 \end{pmatrix},$$

Moreover, the eigenvectors and eigenvalues of B_j are

$$\xi_j^{\pm} = \begin{pmatrix} \pm i/\lambda \\ 1 \end{pmatrix}, \quad \xi_0^N = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \mu_j = \beta_j \pm i\gamma_j.$$

Introducing the linear operator $\mathcal{S} : L^2(\Omega) \times L^2(\Omega) \to L^2(\Omega) \times L^2(\Omega)$,

$$\mathcal{S} \sum_{j=0}^{\infty} \begin{bmatrix} \hat{u}_j^R \\ \hat{u}_j^I \end{bmatrix} \phi_j(x) = \sum_{j=0}^{\infty} B_j \begin{bmatrix} \hat{u}_j^R \\ \hat{u}_j^I \end{bmatrix} \phi_j(x), \tag{3.10}$$

we may write the iteration as

$$\boxed{\begin{bmatrix} v \\ v' \end{bmatrix}^{(n+1)} = \Pi \begin{bmatrix} v \\ v' \end{bmatrix}^{(n)} = \begin{bmatrix} u^R \\ \omega u^I \end{bmatrix} + \mathcal{S} \left(\begin{bmatrix} v \\ v' \end{bmatrix}^{(n)} - \begin{bmatrix} u^R \\ \omega u^I \end{bmatrix} \right).}$$

We note that, in contrast to the simplified iteration analyzed in [14], the operator $\mathcal{S}$ is not symmetric for the general iteration. Despite this we may identify the eigenmodes of $\mathcal{S}$ from the eigenvectors of B_j via $\xi_j^{\pm}\phi_j$ with eigenvalues $\mu_j = \beta_j \pm i\gamma_j$ and $\xi_0^N = \xi_0^N \phi_0$ with eigenvalue $\mu_0^N = -1/2$.

From (3.9), we see that the iteration for each mode takes the form

$$\begin{bmatrix} v_{0,j}^{n+1} \\ v_{1,j}^{n+1} \end{bmatrix} = \left(\Pi \begin{bmatrix} v_0^n \\ v_1^n \end{bmatrix} \right)_j = \left(I - B_j^n\right) \begin{bmatrix} u_j^R \\ \omega u_j^I \end{bmatrix} + B_j^n \begin{bmatrix} v_{0,j}^0 \\ v_{1,j}^0 \end{bmatrix}$$

so that

$$\begin{bmatrix} v_{0,j}^{n+1} - u_j^R \\ v_{1,j}^{n+1} - \omega u_j^I \end{bmatrix} = B_j^n \begin{bmatrix} v_{0,j}^0 - u_j^R \\ v_{1,j}^0 - \omega u_j^I. \end{bmatrix}. \tag{3.11}$$

We thus require that $B_j^n \to 0$ to ensure convergence of the fixed-point iteration to the solution, $[u^R, \omega u^I]^T$, which is true if and only if the spectral radius of B_j is less than unity uniformly in j. That is, we require that $|\mu_j| < 1$ uniformly in j. Defining the filter function $\mu(\lambda) := \beta(\lambda) + i\gamma(\lambda)$, we may show (with a proof in Appendix .4) the following lemma

Lemma 3.1.1. *The complex-valued filter function μ satisfies $\mu(\omega) = 1$ and*

$$\begin{aligned} 0 \le |\mu(\lambda)| &\le 1 - \frac{15}{32}\left(\frac{\lambda-\omega}{\omega}\right)^2, \quad \text{when } \left|\frac{\lambda-\omega}{\omega}\right| \le \frac{1}{2}, \\ |\mu(\lambda)| &\le \frac{7}{3\pi} \approx 0.74, \qquad\qquad \text{when } \left|\frac{\lambda-\omega}{\omega}\right| \ge \frac{1}{2}, \\ |\mu(\lambda)| &\le b_0 \frac{\omega}{\lambda-\omega}, \qquad\qquad\qquad \text{when } \lambda > \omega, \end{aligned}$$

where $b_0 = 3/2\pi$. Moreover, close to ω we have the local expansion

$$\begin{aligned} |\mu(\omega + r)| &= 1 - b_1\left(\frac{r}{\omega}\right)^2 + R(r/\omega)\left(\frac{r}{\omega}\right)^3, \\ b_1 = \frac{\pi^2}{6} - \frac{1}{4} \approx 1.39, \quad ||R||_\infty &\le \frac{25\pi^4}{4}\left(36 + 20\pi + 250\pi^2 + 75\pi^3\right). \end{aligned} \tag{3.12}$$

We denote

$$\delta_j = \frac{\lambda_j - \omega}{\omega},$$

the relative size of the gap between λ_j and the Helmholtz frequency, and then denote the smallest gap (in magnitude) by δ,

$$\delta = \delta_{j^*}, \quad j^* = \text{argmin}_j |\delta_j|.$$

Then we have the following lemma

Lemma 3.1.2. *Suppose $\delta > 0$. Then, the spectral radius ρ of $\mathcal{S}$ is strictly less than one, and for small δ,*

$$\rho = 1 - b_1\delta^2 + \mathcal{O}(\delta^3), \tag{3.13}$$

with b_1 as in Lemma 3.1.1. Moreover, $\mathcal{S}$ is a bounded linear map from $L^2(\Omega) \times L^2(\Omega)$ to $H^1(\Omega) \times L^2(\Omega)$, and from $H^1(\Omega) \times L^2(\Omega)$ to $H^1(\Omega) \times H^1(\Omega)$.

Proof. From Lemma 3.1.1 we get

$$\rho = \sup_j |\mu(\lambda_j)| \le \sup_j \max\left(1 - \frac{15}{32}\delta_j^2,\ \frac{7}{3\pi}\right) \le \max\left(1 - \frac{15}{32}\delta^2,\ \frac{7}{3\pi}\right) < 1.$$

For the more precise estimate when δ is small we will use (3.12). Since $1 > \rho \ge |\mu(\omega + \omega\delta)| \to 1$ as $\delta \to 0$, we can assume that $\rho > 1 - \eta^2/2$, with $\eta := b_1/2||R||_\infty$, for small enough δ. Then, since $|\mu(\omega + \omega\delta_j)| \le 1 - \eta^2/2$ for $|\delta_j| > \eta$ by Lemma 3.1.1, we have

$$\rho = \sup_{|\delta_j| \le \eta} |\mu(\omega + \omega\delta_j)| = |\mu(\omega + \omega\delta_{k^*})|,$$

for some k^* with $|\delta_{k^*}| \le \eta$. If $\delta_{k^*} = \delta_{j^*}$ (where $\delta = |\delta_{j^*}|$) then (3.12) gives (3.13). If not, we have $\eta \ge |\delta_{k^*}| \ge \delta$ and by Lemma 3.1.1

$$0 \le |\mu(\omega+\omega\delta_{k^*})|-|\mu(\omega+\omega\delta_{j^*})| = -b_1(\delta_{k^*}^2-\delta^2)+R(\delta_{k^*})\delta_{k^*}^3-R(\delta_{j^*})\delta_{j^*}^3 \le -b_1(\delta_{k^*}^2-\delta^2)+\frac{b_1}{2}(\delta_{k^*}^2+\delta^2),$$

which implies that $\delta_k^2 \le 3\delta^2$ and therefore

$$\rho = 1-b_1\delta_{k^*}^2+\mathcal{O}(\delta_{k^*}^3) = 1-b_1\delta^2+b_1(\delta^2-\delta_{k^*}^2)+\mathcal{O}(\delta_{k^*}^3) = 1-b_1\delta^2+\mathcal{O}(\delta_{k^*}^3+\delta^3) = 1-b_1\delta^2+\mathcal{O}(\delta^3).$$

From which (3.13) follows.

Letting $D := \omega\min(1, b_0(1 + 1/|\delta|))$, we note that by Lemma 3.1.1,

$$|\lambda_j\mu(\lambda_j)| \le \omega \le D, \quad \lambda_j \le \omega,$$
$$|\lambda_j\mu(\lambda_j)| \le \omega\frac{b_0\lambda_j}{\lambda_j - \omega} = \omega b_0(1 + 1/\delta_j) \le D, \quad \lambda_j > \omega.$$

Moreover, triangle inequality gives that $|\beta(\lambda_j)|, |\gamma(\lambda_j)| \le |\mu(\lambda_j)|$, which implies both $\lambda_j|\beta(\lambda_j)| \le D$ and $\lambda_j|\gamma(\lambda_j)| \le D$.

Suppose now that g, $h \in L^2(\Omega)$ and

$$g(x) = \sum_{j=0}^{\infty} \hat{g}_j \phi_j(x), \quad h(x) = \sum_{j=0}^{\infty} \hat{h}_j \phi_j(x).$$

Letting $z(x) = [g(x), h(x)]^T$, $\|z\|_{L^2 \times L^2} = 1$, we may split the norm of $\mathcal{S}z$ into

$$||\mathcal{S}z||^2_{H^1(\Omega)\times L^2(\Omega)} = ||\mathcal{S}z||^2_{L^2(\Omega)\times L^2(\Omega)} + ||\nabla \mathcal{S}z||^2_{L^2(\Omega)\times L^2(\Omega)}. \tag{3.14}$$

Letting $C := \max\{D, |\gamma(\lambda_j)|/\lambda_j\}$, which is bounded via the estimate (3.8), straightforward algebra gives the bound

$$\begin{aligned}
||\mathcal{S}z||^2_{L^2(\Omega)\times L^2(\Omega)} &= \sum_{j=0}^{\infty} |\beta(\lambda_j)\hat{g}_j + \frac{\gamma(\lambda_j)}{\lambda_j}\hat{h}_j|^2 + |\lambda_j\gamma(\lambda_j)\hat{g}_j - \beta(\lambda_j)\hat{h}_j|^2 \\
&\leq \sum_{j=0}^{\infty} \left(|\hat{g}_j| + \frac{|\gamma(\lambda_j)|}{\lambda_j}|\hat{h}_j|\right)^2 + \left(\lambda_j|\gamma(\lambda_j)||\hat{g}_j| + |\hat{h}_j|\right)^2 \\
&\leq \sum_{j=0}^{\infty} (1 + C^2)(|\hat{g}_j|^2 + |\hat{h}_j|^2) + 4C|\hat{g}_j||\hat{h}_j| \\
&\leq \sum_{j=0}^{\infty} (1 + C^2)(|\hat{g}_j|^2 + |\hat{h}_j|^2) + 4C(|\hat{g}_j|^2 + |\hat{h}_j|^2) \\
&= \left(1 + C^2 + 4C\right) ||z||^2_{L^2(\Omega)\times L^2(\Omega)},
\end{aligned}$$

since $ab \leq a^2 + b^2$ for $a, b \in [0, 1]$ and z has unit norm. For the second term of (3.14) we find

$$||\nabla \mathcal{S}z||^2_{L^2(\Omega)\times L^2(\Omega)} \leq \underbrace{\sum_{j=0}^{\infty} \frac{\lambda_j^2}{c_{\min}^2}|\beta(\lambda_j)\hat{g}_j + \frac{\gamma(\lambda_j)}{\lambda_j}\hat{h}_j|^2}_{S_1} + \underbrace{\sum_{j=0}^{\infty} \frac{\lambda_j^2}{c_{\min}^2}| - \gamma(\lambda_j)\lambda_j\hat{g}_j + \beta(\lambda_j)\hat{h}_j|^2}_{S_2}.$$

For $g, h \in L^2(\Omega)$, it follows that

$$\begin{aligned}
S_1 = \sum_{j=0}^{\infty} \frac{\lambda_j^2}{c_{\min}^2}|\beta(\lambda_j)\hat{g}_j + \frac{\gamma(\lambda_j)}{\lambda_j}\hat{h}_j|^2 &\leq \sum_{j=0}^{\infty} \frac{D^2}{c_{\min}^2}\left(|\hat{g}_j|^2 + |\hat{h}_j|^2\right) + \frac{2\lambda}{c_{\min}^2}|\beta(\lambda_j)||\gamma(\lambda_j)||\hat{g}_j||\hat{h}_j| \\
&\leq \sum_{j=0}^{\infty} \frac{D^2}{c_{\min}^2}\left(|\hat{g}_j|^2 + |\hat{h}_j|^2\right) + \frac{2D}{c_{\min}^2}\left(|\hat{g}_j|^2 + |\hat{h}_j|^2\right) \\
&= \frac{D^2 + 2D}{c_{\min}^2}\|z\|_{L^2(\Omega)\times L^2(\Omega)},
\end{aligned}$$

which gives

$$\|\mathcal{S}\|^2_{H^1(\Omega)\times L^2(\Omega)} = \sup_{\|z\|=1} ||\mathcal{S}z||^2_{H^1(\Omega)\times L^2(\Omega)} \leq \left(1 + C^2 + 4C + \frac{D^2 + 2D}{c_{\min}^2}\right),$$

showing that $\mathcal{S}$ is a bounded linear map from $L^2(\Omega) \times L^2(\Omega)$ to $H^1(\Omega) \times L^2(\Omega)$.

If instead $g \in H^1(\Omega)$ and $h \in L^2(\Omega)$, then

$$\begin{aligned} S_2 &= \sum_{j=0}^{\infty} \frac{\lambda_j^2}{c_{\min}^2} | - \gamma(\lambda_j)\lambda_j \hat{g}_j + \beta(\lambda_j)\hat{h}_j|^2 \\ &\leq \sum_{j=0}^{\infty} \frac{1}{c_{\min}^2} \left(\lambda_j^4 \gamma^2(\lambda_j)|\hat{g}_j|^2 + 2\lambda_j^3 |\beta(\lambda_j)||\gamma(\lambda_j)||\hat{g}_j||\hat{h}_j| + \lambda_j^2 \beta^2(\lambda_j)|\hat{h}_j|^2 \right) \\ &\leq \sum_{j=0}^{\infty} \frac{D^2}{c_{\min}^2} \left(\lambda_j^2 |\hat{g}_j|^2 + 2\lambda_j |\hat{g}_j||\hat{h}_j| + |\hat{h}_j|^2 \right). \end{aligned}$$

We note that Hölder's inequality gives

$$\sum_{j=0}^{\infty} \lambda_j |\hat{g}_j||\hat{h}_j| \leq c_{\max} \|\nabla g\|_{L^2(\Omega)} \|h\|_{L^2(\Omega)},$$

so that

$$S_2 \leq \frac{D^2}{c_{\min}^2} \left(c_{\max}^2 \|\nabla g\|_{L^2(\Omega)}^2 + 2c_{\max} \|\nabla g\|_{L^2(\Omega)} \|h\|_{L^2(\Omega)} + \|h\|_{L^2(\Omega)}^2 \right),$$

and thus

$$\|\mathcal{S}\|_{H^1(\Omega)\times H^1(\Omega)}^2 \leq \|\mathcal{S}\|_{H^1(\Omega)\times L^2(\Omega)}^2 + \sup_{\|z\|=1} S_2 \leq \|\mathcal{S}\|_{H^1(\Omega)\times L^2(\Omega)}^2 + \frac{D^2}{c_{\min}^2}(c_{\max}^2 + 2c_{\max} + 1),$$

which shows that $\mathcal{S}$ is a bounded linear map from $H^1(\Omega) \times L^2(\Omega)$ to $H^1(\Omega) \times H^1(\Omega)$, proving the lemma. □

Further, denoting $e^n := [\mathrm{Re}\{u\} - v_0^n, \omega \mathrm{Im}\{u\} - v_1^n]^T = [e_0^n, e_1^n]^T$, from (3.11) we obtain

$$e^n = \mathcal{S}[\mathrm{Re}\{u\} - v_0^{n-1}, \omega \mathrm{Im}\{u\} - v_1^{n-1}]^T = \mathcal{S}^n[\mathrm{Re}\{u\} - v_0^0, \omega \mathrm{Im}\{u\} - v_1^0]^T = \mathcal{S}^n e^0,$$

which shows that $e^n \to 0$ since $S^n \to 0$. Thus the iterates $[v_0^n, v_1^n]^T$ converge to $[\mathrm{Re}\{u\}, \omega \mathrm{Im}\{u\}]^T$ in $L^2(\Omega) \times L^2(\Omega)$. Since $v_0^0 = v_1^0 = 0$, by Lemma 3.1.2 it follows that the iterates $[v_0^n, v_1^n]^T \in H^1(\Omega) \times L^2(\Omega)$ for $n > 0$. Additionally we have that the iterates $[v_0^n, v_1^n]^T \in H^1(\Omega) \times H^1(\Omega)$ for $n > 1$. We can therefore also get convergence in $H^1(\Omega) \times H^1(\Omega)$. To show this, let

$$\beta_j + i\gamma_j = r_j \exp(i\phi_j), \quad r_j^2 = |\beta_j|^2 + |\gamma_j|^2, \quad \phi_j = \arctan(\gamma_j/\beta_j).$$

It can then be shown that powers of the operator B_j can be written as

$$B_j^n = r_j^n \begin{pmatrix} \cos(n\phi_j) & \sin(n\phi_j)/\lambda_j \\ -\lambda_j \sin(n\phi_j) & \cos(n\phi_j) \end{pmatrix},$$

where each entry is bounded and goes to zero in the limit as $n \to \infty$ since the spectral radius of B_j is less than one. From the Hölder inequality it follows that

$$\sum_{j=0}^{\infty} \lambda_j |\hat{e}_{j,0}^0||\hat{e}_{j,1}^1| \leq c_{\max} \|\nabla e_0^0\|_{L^2(\Omega)} \|e_1^0\|_{L^2(\Omega)},$$

so that

$$\begin{aligned}
||\nabla \mathcal{S}^n e^0||^2_{L^2(\Omega)\times L^2(\Omega)} &\leq \sum_{j=0}^{\infty} \frac{\lambda_j^2 |r_j|^{2n}}{c_{\min}^2} \left(\left| \cos(n\phi_j)\hat{e}_{j,0}^0 + \frac{\sin(n\phi_j)}{\lambda_j} \hat{e}_{j,1}^1 \right|^2 + \left| -\lambda_j \sin(n\phi_j)\hat{e}_{j,0}^0 + \cos(n\phi_j)\hat{e}_{j,1}^1 \right|^2 \right) \\
&\leq \sum_{j=0}^{\infty} \frac{|\mu_j|^{2n}}{c_{\min}^2} \left(\lambda_j^2 |\hat{e}_{j,0}^0|^2 + \lambda_j |\hat{e}_{j,0}^0||\hat{e}_{j,1}^1| + |\hat{e}_{j,1}^1|^2 \right) + \frac{D^2 |\mu_j|^{2n-2}}{c_{\min}^2} \left(\lambda_j^2 |\hat{e}_{j,0}^0|^2 + \lambda_j |\hat{e}_{j,0}^0||\hat{e}_{j,1}^1| + |\hat{e}_{j,1}^1|^2 \right) \\
&\leq \left(\sup_j |\mu_j| \right)^{2n-2} \frac{1+D^2}{c_{\min}^2} \sum_{j=0}^{\infty} \left(\lambda_j^2 |\hat{e}_{j,0}^0|^2 + \lambda_j |\hat{e}_{j,0}^0||\hat{e}_{j,1}^1| + |\hat{e}_{j,1}^1|^2 \right) \\
&\leq \rho^{2n-2} \frac{1+D^2}{c_{\min}^2} \left(c_{\max}^2 \|\nabla e_0^0\|^2_{L^2(\Omega)} + c_{\max} \|\nabla e_0^0\|_{L^2(\Omega)} \|e_1^0\|_{L^2(\Omega)} + \|e_1^0\|^2_{L^2(\Omega)} \right) \to 0.
\end{aligned}$$

We conclude that the iteration converges in $H^1(\Omega) \times H^1(\Omega)$ with convergence rate ρ. By Lemma 3.1.2 we have $\rho \sim 1 - 1.39\delta^2$ so that the smallest gap, δ, determines the convergence rate. We thus have proven the following theorem

Theorem 3.1.3. *The iteration in* (3.7) *and* (3.5) *converges in* $H^1(\Omega) \times H^1(\Omega)$ *for the Dirichlet and Neumann problems away from resonances to the solution of the Helmholtz equation (3.1). The convergence rate is* $1 - \mathcal{O}(\delta^2)$*, where* δ *is the minimum gap between* ω *and the eigenvalues of* $-\nabla \cdot (c^2(x)\nabla)$.

As seen in [14], we may reformulate the iteration as the linear system

$$(I - \mathcal{S})\, v =: \mathcal{A} v = b := \Pi 0,$$

which allows the convergence to be accelerated by a Krylov method. We note that the evaluation of $\mathcal{A}$ can be done by evolving the wave equation for one period in time with initial data v without the need to explicitly form the matrix $\mathcal{A}$.

Remark 3.1.1. *The operator $\mathcal{A}$ for the general iteration is not symmetric unlike the simplified iteration for energy-conserving problems where $v_1 = 0$. For interior, energy-conserving problems we recommend the use of the simplified iteration so that the conjugate gradient method may be used to accelerate convergence. For other boundary conditions, the general WaveHoltz iteration is required and a more versatile Krylov method, such as GMRES, should be used.*

3.1.2 Convergence in the Non-Energy Conserving Case

With Theorem 3.1.3 providing convergence of the general WaveHoltz iteration in the energy-conserving case, we turn toward proving convergence for problems with impedance boundary conditions. For simplicity we prove convergence in a single spatial dimension. This is not restrictive, as it is possible for the following technique to be extended to higher dimensions for particular problems. Consider now the following Helmholtz problem with impedance boundary conditions

$$\begin{aligned} \left[c^2(x)u'(x)\right]' + \omega^2 u(x) &= f(x), \quad a \le x \le b, \\ i\alpha\omega u(a) - \beta c(x) u_x(a) &= 0, \\ i\alpha\omega u(b) + \beta c(x) u_x(b) &= 0, \end{aligned} \tag{3.15}$$

where α, $\beta \neq 0$. Here we assume $1/c \in L^1_{\text{loc}}([a,b])$, $f \in L^2([a,b])$ is compactly supported away from the boundary, and that $c(a) = c_a$, $c(b) = c_b$ where c is constant in a neighborhood of the endpoints. We reformulate this in the time domain as

$$\begin{aligned} &w_{tt} = \frac{\partial}{\partial x}\left[c^2(x)\frac{\partial}{\partial x} w\right] - f(x)e^{-i\omega t}, \quad a \le x \le b, \quad 0 \le t \le T, \\ &w(0,x) = v_0(x), \quad w_t(0,x) = v_1(x), \\ &\alpha w_t(t,a) - \beta c(x) w_x(t,a) = 0, \\ &\alpha w_t(t,b) + \beta c(x) w_x(t,b) = 0. \end{aligned}$$

In general, the solution of the above equation will yield complex-valued solutions and so we take the real part of the equation as shown earlier and use the general iteration (3.7). Note that in 1D the impedance boundary conditions with $\alpha = \beta = 1/\sqrt{2}$ are equivalent to outflow/radiation conditions

when the initial data is compactly supported in the interval $[a,b]$. If $\alpha \neq \beta$ with α, $\beta \neq 0$, then in addition to outgoing waves at the boundary there will be reflections due to the impedance boundary condition. In either case, if we let $\tilde{a} < a - c_a T/2$ and $\tilde{b} > b + c_b T/2$, then w is equal to $\tilde{w}$ on $[a,b]$ for $t \in [0,T]$ if $\tilde{w}$ solves the following Neumann problem (with an outline of the construction in Appendix .5)

$$\begin{aligned} &\tilde{w}_{tt} = \frac{\partial}{\partial x}\left[\tilde{c}^2(x)\frac{\partial}{\partial x}\tilde{w}\right] - \mathrm{Re}\{\tilde{f}(x)e^{-i\omega t}\}, \quad \tilde{a} \leq x \leq \tilde{b}, \quad 0 \leq t \leq T, \\ &\tilde{w}(0,x) = \tilde{v}_0(x), \quad \tilde{w}_t(0,x) = \tilde{v}_1(x), \\ &\tilde{w}_x(t,a) = 0, \quad \tilde{w}_x(t,b) = 0, \end{aligned} \tag{3.16}$$

where $\tilde{v}_0$ and $\tilde{c}$ are the constant extensions (with $\gamma = \alpha/\beta$)

$$\tilde{v}_0(x) = \begin{cases} v_0(a_0), & a \leq x < a_0, \\ v_0(x), & a_0 \leq x \leq b_0, \\ v_0(b_0), & b_0 < x \leq b, \end{cases} \quad \tilde{c}(x) = \begin{cases} \gamma c_a, & a \leq x < a_0, \\ c(x), & a_0 \leq x \leq b_0, \\ \gamma c_b, & b_0 < x \leq b, \end{cases}$$

and $\tilde{v}_1, \tilde{f}$ are zero extensions of v_1 and f, respectively.

That is, we extend the domain such that traveling waves may reflect off of the Neumann boundary but not re-enter the domain of interest, $a \leq x \leq b$, within a period T. Let Π be the WaveHoltz integral operator (3.5) on the original domain Ω with impedance boundary conditions. We recall that iterates generated by Π at a given point, $x \in \Omega$, are the time-average of the wave solution at x generated by the input data. Since the extended wave solution $\tilde{w}(t,x) = w(t,x)$ for $0 \leq t \leq T$, we may write $\Pi = P\widetilde{\Pi}E$ where P is a projection operator onto the initial interval, i.e. $Pv(x) = v(x)|_{a \leq x \leq b}$, E is the extension operator such that $[v_0, v_1]^T \to [\tilde{v}_0, \tilde{v}_1]^T$, and $\widetilde{\Pi}$ is the WaveHoltz operator on the domain $\widetilde{\Omega}$. If it can be guaranteed that $\omega^2 \neq \lambda_j^2$ where λ_j^2 is an eigenvalue of the operator $-\partial_x(\tilde{c}^2(x)\partial_x)$, then we may prove convergence as was done for Theorem 3.1.3.

To show this, results on the continuity of eigenvalues of the Laplacian from [74] will be used. We present the framework of [74] needed here and consider the following differential equation

$$-(c^2 y')' = \lambda y, \quad x \in (a', b'), \quad -\infty \leq a' < b' \leq \infty, \quad \lambda \in \mathbb{R}, \tag{3.17}$$

where $c^2 : (a', b') \to \mathbb{R}$ and $1/c^2 \in L^1_{\text{loc}}(a', b')$. Leting $I = [a, b], \quad a' < a < b < b'$ and additionally imposing the Neumann conditions $y'(a) = 0 = y'(b)$, the above Sturm-Liouville (SL) problem is such that all eigenvalues are real, simple, and can be ordered to satisfy

$$0 \leq \lambda_0^2 < \lambda_1^2 < \lambda_2^2 < \ldots; \quad \lim_{n\to\infty} \lambda_n^2 = +\infty. \tag{3.18}$$

We thus immediately have that the eigenvalues of the Laplacian are countable.

Under the above assumptions, we state the following theorem that is proven in [74].

Theorem 3.1.4 (Kong & Zettle)**.** *Let $1/c^2 \in L^1_{loc}(a', b')$, fix a', b', and suppose b is such that $a' < b < b'$. Let $\lambda_n(b)$ be an eigenvalue of the SL problem* (3.17) *with homogeneous Neumann boundary conditions with corresponding eigenfunction $u_n(x; b)$. Then the eigenvalue $\lambda(b) \in C^1([a', b'])$ satisfies the following differential equation:*

$$\lambda_n'(b) = -\lambda_n(b) u^2(b; b).$$

That is, the eigenvalues of the SL problem (3.17) are differentiable functions of the endpoint b. This gives us the following useful corollary.

Corollary 3.1.4.1. *For $n = 1, 2, \cdots$, $\lambda_n(b)$ is a strictly decreasing function of b on $[a', b']$.*

Proof. For homogeneous Neumann conditions, we have that $u_n'(b; b) = 0$. It follows that $u_n(b; b) \neq 0$ as otherwise $u_n(b; b) \equiv 0$ since u_n satisfies a linear, homogeneous second order ODE. As $\lambda_n(b) > 0$ for $n > 0$ we then have

$$\lambda_n'(b) = -\lambda_n(b) u_n^2(b, b) < 0,$$

so that $\lambda_n(b)$ is a strictly decreasing function of the endpoint b. □

As a consequence of Theorem 3.1.4, we have

Lemma 3.1.5. *Suppose $\tilde{c} \geq 0$ a.e. and $\omega > 0$. Fix a, and consider the Neumann eigenvalues $\lambda_n(b)$ for $b \in (b_0 + c_b T/2, r)$ where $r > b_0 + c_b T/2$. Then there exists an endpoint $\tilde{b} \in (b_0 + c_b T/2, r)$ such that $\omega^2 \neq \lambda_n(\tilde{b})$ for each $n \in \mathbb{N}_0$.*

Proof. Clearly we have $\lambda_0(b) = 0$ for every b, and since $\omega > 0$ we have $\omega^2 \neq \lambda_0(b)$. Suppose now that b is such that $\omega^2 = \lambda_n(b)$ for some $n \in \mathbb{N}$. Recall that by (3.18) we have that $\omega^2 = \lambda_n(b) < \lambda_{n+1}(b)$. Since $\lambda_n(b), \lambda_{n+1}(b)$ are continuous, decreasing functions of the endpoint by Corollary 3.1.4.1, there necessarily exists $\delta > 0$ such that

$$\lambda_n(b+\delta) < \omega^2 < \lambda_{n+1}(b+\delta),$$

Letting $\tilde{b} = b + \delta$ we thus have that $\omega^2 \neq \lambda_n(\tilde{b})$ for each $n \in \mathbb{N}_0$, as desired. □

From this we can prove the following theorem, in which we note we demonstrate convergence in $H^1(\Omega) \times L^2(\Omega)$ rather than $H^1(\Omega) \times H^1(\Omega)$.

Theorem 3.1.6. *Let the 1D domain $\Omega = [a, b]$ be a bounded interval. Suppose $f \in L^2(\Omega)$ is compactly supported in Ω away from the boundary, $1/c^2 \in L^1_{loc}(\Omega)$, and $c(a) = c_a$, $c(b) = c_b$, with c constant near the endpoints. Under these conditions, the iteration* (3.7) *and* (3.5) *converges in $H^1(\Omega) \times L^2(\Omega)$ for the Helmholtz problem with impedance boundary conditions to the solution of the Helmholtz equation* (3.15).

Proof. By Lemma 3.1.5, there exists an extended wave equation (3.16) on the domain $\tilde{\Omega} = [\tilde{a}, \tilde{b}]$ with homogeneous Neumann boundary conditions such that the eigenvalues $\tilde{\lambda}_j$ of the Laplacian, $-\partial_x(\tilde{c}^2 w_x)$, on $\tilde{\Omega}$ are not in resonance. Defining $\tilde{\beta}_j = \beta(\tilde{\lambda}_j)$, $\tilde{\gamma}_j = \gamma(\tilde{\lambda}_j)$, and $\tilde{\mu}_j = \mu(\tilde{\lambda}_j)$, this immediately gives that the spectral radius of the WaveHoltz operator, $\tilde{\rho} = \sup_j |\tilde{\mu}_j|$, is smaller than one. Moreover, the extended wave solution $\tilde{w}$ on $\tilde{\Omega}$ coincides with the interior impedance wave solution w on Ω for $t \in [0, T]$.

Letting u be the solution of the Helmholtz equation (3.15), we define $q(t, x) = \cos(\omega t)[\mathrm{Re}\{u\}, \omega \mathrm{Im}\{u\}]^T$ the time-harmonic Helmholtz solution in Ω and $\tilde{w}^n(t, x)$ the solution of (3.16) with initial data v_0^n, v_1^n. Letting the error be $e^n := [\mathrm{Re}\{u\} - v_0^n, \omega \mathrm{Im}\{u\} - v_1^n]^T = [e_0^n, e_1^n]^T$, it is clear that the

difference $d(t,x) = q(t,x) - w(t,x)$ satisfies the unforced, homogeneous wave equation

$$\begin{aligned} &d_{tt} = \frac{\partial}{\partial x}\left[c^2(x)\frac{\partial}{\partial x}d\right], \quad a \le x \le b, \quad 0 \le t \le T, \\ &d(0,x) = e_0(x), \quad d_t(0,x) = e_1(x), \\ &\alpha d_t(t,a) - \beta c(x) d_x(t,a) = 0, \\ &\alpha d_t(t,b) + \beta c(x) d_x(t,b) = 0. \end{aligned}$$

It follows that the WaveHoltz iteration applied to the error is of the form

$$\begin{bmatrix} e_0^{n+1} \\ e_1^{n+1} \end{bmatrix} = \Pi \begin{bmatrix} e_0^{n} \\ e_1^{n} \end{bmatrix} = P\widetilde{\mathcal{S}}E \begin{bmatrix} e_0^{n} \\ e_1^{n} \end{bmatrix} = (P\widetilde{\mathcal{S}}E)^{n+1} \begin{bmatrix} e_0^{0} \\ e_1^{0} \end{bmatrix}, \tag{3.19}$$

where $\widetilde{\mathcal{S}}$ is the representation of the operator $\widetilde{\Pi}$, as defined in (3.10), with respect to the eigenbasis of the extended Laplacian. We may rearrange the above iteration as

$$(P\widetilde{\mathcal{S}}E)^{n+1}e^0 = P(\widetilde{\mathcal{S}}EP)^n\widetilde{\mathcal{S}}Ee^0 = P(\widetilde{\mathcal{S}}EP)^n\tilde{e}^0,$$

where $\tilde{e}^0 = \widetilde{\mathcal{S}}Ee^0 \in H^1(\tilde{\Omega}) \times L^2(\tilde{\Omega})$ since $e^0 \in L^2(\Omega) \times L^2(\Omega)$ and $\widetilde{\mathcal{S}}$ is a bounded linear map from $L^2(\tilde{\Omega}) \times L^2(\tilde{\Omega})$ to $H^1(\tilde{\Omega}) \times L^2(\tilde{\Omega})$ by Lemma 3.1.2. We then have that $(\widetilde{\mathcal{S}}EP)^n : H^1(\tilde{\Omega}) \times L^2(\tilde{\Omega}) \to H^1(\tilde{\Omega}) \times L^2(\tilde{\Omega})$ and we may obtain convergence if $\|(\widetilde{\mathcal{S}}EP)^n\tilde{e}^0\|_c$ goes to zero.

Let $\tilde{z} = [\tilde{v}_0, \tilde{v}_1]^T$ with $\tilde{z} \in H^1(\tilde{\Omega}) \times L^2(\tilde{\Omega})$. We define the energy semi-norm $\|\cdot\|_c$ on $H^1(\tilde{\Omega})$,

$$\|\tilde{v}_0\|_c^2 = \left\|\tilde{c}\frac{\partial}{\partial x}\tilde{v}_0\right\|_{L^2(\tilde{\Omega})}^2 = \sum_{j=0}^{\infty} \tilde{\lambda}_j^2 |\tilde{v}_{0,j}|^2,$$

with the associated semi-norm on $H^1(\tilde{\Omega}) \times L^2(\tilde{\Omega})$

$$\|\tilde{z}\|_c^2 = \|\tilde{v}_0\|_c^2 + \|\tilde{v}_1\|_{L^2(\tilde{\Omega})}^2 = \left\|\tilde{c}\frac{\partial}{\partial x}\tilde{v}_0\right\|_{L^2(\tilde{\Omega})}^2 + \|\tilde{v}_1\|_{L^2(\tilde{\Omega})}^2.$$

Note that in this semi-norm we have that

$$\begin{aligned} \|EP\tilde{z}\|_c^2 = \int_{\tilde{\Omega}} \left|\tilde{c}\frac{\partial}{\partial x}EP\tilde{v}_0\right|^2 + |EP\tilde{v}_1|^2\,dx = \int_{\Omega} \left|\tilde{c}\frac{\partial}{\partial x}\tilde{v}_0\right|^2 + |\tilde{v}_1|^2\,dx \le \int_{\tilde{\Omega}} \left|\tilde{c}\frac{\partial}{\partial x}\tilde{v}_0\right|^2 + |\tilde{v}_1|^2\,dx \\ \le \|\tilde{z}\|_c^2. \end{aligned}$$

We define $\tilde{y} = EP\tilde{z}$, where $\tilde{y}$ has the form

$$\tilde{y} = EP\sum_{j=0}^{\infty}\begin{bmatrix}\tilde{v}_{0,j}\\ \\ \tilde{v}_{1,j}\end{bmatrix}\phi_j = \sum_{j=0}^{\infty}\begin{bmatrix}\tilde{y}_{0,j}\\ \\ \tilde{y}_{1,j}\end{bmatrix}\phi_j.$$

It follows that

$$\widetilde{\mathcal{S}}EP\tilde{z} = \sum_{j=0}^{\infty} B_j\begin{bmatrix}\tilde{y}_{0,j}\\ \\ \tilde{y}_{1,j}\end{bmatrix}\phi_j = \sum_{j=0}^{\infty}\begin{bmatrix}\tilde{\beta}_j\tilde{y}_{0,j} + \tilde{\gamma}_j\tilde{y}_{1,j}/\tilde{\lambda}_j\\ \\ -\tilde{\lambda}_j\tilde{y}_{0,j} + \tilde{\beta}_j\tilde{y}_{1,j}\end{bmatrix}\phi_j.$$

so that

$$\|\widetilde{\mathcal{S}}EP\tilde{z}\|_c^2 = \sum_{j=0}^{\infty}\tilde{\lambda}_j^2\left(\tilde{\beta}_j\tilde{y}_{0,j} + \frac{\tilde{\gamma}_j}{\tilde{\lambda}_j}\tilde{y}_{1,j}\right)^2 + \sum_{j=0}^{\infty}\left(-\tilde{\lambda}_j\tilde{\gamma}_j\tilde{y}_{0,j} + \tilde{\beta}_j\tilde{y}_{1,j}\right)^2.$$

Since $\tilde{\beta}_j^2 + \tilde{\gamma}_j^2 = |\tilde{\mu}_j|^2 \leq \tilde{\rho}^2 < 1$, a simple expansion shows that

$$\|\widetilde{\mathcal{S}}EP\tilde{z}\|_c^2 = \sum_{j=0}^{\infty}(\tilde{\beta}_j^2 + \tilde{\gamma}_j^2)(\tilde{\lambda}_j^2|\tilde{y}_{0,j}|^2 + |\tilde{y}_{1,j}|^2) \leq \left(\sup_j |\tilde{\mu}_j|^2\right)\sum_{j=0}^{\infty}\tilde{\lambda}_j^2|\tilde{y}_{0,j}|^2 + |\tilde{y}_{1,j}|^2 \leq \tilde{\rho}^2\|\tilde{z}\|_c^2.$$

It follows that

$$\|(\widetilde{\mathcal{S}}EP)^n\tilde{e}^0\|_c^2 \leq \tilde{\rho}^2\|(\widetilde{\mathcal{S}}EP)^{n-1}\tilde{e}^0\|_c^2 \leq \cdots \leq \tilde{\rho}^{2n}\|\tilde{e}^0\|_c^2 \to 0,$$

so that $\|\nabla e_0^n\|_{L^2(\Omega)}, \|e_1^n\|_{L^2(\Omega)} \to 0$.

With $\tilde{e}_0^n = (\widetilde{\mathcal{S}}EP)^n\tilde{e}^0$, an application of the triangle and Poincaré inequality now gives

$$\begin{aligned}\|\tilde{e}_0^n\|_{H^1(\tilde{\Omega})}^2 = \|\nabla\tilde{e}_0^n\|_{L^2(\tilde{\Omega})}^2 + \|\tilde{e}_0^n\|_{L^2(\tilde{\Omega})}^2 &\leq \|\nabla\tilde{e}_0^n\|_{L^2(\tilde{\Omega})}^2 + \|\tilde{e}_0^n - \frac{1}{2}\tilde{e}_{0,0}^n\phi_0\|_{L^2(\tilde{\Omega})}^2 + \frac{1}{2}\|\tilde{e}_{0,0}^n\phi_0\|_{L^2(\tilde{\Omega})}^2\\ &\leq \|\nabla\tilde{e}_0^n\|_{L^2(\tilde{\Omega})}^2 + C\|\nabla\tilde{e}_0^n\|_{L^2(\tilde{\Omega})}^2 + \frac{1}{2}\|\tilde{e}_{0,0}^n\phi_0\|_{L^2(\tilde{\Omega})}^2\\ &\leq (C+1)\|\nabla\tilde{e}_0^n\|_{L^2(\tilde{\Omega})}^2 + \frac{1}{2}\|\tilde{e}_{0,0}^n\phi_0\|_{L^2(\tilde{\Omega})}^2, \qquad (3.20)\end{aligned}$$

where $\phi_0 \in \widetilde{\Omega}$ is a constant eigenfunction of the Laplacian (and thus of $\widetilde{\mathcal{S}}$) with eigenvalue $\lambda_0 = 0$. It follows that to obtain convergence in $H^1(\tilde{\Omega})$ of the error $\tilde{e}_0^n$ we must examine the convergence of $\tilde{e}_{0,0}$ separately.

It is clear that any constant function on $\widetilde{\Omega}$ is an eigenfunction of the operator EP. With $\tilde{e}_0 = \widetilde{\mathcal{S}}Ee_0$ we have

$$\widetilde{\mathcal{S}}EP\begin{bmatrix}\tilde{e}_{0,0}\\ \\ 0\end{bmatrix}\phi_0 = \widetilde{\mathcal{S}}\begin{bmatrix}\tilde{e}_{0,0}\\ \\ 0\end{bmatrix}\phi_0 = B_0^N\begin{bmatrix}\tilde{e}_{0,0}\\ \\ 0\end{bmatrix}\phi_0 = -\frac{1}{2}\begin{bmatrix}\tilde{e}_{0,0}\\ \\ 0\end{bmatrix}\phi_0,$$

so that with $\tilde{z}_0 = [\tilde{e}_{0,0}, 0]^T \phi_0$ then

$$\left\| (\widetilde{\mathcal{S}}EP)^n \tilde{z}_0 \right\|_{L^2(\tilde{\Omega})\times L^2(\tilde{\Omega})} \leq 2^{-1} \left\| (\widetilde{\mathcal{S}}EP)^{n-1} \tilde{z}_0 \right\|_{L^2(\tilde{\Omega})\times L^2(\tilde{\Omega})} \leq \cdots \leq 2^{-n} \left\| \tilde{z}_0 \right\|_{L^2(\tilde{\Omega})\times L^2(\tilde{\Omega})} \to 0.$$

It then follows that

$$\lim_{n\to\infty} (P\widetilde{\mathcal{S}}E)^{n+1} e^0 = \lim_{n\to\infty} P(\widetilde{\mathcal{S}}EP)^n \tilde{e}^0 = 0 \implies \lim_{n\to\infty} \frac{1}{2} \|\tilde{e}^n_{0,0}\phi_0\|^2_{L^2(\tilde{\Omega})} = 0.$$

Thus taking the limit of (3.20) gives $\|\tilde{e}^n_0\|^2_{H^1(\tilde{\Omega})} \to 0$, so that we obtain convergence of the iteration in $H^1(\Omega) \times L^2(\Omega)$. □

Remark 3.1.2. *The above analysis is for a single spatial dimension, but we note that it in certain situations it may be extended to higher dimensions. For instance, interior impedance problems with constant coefficients may be extended by an appropriate enclosing box from which the above arguments can give convergence. In general, it is difficult to prove convergence in higher dimensions as care needs to be taken to make appropriate wavespeed extensions that avoid reflections due to potentially discontinuous wavespeeds close to boundaries with impedance conditions.*

3.2 Damped Wave/Helmholtz Equation

As mentioned in the introduction, a popular preconditioning approach for solving Helmholtz problems is to introduce a damping term as in the shifted Laplacian preconditioners [44]. In this section we similarly consider the complex-valued damped wave equation

$$w_{tt} + \eta w_t = \nabla \cdot \left[c^2(x) \nabla w \right] - f(x) e^{-i\omega t},$$

from which we note that if $w(t, x) = u(x)e^{-i\omega t}$ then

$$\nabla \cdot \left[c^2(x) \nabla u \right] + \left(\omega^2 + i\eta\omega \right) u = f(x),$$

so that we essentially have added a purely imaginary shift of the Laplacian

$$\mathcal{L} = -\nabla \cdot \left[c^2(x) \nabla \right] - i\eta\omega.$$

While for the sake of simplicity we consider the complex-valued problem in this section, in practice we solve the real-valued problem as presented in Section 3.1 with the filter (3.5). For the above complex-valued problem, we may then similarly prove an analogous result to Theorem 3.1.3

Theorem 3.2.1. *The iteration* (3.7) *with the complex-valued filter*

$$\Pi \begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = \frac{1}{T}\int_0^T e^{i\omega t} \begin{bmatrix} w(t,x) \\ w_t(t,x) \end{bmatrix} dt, \quad T = \frac{2\pi}{\omega},$$

converges for every $\eta > 0$ *with a convergence rate bounded by* $2(1-e^{-\eta T/2})/\eta T$.

Proof. Suppose (λ_j^2, ϕ_j) are the eigenmodes of the real-valued Laplacian in the domain Ω. We note that the shifted Laplacian now has a spectrum that is $\lambda_j^2 - i\eta\omega$. Expanding in terms of this basis and taking inner products, we can see that

$$(\omega^2 + i\eta\omega - \lambda_j^2)\hat{u}_j = \hat{f}_j,$$

where we expand the real and imaginary parts of u and f as $\hat{u}_j = u_j^R + iu_j^I$ and $\hat{f}_j = f_j^R + if_j^I$. Let the damped wave equation solution have the form

$$\sum_{n=0}^{\infty} w_j(t)\phi_j(x).$$

Defining $\alpha_j = \sqrt{4\lambda_j^2 - \eta^2}/2$, then the solution can be shown to be given by

$$w_j(t) = \hat{u}_j\left(e^{-i\omega t} - e^{-\frac{\eta t}{2}}\left[\cos(\alpha_j t) + \frac{\eta - 2i\omega}{2\alpha_j}\sin(\alpha_j t)\right]\right) + \hat{v}_{0,j}e^{-\frac{\eta t}{2}}\left[\cos(\alpha_j t) + \frac{\eta}{2\alpha_j}\sin(\alpha_j t)\right] + \frac{\hat{v}_{1,j}}{\alpha_j}e^{-\frac{\eta t}{2}}\sin(\alpha_j t),$$

from which we note that we arrive at exactly the same set of coefficients as in the previous analysis if $\eta = 0$ and the real part of the solution is taken. Using the complex-valued filters

$$\hat{\beta}(\alpha) := \frac{1}{T}\int_0^T e^{(i\omega - \eta/2)t}\cos(\alpha t)dt, \quad \hat{\gamma}(\alpha) := \frac{1}{T}\int_0^T e^{(i\omega-\eta/2)t}\sin(\alpha t)dt,$$

we can write the iteration as

$$\begin{pmatrix} v_{0,j}^{n+1} \\ v_{1,j}^{n+1} \end{pmatrix} = \Pi \begin{pmatrix} v_{0,j}^{n} \\ v_{1,j}^{n} \end{pmatrix} = \left(I - \hat{B}_j\right)\begin{pmatrix} u_j \\ i\omega u_j \end{pmatrix} + \hat{B}_j \begin{pmatrix} v_{0,j}^{n} \\ v_{1,j}^{n} \end{pmatrix}, \tag{3.21}$$

where if $\hat{\beta}_j = \hat{\beta}(\alpha_j)$ and $\hat{\gamma}_j = \hat{\gamma}(\alpha_j)$ then

$$\hat{B}_j = \begin{pmatrix} \hat{\beta}_j + \frac{\eta}{2\alpha_j}\hat{\gamma}_j & \hat{\gamma}_j/\alpha_j \\ -(\alpha_j + \frac{\eta^2}{4\alpha_j})\hat{\gamma}_j & \hat{\beta}_j - \frac{\eta}{2\alpha_j}\hat{\gamma}_j \end{pmatrix}.$$

As in the previous analysis, we require that the spectral radius of $\hat{B}_j$ be less than one. The eigenvalues are given by $\hat{\mu}_j = \hat{\beta}_j \pm i\hat{\gamma}_j$ so that by definition

$$|\hat{\mu}_j| = |\hat{\beta}_j \pm i\hat{\gamma}_j| = \left| \frac{1}{T} \int_0^T e^{i(\omega \pm \alpha_j)t} e^{-\eta t/2}\, dt \right| \leq \frac{2}{\eta T}(1 - e^{-\eta T/2}) < 1, \tag{3.22}$$

given that $\eta > 0$. □

Thus the iteration **always** converges in the damped case. From (3.22) we see that for a desired fixed rate of convergence the damping parameter η must grow proportionally to ω, and that frequency-independent convergence is achieved by choosing $\eta = \mathcal{O}(\omega)$.

3.3 Analysis of Higher Order Time Stepping Schemes for the Discrete Iteration

We introduce the temporal grid points $t_n = n\Delta t$ and a spatial grid with N points together with the vector $w^n \in \mathbb{R}^N$ containing the grid function values of the approximation at $t = t_n$. We also let $f \in \mathbb{R}^N$ hold the corresponding values of the right hand side. The discretization of the continuous spatial operator $-\nabla \cdot (c^2(x)\nabla)$, including the boundary conditions, is denoted L_h and it can be represented as an $N \times N$ matrix. The values $-\nabla \cdot (c^2(x)\nabla w)$ are then approximated by $L_h w^n$. As in the continuous case, we assume L_h has the eigenmodes (λ_j^2, ϕ_j), such that $L_h\phi_j = \lambda_j^2\phi_j$ for $j = 1, \ldots, N$, where all λ_j are strictly positive and ordered as $0 \leq \lambda_1 \leq \ldots \leq \lambda_N$.

We let the discrete Helmholtz solution u be defined through

$$-L_h u + \omega^2 u = f.$$

The numerical approximation of the iteration operator is denoted Π_h, and it is implemented as follows. Given $v \in \mathbb{R}^N$, we use the leap frog method to solve the wave equation and add in higher

order corrections as in the Modified Equation (ME) approach [99, 5]. For a general $2m$ scheme, recall that via Taylor expansion

$$\frac{w^{n+1}-2w^n+w^{n-1}}{\Delta t^2} = w_{tt} + 2\sum_{k=2}^{\infty} \frac{\Delta t^{2(k-1)}}{(2k)!}\frac{\partial^{2k}}{\partial t^{2k}} w^n.$$

Then using the PDE to convert time derivatives to spatial derivatives we get the expression

$$\frac{\partial^{2k}}{\partial t^{2k}} w^n = L_h^k w^n + \cos(\omega t_n)\sum_{\ell=0}^{k-1}(-1)^{k+\ell}\omega^{2(k-\ell-1)} L_h^\ell f,$$

for $k = 1, 2, \ldots$. Then for a $2m$ order scheme we have

$$\frac{w^{n+1}-2w^n+w^{n-1}}{\Delta t^2} - 2\sum_{k=2}^{m}\frac{\Delta t^{2(k-1)}}{(2k)!}\left[L_h^k w^n + \cos(\omega t_n)\sum_{\ell=0}^{k-1}(-1)^{k+\ell}\omega^{2(k-\ell-1)}L_h^\ell f\right] = L_h w^n - f\cos(\omega t_n), \tag{3.23}$$

with time step $\Delta t = T/M$ for some integer M, and initial data

$$w^0 = v, \qquad w^{-1} = v + \sum_{k=1}^{m}\frac{(-1)^k \Delta t^{2k}}{(2k)!}\left[-L_h^k v + \sum_{\ell=0}^{k-1}(-1)^\ell \omega^{2(k-\ell-1)} L_h^\ell f\right].$$

The trapezoidal rule is then used to compute $\Pi_h v$,

$$\Pi_h v = \frac{2\Delta t}{T}\sum_{n=0}^{M}\eta_n\left(\cos(\omega t_n) - \frac{1}{4}\right)w^n, \qquad \eta_n = \begin{cases} \frac{1}{2}, & n = 0 \text{ or } n = M, \\ 1, & 0 < n < M. \end{cases} \tag{3.24}$$

We may then prove the following theorem that is a generalization of Theorem 2.4 of [14].

Theorem 3.3.1. *Suppose there are no resonances, such that $\delta_h = \min_j |\lambda_j - \omega|/\omega > 0$. Moreover, assume that Δt satisfies the stability and accuracy requirements*

$$\Delta t < \frac{2}{\lambda_N + 2\omega/\pi}, \qquad \Delta t\omega \le \min(\delta_h, 1). \tag{3.25}$$

Then the fixed point iteration $v^{(k+1)} = \Pi_h v^{(k)}$ with $v^{(0)} = 0$ converges to v^∞ which is a solution to the discretized Helmholtz equation with the modified frequency $\tilde{\omega}$,

$$-L_h v^\infty + \tilde{\omega}^2 v^\infty = f, \qquad \sin^2(\omega\Delta t/2) = \sum_{j=1}^{m}\frac{(-1)^{j+1}(\Delta t\tilde{\omega})^{2j}}{2(2j)!} = \sin^2(\tilde{\omega}\Delta t/2) + \mathcal{O}(\Delta t^{2m+2}),$$

where $2m$ is the order of the ME time stepping scheme. Moreover, $|\omega - \tilde{\omega}| = \mathcal{O}(\Delta t^{2m})$, $\|u - v^\infty\| = \mathcal{O}(\Delta t^{2m})$, and the convergence rate is at least $\rho_h = \max(1 - 0.3\delta_h^2, 0.6)$.

Proof. We expand all functions in eigenmodes of L_h,

$$w^n = \sum_{j=1}^{N} \hat{w}_j^n \phi_j, \qquad f = \sum_{j=1}^{N} \hat{f}_j \phi_j, \qquad u = \sum_{j=1}^{N} \hat{u}_j \phi_j, \qquad v = \sum_{j=1}^{N} \hat{v}_j \phi_j, \qquad v^\infty = \sum_{j=1}^{N} \hat{v}_j^\infty \phi_j.$$

Then the Helmholtz eigenmodes of u and v^∞ satisfy

$$\hat{u}_j = \frac{\hat{f}_j}{\omega^2 - \lambda_j^2}, \qquad \hat{v}_j^\infty = \frac{\hat{f}_j}{\tilde{\omega}^2 - \lambda_j^2}.$$

We note that $\tilde{\omega}$ is well-defined, with a verification in Appendix .7. Moreover, $\tilde{\omega}$ is not resonant and $\hat{v}_j^\infty$ is well-defined for all j, since by (11) and (3.25)

$$|\tilde{\omega} - \lambda_j| \geq |\omega - \lambda_j| - |\tilde{\omega} - \omega| \geq \omega\delta_h - \frac{\Delta t^{2m}\omega^{2m+1}}{(2m+2)!} \geq \omega\left(\delta_h - \frac{1}{(2m+2)!}\min(\delta_h, 1)^{2m}\right) > 0.$$

The wave solution eigenmodes to (3.23) are given by the difference equation

$$\hat{w}_j^{n+1} - 2\hat{w}_j^n + \hat{w}_j^{n-1} + 2\left[\sum_{k=1}^{m} \frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!}\right]\hat{w}_j^n = 2\left[\sum_{k=1}^{m} \frac{(-1)^k \Delta t^{2k}}{(2k)!}\sum_{\ell=0}^{k-1}\omega^{2(k-\ell-1)}\lambda_j^{2\ell}\right]\hat{f}_j \cos(\omega t_n), \tag{3.26}$$

with initial data

$$\hat{w}_j^0 = \hat{v}_j, \qquad \hat{w}_j^{-1} = \hat{v}_j\left(1 + \sum_{k=1}^{m}\frac{(-1)^k\Delta t^{2k}}{(2k)!}\lambda_j^{2k}\right) + \hat{f}_j\left(\sum_{k=1}^{m}\frac{(-1)^k\Delta t^{2k}}{(2k)!}\sum_{\ell=0}^{k-1}\omega^{2(k-\ell-1)}\lambda_j^{2\ell}\right).$$

By (3.25), the discrete solution is stable and given by (the verification of which is found in Appendix .6)

$$\hat{w}_j^n = (\hat{v}_j - \hat{v}_j^\infty)\cos(\tilde{\lambda}_j t_n) + \hat{v}_j^\infty \cos(\omega t_n), \tag{3.27}$$

where $\tilde{\lambda}_j$ is well-defined by the relation (verification in (10) of Appendix .7)

$$\sin^2(\tilde{\lambda}_j \Delta t/2) = \sum_{k=1}^{m} \frac{(-1)^{k+1}\left(\Delta t \lambda_j\right)^{2k}}{2(2k)!}. \tag{3.28}$$

Since $|\omega - \tilde{\omega}| \leq \Delta t^2\omega^3/24$, the following lemma (restated from [14]) gives convergence of the discrete iteration.

Lemma 3.3.2. *Under the assumptions of Theorem 3.3.1,*

$$\max_{1\leq j\leq N} |\beta_h(\tilde{\lambda}_j)| \leq \rho_h =: \max(1 - 0.3\delta_h^2, 0.6). \tag{3.29}$$

From (11), it follows that $|\omega - \tilde{\omega}| = \mathcal{O}(\Delta t^{2m})$. Letting $e = u - v^{\infty}$ be the error in the discrete solutions, the components of the error in the basis of the Laplacian satisfy

$$\begin{aligned}
|e_j| = |\hat{u}_j - \hat{v}_j^{\infty}| = \left| \hat{f}_j \left(\frac{1}{\omega^2 - \lambda_j^2} - \frac{1}{\tilde{\omega}^2 - \lambda_j^2} \right) \right| &= \left| \hat{f}_j \left(\frac{\tilde{\omega}^2 - \omega^2}{(\tilde{\omega}^2 - \lambda_j^2)(\omega^2 - \lambda_j^2)} \right) \right| \\
&= \left| \hat{f}_j \left(\frac{(\tilde{\omega} - \omega)(\tilde{\omega} + \omega)}{(\tilde{\omega} - \lambda_j)(\tilde{\omega} + \lambda_j)(\omega - \lambda_j)(\omega + \lambda_j)} \right) \right| \\
&= \left| \hat{f}_j \left(\frac{(\tilde{\omega} - \omega)(\tilde{\omega} + \omega)}{\tilde{\omega}\tilde{\delta}_j(\tilde{\omega} + \lambda_j)\omega\delta_j(\omega + \lambda_j)} \right) \right| \\
&\leq \left| \hat{f}_j(\tilde{\omega} - \omega) \right| \left(\frac{(\tilde{\omega} + \omega)}{\tilde{\omega}\tilde{\delta}^*(\tilde{\omega} + \lambda_1)\omega\delta^*(\omega + \lambda_1)} \right),
\end{aligned}$$

where $\delta^* = \min_j \delta_j = (\omega - \lambda_j)/\omega$ and $\tilde{\delta}^* = \min_j \tilde{\delta}_j = (\tilde{\omega} - \lambda_j)/\omega$. This gives

$$\|u - v^{\infty}\|_2 = \|e\|_2 \leq |\tilde{\omega} - \omega| \left(\frac{(\tilde{\omega} + \omega)}{\tilde{\omega}\tilde{\delta}^*(\tilde{\omega} + \lambda_1)\omega\delta^*(\omega + \lambda_1)} \right) \|f\|_2 \approx \mathcal{O}(\Delta t^{2m}),$$

since δ_*, $\tilde{\delta}_* > 0$, concluding the proof of the theorem. □

Remark 3.3.1. *As alluded to in Remark 6 of [14], knowledge of how a particular discretization approximates the eigenvalues of the continuous operator can be used to improve the iteration. In fact, the above error due to time discretization can be removed by defining $\bar{\omega}$ by the relation*

$$\sin^2(\bar{\omega}\Delta t/2) = \sum_{k=1}^{m} \frac{(-1)^{k+1} (\Delta t\omega)^{2k}}{2(2k)!}.$$

Then using $f\cos(\bar{\omega}t_n)$ instead of $f\cos(\omega t_n)$ in the time stepping (3.23)*, in addition to the modified trapezoidal quadrature rule (first introduced in [93])*

$$\Pi_h v = \frac{2\Delta t}{T} \sum_{n=0}^{M} \eta_n \frac{\cos(\omega t_n)}{\cos(\bar{\omega}t_n)} \left(\cos(\omega t_n) - \frac{1}{4} \right) w^n, \qquad \eta_n = \begin{cases} \frac{1}{2}, & n = 0 \text{ or } n = M, \\ 1, & 0 < n < M, \end{cases}$$

gives that the limit will be precisely the discrete Helmholtz solution, $v^{\infty} = u$, as long as the time step size is chosen so that $\cos(\bar{\omega}t_n) \neq 0$. Moreover, the first timestep restriction of (3.25) arising from the usual CFL condition for the second order scheme may be relaxed (expressions for which may be found in [56]) though the condition $\Delta t\omega \leq \min(\delta_h, 1)$ may be more restrictive for problems close to resonance. We additionally note that in [101] an alternative approach to remove time-discretization

error was presented, however the approach modified the timestepping scheme whereas we modify the frequency of the forcing and update our quadrature rule.

3.4 Wave Equation Solvers

In this section we briefly outline the numerical methods we use in the experimental section below. We consider both discontinuous Galerkin finite element solvers and finite difference solvers. In all the experiments we always use the trapezoidal rule to compute the integral in the WaveHoltz iteration.

3.4.1 The Energy Based Discontinuous Galerkin Method

Our spatial discretization is a direct application of the formulation described for general second order wave equations in [9, 10]. Here we outline the spatial discretization for the special case of the scalar wave equation in one dimension and refer the reader to [9] for the general case.

The energy of the scalar wave equation is

$$H(t) = \int_D \frac{v^2}{2} + G(x, w_x) dx,$$

where

$$G(x, w_x) = \frac{c^2(x) w_x^2}{2},$$

is the potential energy density, v is the velocity (not to be confused with the iterates v^n above) or the time derivative of the displacement, $v = w_t$. The wave equation, written as a second order equation in space and first order in time then takes the form

$$\begin{aligned} w_t &= v, \\ v_t &= -\delta G, \end{aligned}$$

where δG is the variational derivative of the potential energy

$$\delta G = -(G_{w_x})_x = -(c^2(x) w_x)_x.$$

For the continuous problem the change in energy is

$$\frac{dH(t)}{dt} = \int_D v v_t + w_t (c^2(x) w_x)_x \, dx = [w_t (c^2(x) w_x)]_{\partial D}, \tag{3.30}$$

where the last equality follows from integration by parts together with the wave equation. Now, a variational formulation that mimics the above energy identity can be obtained if the equation $v - w_t = 0$ is tested with the variational derivative of the potential energy. Let Ω_j be an element and $\Pi^s(\Omega_j)$ be the space of polynomials of degree s, then the variational formulation on that element is:

Problem 2. *Find $v^h \in \Pi^s(\Omega_j)$, $w^h \in \Pi^r(\Omega_j)$ such that for all $\psi \in \Pi^s(\Omega_j)$, $\phi \in \Pi^r(\Omega_j)$*

$$\int_{\Omega_j} c^2 \phi_x \left(\frac{\partial w_x^h}{\partial t} - v_x^h \right) dx = [c^2 \phi_x \cdot n \left(v^* - v^h \right)]_{\partial \Omega_j}, \tag{3.31}$$

$$\int_{\Omega_j} \psi \frac{\partial v^h}{\partial t} + c^2 \psi_x \cdot w_x^h \, dx = [\psi \, (c^2 \, w_x)^*]_{\partial \Omega_j}. \tag{3.32}$$

Let $[[f]]$ and $\{f\}$ denote the jump and average of a quantity f at the interface between two elements, then, choosing the numerical fluxes as

$$\begin{aligned} v^* &= \{v\} - \tau_1 [[c^2 \, w_x]] \\ (c^2 \, w_x)^* &= \{c^2 \, w_x\} - \tau_2 [[v]], \end{aligned}$$

will yields a contribution $-\tau_1 ([[c^2 \, w_x]])^2 - \tau_2 ([[v]])^2$ from each element face to the change of the discrete energy

$$\frac{dH^h(t)}{dt} = \frac{d}{dt} \sum_j \int_{\Omega_j} \frac{(v^h)^2}{2} + G(x, w_x^h).$$

Physical boundary conditions can also be handled by appropriate specification of the numerical fluxes, see [9] for details. The above variational formulation and choice of numerical fluxes results in an energy identity similar to (3.30). However, as the energy is invariant to certain transformations the variational problem does not fully determine the time derivatives of w^h on each element and independent equations must be introduced. In this case there is one invariant and an independent equation is $\int_{\Omega_j} \left(\frac{\partial w^h}{\partial t} - v^h \right) = 0$. For the general case and for the elastic wave equation see [9] and [10].

In this chapter we always choose $\tau_i > 0$ (so-called upwind or Sommerfeld fluxes) and we always choose the approximation spaces to be of the same degree $r = s$. These choices result in methods that are $r + 1$ order accurate in space.

3.4.2 Symmetric Interior Penalty Discontinuous Galerkin Method

In addition to the above energy DG method, we also consider the Symmetric Interior Penalty DG (SIPDG) discretization, [62], for examples in two dimensions. The bilinear form in this case is

$$a_h(u,v) = \sum_{K\in\mathcal{T}_h}\int_K c^2\nabla u\cdot\nabla v\,dx - \sum_{f\in\mathcal{F}_h}\int_F [[u]]\cdot\{c^2\nabla v\}\,ds - \sum_{f\in\mathcal{F}_h}\int_F [[v]]\cdot\{c^2\nabla u\}\,ds + \sum_{f\in\mathcal{F}_h}\int_F \gamma h_F^{-1}c^2[[u]]\cdot[[v]]\,ds,$$

where $\mathcal{T}_h$ is a collection of triangular elements, $\mathcal{F}_h$ is the collection of element faces, h_F is the diameter of the edge or face F, and γ is the interior penalty stabilization parameter which must be chosen to be sufficiently large to ensure the system is positive-definite.

3.4.3 Finite Difference Discretizations

For the finite difference examples in a single dimension, we consider discretizations by uniform grids $x_i = x_L + ih_x,$, with $i = -1,\ldots,n+1$ and $h_x = (x_R - x_L)/n$. To impose impedance boundary conditions of the form $w_t \pm \vec{n}\cdot\nabla w = 0$ we evolve the wave equation as a first order system in time according to the semi-discrete approximation

$$\frac{dv_i(t)}{dt} = (D_+D_-)w_i, \tag{3.33}$$

$$\frac{dw_i(t)}{dt} = v_i, \tag{3.34}$$

and for the boundaries we find the ghost point values by enforcing

$$v_0 - D_0w_0 = 0, \quad v_n - D_0w_n = 0. \tag{3.35}$$

Here we have used the standard forward, backward and centered finite difference operators, for example $hD_+w_i = w_{i+1} - wi$ etc.

3.4.4 Time Discretization

For some of the numerical examples in a single dimension, we use either an explicit second order accurate centered discretization of w_{tt} or use the higher order corrected ME methods described in Section 3.3.

For the DG discretizations we employ Taylor series time-stepping in order to match the order of accuracy in space and time. Assuming that all the degrees of freedom have been assembled into a vector $\mathbf{w}$ we can write the semi-discrete method as $\mathbf{w}_t = Q\mathbf{w}$ with Q being a matrix representing the spatial discretization. Assuming we know the discrete solution at the time t_n we can advance it to the next time step $t_{n+1} = t_n + \Delta t$ by the simple formula

$$\begin{aligned} \mathbf{w}(t_n + \Delta t) &= \mathbf{w}(t_n) + \Delta t \mathbf{w}_t(t_n) + \frac{(\Delta t)^2}{2!}\mathbf{w}_{tt}(t_n)\dots \\ &= \mathbf{w}(t_n) + \Delta t Q\mathbf{w}(t_n) + \frac{(\Delta t)^2}{2!}Q^2\mathbf{w}(t_n)\dots \end{aligned}$$

The stability domain of the Taylor series which truncates at time derivative number N_{T} includes the imaginary axis if $\mathrm{mod}(N_{\mathrm{T}}, 4) = 3$ or $\mathrm{mod}(N_{\mathrm{T}}, 4) = 0$. However as we use a slightly dissipative spatial discretization the spectrum of our discrete operator will be contained in the stability domain of all sufficiently large choices of N_{T} (i.e. the N_{T} should not be smaller than the spatial order of approximation). Note also that the stability domain grows linearly with the number of terms.

3.5 Numerical Examples

In this section we illustrate the properties of the proposed iteration and its Krylov accelerated version by a sequence of numerical experiments in one and two spatial dimensions.

3.5.1 Examples in One Dimension

3.5.1.1 Convergence Rate for Impedance Boundary Conditions

In [14], an application of Weyl asymptotics [109] revealed that the mininal relative gap to resonance, $\delta = \min_j |\omega - \lambda_j|/\omega$ where λ_j^2 are the eigenvalues of the Laplacian, shrinks as ω^{-d} where

d is the spatial dimension of the Helmholtz problem of interest. Analysis of the symmetric, positive definite formulation of the iteration then yielded a convergence rate of $1 - \mathcal{O}(\delta^2) \approx 1 - \mathcal{O}(\omega^{-2d})$. However, numerical experiments with Helmholtz problems with certain open/outflow boundary conditions suggest a much more attractive convergence rate than the unacceptable $1 - \mathcal{O}(\omega^{-2d})$ rate. A natural question then is whether or not this seemingly pessimistic convergence rate can be observed for outflow boundary conditions which are much more common in practical applications.

To that end, we consider a set of sample Helmholtz problems in a single spatial dimension with a constant (normalized) speed of sound, $c = 1$, in the domain $0 \leq x \leq 2$ where we impose the impedance boundary condition

$$w_t + \vec{n} \cdot w_x = 0,$$

which we note is equivalent to the Sommerfeld radiation condition. The Helmholtz problem under consideration has no forcing and so $f = 0$. In this case the solution is not unique (we have $\sin(\omega x)$ and $\cos(\omega x)$ as solutions), but we nevertheless may apply the WaveHoltz iteration. We formulate the wave equation in first order form and apply the extended iteration (3.7) since the boundary conditions do not conserve energy. The Laplacian is discretized with a standard three-point finite difference approximation, and a fourth order Taylor scheme is used for timestepping. We define the initial conditions as

$$v_0(x) = \sin(\omega x) - \frac{1}{2}\left(\sin((\omega + 2\pi)x) + \sin((\omega - 2\pi)x)\right), \quad v_1(x) = -\frac{d}{dx} v_0(x),$$

which are shown in Figure 3.1.

By definition, $\|\mathcal{S}\|_2 = \sup_{\|z\|_2 \neq 0} \|\mathcal{S}z\|_2 / \|z\|_2 \geq \|\mathcal{S}z^0\|_2 / \|z^0\|_2$ so that if $\|Sz^0\|_2 / \|z^0\| \approx 1 - \mathcal{O}(\omega^{-2})$ is observed then the estimate of the spectral radius of the fixed point operator $\mathcal{S}$ is tight even for the problem with impedance boundary conditions. We consider a sweep of Helmholtz frequencies $\omega = 10\pi, 15\pi, 20\pi, \ldots, 120\pi$ with fifty points per wavelength and a CFL number of $1/10$ for the solution of the wave equation. The results of this experiment are shown in Figure 3.1 below.

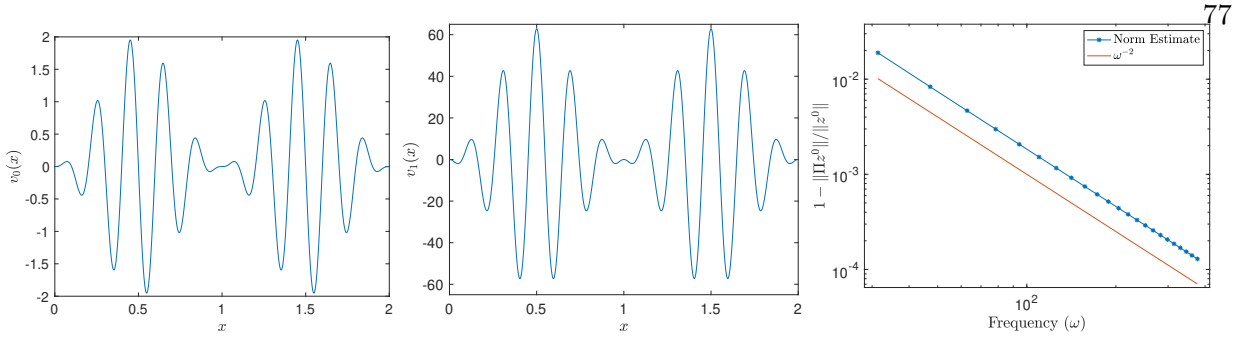


Figure 3.1: (Left, Middle) The initial conditions v_0 and v_1 for a Helmholtz frequency of $\omega = 10\pi$. (Right) The estimate of the quantity $1 - \|\mathcal{S}\|$ with increasing Helmholtz frequency ω.

On the left of Figure 3.1 we see the first part of the initial condition v^0 for a frequency of $\omega = 10\pi$. We note that this specific initial condition is constructed such that it is close to a resonant mode - which the filter-transfer function β weakly damps - as well as being close to zero at the boundary so that a negligible amount of energy exits the system due to the impedance boundary conditions in a single iteration. These two defining characteristics of the initial condition lead to the norm estimate of the fixed-point iteration operator $\mathcal{S}$ on the right of Figure 3.1. We observe that the norm of $\mathcal{S}$ does indeed approach unity at a rate of ω^{-2}, as predicted by theory. Thus, while the preceeding analysis "artificially" leveraged energy-conserving boundary conditions to obtain an estimate of the convergence rate for open problems, it is possible to realize the 'worst-case' rate implied by the energy-conserving regime.

Remark 3.5.1. *We note that the estimate for the convergence rate is a* ***pointwise*** *estimate. Repeated application of the fixed-point iteration will (eventually) remove the modes close to resonance and a faster convergence rate is observed. In Figure 3.2 we repeat the above experiment for the frequencies* $\omega = 10\pi, 40\pi,$ *and* 70π *but continue the iteration until the iterates converge to the zero solution. We observe that after an initial phase the rate of convergence of the iterates to the solution increases significantly since the data has propagated and exited the domain. We believe that the average behavior over many fixed-point iterations leads to the much more attractive rates seen in the Krylov-accelerated numerical experiments of [14]. Moreover, this example was pathologically*

constructed and we note that so far we have been unable to construct initial conditions to realize the worst-case rate in higher than one dimension.

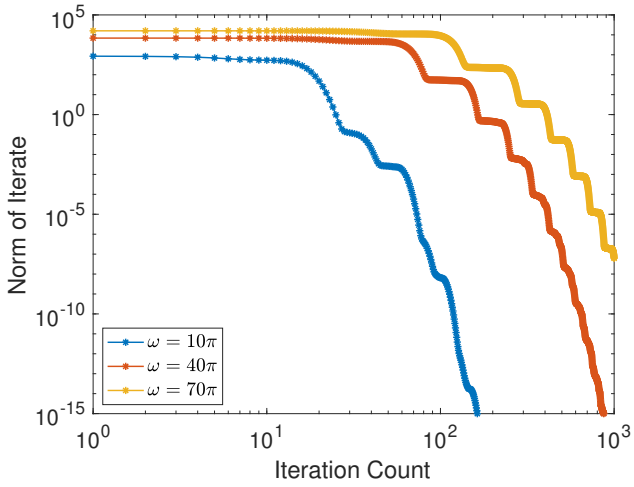


Figure 3.2: The norm of WaveHoltz iterates for increasing Helmholtz frequencies of $\omega = 10\pi, 40\pi$, and 70π for the adversarial example of Figure 3.1.

Assuming radially symmetric solutions to the Helmholtz equation, it is possible to cast higher dimensional problems as 1D problems. Specifically, the wave equation in cylindrical/spherical coordinates is

$$\begin{aligned}
&\frac{\partial^2 w}{\partial t^2} = \frac{\partial^2 w}{\partial r^2} + \frac{\alpha}{r}\frac{\partial w}{\partial r}, \quad r \in \Omega, \quad 0 \le t \le T, \\
&\frac{\partial w}{\partial r}(t,0) = 0, \quad 0 \le t \le T, \\
&\frac{\partial w}{\partial n}(t,r) + \frac{\partial w}{\partial t}(t,r) = 0, \quad x \in \partial\Omega,
\end{aligned}$$

where $\alpha = 1$ corresponds to cylindrical coordinates and $\alpha = 2$ corresponds to spherical. We use a second order finite difference discretization (see [89] for details) with $\Omega = [0,1]$. The initial condition is analagous to the previous example,

$$v_0(r) = \sin(\omega(r+1)) - \frac{1}{2}\left(\sin((\omega+2\pi)(r+1)) + \sin((\omega-2\pi)(r+1))\right), \quad v_1(r) = -\frac{d}{dr}v_0(r).$$

We consider a set of frequencies $10\pi, 11\pi, \ldots, 30\pi$ and use fifty points per wavelength in the computation with a CFL of 1/100. Below we show the results of the experiment.

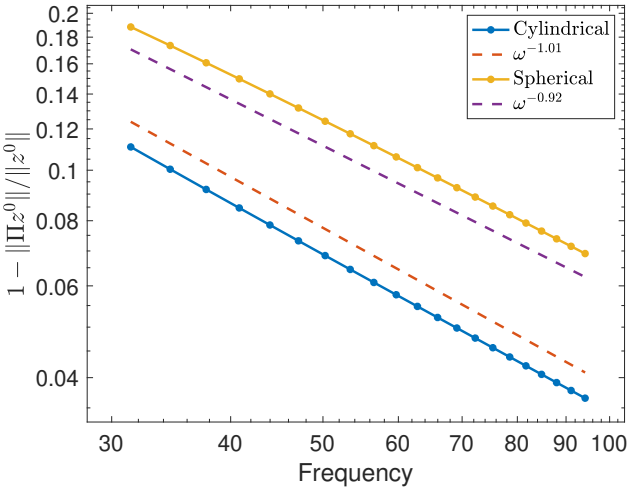


Figure 3.3: The estimate of the quantity $1 - \|\mathcal{S}\|$ with increasing Helmholtz frequency ω for a radially symmetric initial condition.

From Figure 3.3 we observe that the norm of $\mathcal{S}$ approaches unity at a nearly linear rate in the frequency ω in 2D and a sublinear rate for the 3D problem, both of which are more favorable than the quadratic rate in a single spatial dimension.

Remark 3.5.2. *From Figure 3.3 it is clear that with a fixed discretization and initial condition, the convergence rate improves with increasing dimension. This is perhaps unsurprising given an increase in the local energy decay rate for the wave equation from two to three dimensions, along with a richer set of directions in which waves may propagate and leave the domain.*

3.5.1.2 Time Discretization

We consider solving the Helmholtz equation with $c = 1$ and constant exact solution

$$u(x) = 1, \quad 0 \leq x \leq 1.$$

We take the frequency to be $\omega = 1$ and consider Dirichlet boundary conditions. We discretize the Laplacian with the standard three-point finite difference stencil and note that there is no error (aside

from truncation errors) in the solution by a direct solution of the discrete Helmholtz equation. We use a centered modified equation timestepping scheme of both second and fourth order, with both the original frequency and a modified frequency $\tilde{\omega}$ with corresponding quadrature to remove time discretization errors. We use the WaveHoltz iteration as a fixed-point iteration with a convergence criterion that the relative L_2 norm between successive iterations is smaller than 10^{-13}. Using the

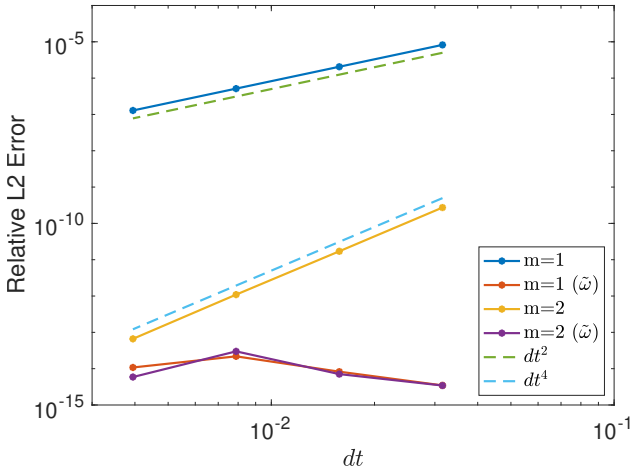


Figure 3.4: Convergence of the discrete WaveHoltz solution to the true solution of the discrete Helmholtz problem with fixed spatial discretization. Solid lines indicate relative errors between discrete solutions. The blue and yellow solid lines indicate relative errors between the usual discrete WaveHoltz solution and the true solution, and the red and purple solid lines indicate relative errors for the frequency corrected solution.

original frequency in the calculation, we see from Figure 3.4 that the WaveHoltz solution converges to the discrete Helmholtz solution with the same order as that of the timestep scheme used. With the modified frequency and quadrature, however, we see that the WaveHoltz iteration converges to the discrete Helmholtz solution up to roundoff errors.

Remark 3.5.3. *While only centered timestepping schemes are presented here, this approach can be extended to arbitrary timesteppers. A careful discrete analysis of the iteration isolated to a single eigenmode of the wave solution reveals what the modified frequency should be, and a modified quadrature as outlined above removes the time discretization error from the converged WaveHoltz solution. Thus, the choice of a timestepper need not need be restricted to have the same order as the spatial discretization. With a corrected scheme it may be more advantageous to take as large a*

timestep as possible with a low order timestepper.

3.5.1.3 Convergence Rate for Damped Helmholtz Equations

To study how the number of iterations scale with the Helmholtz frequency ω we solve the wave equation on the domain $x \in [-6, 6]$ with constant wave speed $c^2(x) = 1$ and with a forcing

$$f(x) = \omega^2 e^{-(\omega x)^2},$$

that results in the solution being $\mathcal{O}(1)$ for all ω. We discretize using the energy based DG method discussed above and use upwind fluxes which adds a small amount of dissipation. We keep the number of degrees of freedom per wave length fixed by letting the number of elements be $5\lceil\omega\rceil$. We always take the polynomial degree to be 7, the number of Taylor series terms in the timestepping to be 6, and use WHI accelerated by GMRES without restarts.

We report the number of iterations it takes to reach a GMRES residual smaller than 10^{-10} for the six possible combinations of Dirichlet, Neumann and impedance boundary conditions for 200 frequencies distributed evenly from 1 to 100. The results for three levels of damping are displayed in Figure 3.5. On the left and middle of Figure 3.5 are damping parameters of $1/2\omega$ and $1/2$ respectively, from which it is clear that the scaling is sub-linear with increasing frequency. On the right in Figure 3.5 are results from a damping parameter that grows with frequency, $\omega/2$, which demonstrates a number of iterations that is both frequency independent and modest for a given GMRES tolerance. Interestingly, in this case the curve for each set of boundary conditions collapses to the same curve so that the iteration is insensitive to boundary conditions for a sufficiently large damping parameter.

Remark 3.5.4. *As seen in the previous chapter, the impedance-impedance conditions take the fewest iterations to reach convergence for lower levels of damping. We point out the preceeding analysis assumes energy-conserving boundary conditions to obtain estimates on the convergence rate of WaveHoltz as a fixed-point iteration. A different approach without the need for a Laplacian with a point-spectrum is needed to obtain rates depending on the specific boundary conditions.*

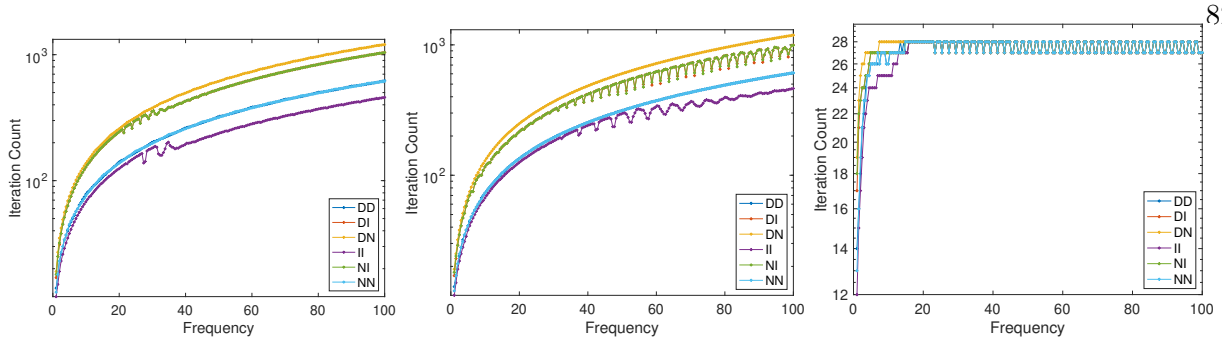


Figure 3.5: Number of iterations as a function of ω for different boundary conditions and damping parameters. Left: $\eta = 1/2\omega$, Middle: $\eta = 1/2$, Right: $\eta = \omega/2$. In the above legends each entry is made up of a two letter string where the first letter indicates the boundary condition on the left at $x = -6$, and the second letter indicates the boundary condition on the right at $x = 6$. Here D indicates Dirichlet, N indicates Neumann, and I indicates impedance/Sommerfeld conditions.

3.5.2 Examples in Two Dimensions

In this section we present experiments in two space dimensions. For the following examples, we consider solving the Helmholtz equation for the wedge model which we adapt from [45, 95]. The domain is the rectangle $[0, 600] \times [0, 1000]$ with the (discontinuous) speed of sound

$$c(x) = \begin{cases} c_1 = 2100, & y \leq x/6 + 400, \\ c_2 = 1000, & x/6 + 400 \leq y \leq 800 - x/3, \\ c_3 = 2900, & \text{else.} \end{cases}$$

The domain and mesh used for the examples is shown in Figure 3.6, where the blue region corresponds to c_1, the green region with c_2, and the magenta region with c_3. On the boundary of the rectangle we impose the impedance boundary condition $w_t + c\nabla w \cdot \vec{n} = 0$. For the spatial discretization we use the SIPDG method with a penalty parameter choice of $\gamma = (p+1)^2$, where $p = 4$ is the polynomial order used in each element which results in a fifth order method. In time we use a fourth order Taylor method for timestepping. For each example, we use the point-source

$$f(x, y) = \omega^2 \delta(|x - x_0|)\delta(|y - y_0|),$$

where $x_0 = 300$, $y_0 = 0$, ω is the Helmholtz frequency, and $\delta(z)$ is the usual Dirac delta function. These examples were implemented in the MFEM finite element discretization library [4].

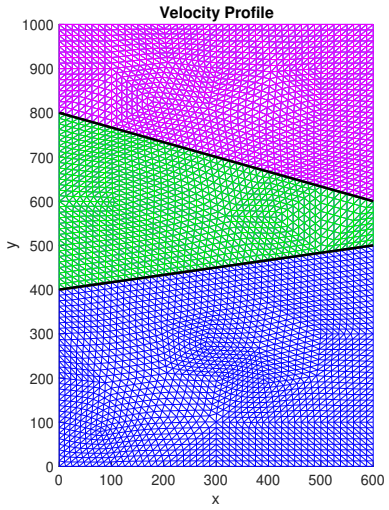


Figure 3.6: Computational domain where the mesh in blue corresponds to a wavespeed of $c = 2100$, the mesh in green corresponds to a speed of $c = 1000$, and the magenta mesh with $c = 2900$. The solid black line is not physical and is meant to more easily distinguish between regions.

3.5.2.1 Convergence for Damped Helmholtz Equations

We again study how the number of GMRES accelerated WHI iterations scale with the Helmholtz frequency ω for the exemplary wedge problem.

We report the number of iterations it takes to reach a GMRES residual smaller than 10^{-10} for the frequencies $1, 2, \ldots, 100$, with damping $\eta = \omega/2$ with either impedance or Neumann conditions on all sides of the rectangular domain.

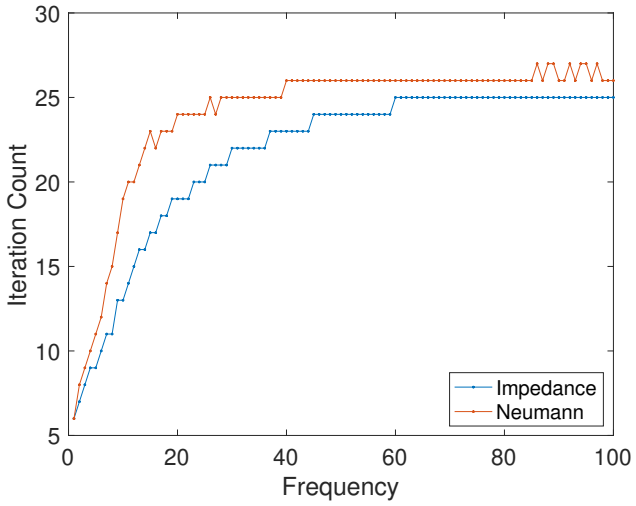


Figure 3.7: Number of iterations to reach a GMRES tolerance of 10^{-10} for the wedge problem in 2D with all Neumann or all impedance boundary conditions.

The results for this experiment are shown in Figure 3.7, from which it is clear that the number of iterations is essentially independent of frequency for larger frequencies as was the case in a single spatial dimension. We again note that energy-conserving boundary conditions require more iterations than the impedance case even in the presence of damping.

For a final example, in Figure 3.8 we display the solution of the damped (and undamped) Helmholtz equation using the GMRES accelerated WHI for a frequency of $\omega = 40\pi$ with damping $\eta = \omega/2$ and 0, respectively.

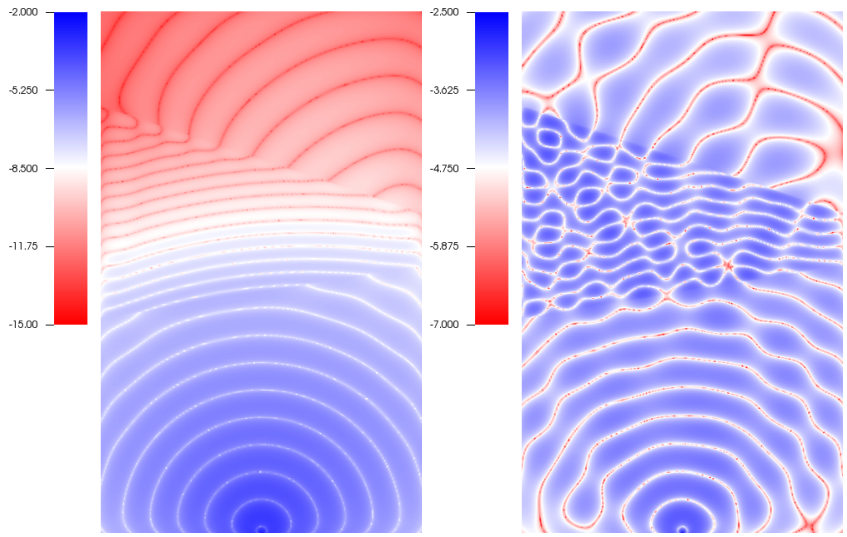


Figure 3.8: In the above we plot the $\log_{10}$ of the absolute value of the real part of the Helmholtz solution with frequency $\omega = 40\pi$ for (Left) damping parameter $\eta = 20\pi$ and (Right) no damping.

3.6 Summary and Future Work

We have presented and extended analysis of the WaveHoltz iteration, an iterative method for solving the Helmholtz equation, applied to wave equations with and without damping. The general iteration has the same rate of convergence as the energy-conserving case presented in [14], but is a more general and appropriate formulation for considering problems with impedance/Sommerfeld boundary conditions. For problems with damping, the WaveHoltz iteration always converges and numerical experiments verify the frequency independent convergence of problems with sufficient levels of damping.

We have provided analysis of the interior impedance problem in a single dimension and constructed an example in which the worst-case convergence rate is realized, despite the numerical results of the previous chapter indicating much more favorable scaling for non-energy conserving boundary conditions.

Finally, here we have only considered acoustic wave propagation. In the following chapter we will apply the WaveHoltz iteration to the elastic Helmholtz equation. Moreover, we have not yet tried to leverage sweeping/domain decomposition ideas here and hope to study the numerical and

theoretical properties of these in the future.

Chapter 4

El WaveHoltz Method

Time harmonic wave propagation problems are notoriously difficult to solve by direct or iterative methods due to the resolution requirements and the indefinite nature of the differential operator, especially at high frequencies. In acoustic media, a prototypical model of time harmonic wave propagation is given by the (heterogeneous) Helmholtz equation

$$\nabla \cdot (c^2(\mathbf{x})\nabla u) + \omega^2 u = f(\mathbf{x}), \qquad \mathbf{x} \in \Omega, \tag{4.1}$$

for a domain Ω, frequency ω, and sound speed $c^2(\mathbf{x})$. The efficient solution of the Helmholtz equation (4.1) via iterative methods is an active area of research with a variety of methods in both the frequency and time domain. We refer to Chapter 2 for a more in-depth overview of the literature on techniques for solving the Helmholtz equation, as well as the review articles [47, 50, 44].

For applications in solid mechanics, seismology and geophysics, however, it is more appropriate to consider the elastic wave equation instead of the acoustic wave equation. In contrast to the literature for the acoustic case, fewer effective solvers and preconditioners are available for time harmonic elastic waves governed by the elastic "Helmholtz" equation (also known as the Navier equation)

$$\rho\omega^2 \mathbf{v} + \nabla \cdot \mathcal{T}(\mathbf{v}) = \mathbf{f}(\mathbf{x}), \quad \mathbf{x} \in \Omega.$$

Here $\mathbf{v} \in \mathbb{R}^d$ is the displacement vector and d the spatial dimension. The elastic wave equation models both pressure and shear waves and, as is the case for the Helmholtz equation (4.1), the above system of equations results in a discretization that is highly indefinite for large frequencies.

As for any wave propagation problem the resolution must increase with the frequency, and here the most stringent resolution constraint comes from the (shorter) shear wave wavelength. This, in tandem with d times the number of unknowns leading to larger storage requirements, necessitates parallel, memory lean, and scalable solvers that must be high order accurate to mitigate dispersive errors, [75], causing the so-called *pollution effect* [16].

While most methods have traditionally focused on solving the Helmholtz equation in the frequency domain (we provide a review of some of these below), an alternative approach is to instead construct iterative solvers in the time domain. One such method, the so-called Controllability Method (CM), was first proposed by Bristeau et al. [29] and has recently received renewed interest in a series of papers by Grote [63, 85, 33]. The CM was extended to elastic media in [87]. The unknown in the CM is the initial data to the wave equation. In the CM this initial data is adjusted so that it produces an approximation to the Helmholtz equation by solving a constrained least-squares minimization where the objective function measures the deviation from time-periodicity. The minimization can be efficiently implemented using the conjugate gradient method, where the gradient is computed by solving the adjoint wave equation backwards in time.

The method we present here is an extension of the WaveHoltz method introduced in [14] for the scalar wave equation. As in the CM, the WaveHoltz iteration (and the Elastic version we denote El WaveHoltz) iteratively updates the initial data to the wave equation but it does so by filtering the wave equation solution over one period (or an integer number of periods). The filtered solution is then used as the next initial data and thus the WaveHoltz method only requires one wave solve per iteration while the CM requires two.

In [14] we show that the (linear) iteration is convergent in both the continuous and discretized setting and that, if formulated as a linear system of equations, the underlying matrix is positive definite. We also showed that for closed waveguides with energy conserving boundary conditions (Dirichlet or Neumann) the matrix is also symmetric as long as the numerical method is symmetric.

We emphasize that the filter used in the WaveHoltz method is a bounded operator and therefore the number of iterations (and the condition number of the problem) does **not** depend on

the gridsize h. This is in contrast to methods that discretize and solve the PDE directly. Such methods typically have a condition number that scales as h^{-2} which makes it increasingly difficult to solve the problem as the solution becomes more accurate.

The analysis in [14] predicted that the WaveHoltz method in d dimensions converges to a fixed tolerance in $\mathcal{O}(\omega^d)$ iterations for energy conserving problems and numerical experiments indicated that it converges in $\mathcal{O}(\omega)$ iterations for open problems. Numerical experiments also indicated that some energy conserving examples may exhibit complexity closer to $\mathcal{O}(\omega^{d-0.5})$ for $d = 2$ and 3. The analytical predictions from [14] are expected to hold here as well and in the experiments we carry out below we observe $\mathcal{O}(\omega^d)$. All these results and observations are independent of grid resolution indicating that our method can be particularly suitable when accurate solutions are required.

In this chapter we focus solely on energy conserving boundary conditions (Dirichlet or normal stress) and leave the cases of impedance and non-reflecting boundary conditions to future work. In addition to introducing El WaveHoltz, we present several new results that are also retroactively applicable to our earlier work on WaveHoltz for the scalar wave equation [14] and Maxwell's equations [93].

First, for the energy conserving boundary conditions the continuous WaveHoltz operator is symmetric positive definite. However, unless the semidiscretized wave equation has the form $\mathbf{u}_{tt} = \mathbf{L}_h\mathbf{u}$ with $\mathbf{L}_h$ SPD the discrete WaveHoltz iteration will not result in a symmetric matrix. We show that for schemes that are symmetric in a weighted inner product there is a simple scaling that can be applied to make the discrete WaveHoltz method symmetric. This then allows the conjugate gradient or the conjugate residual method to be used.

Following the ideas of Stolk [101] we introduce two new two-level time-stepping schemes – one explicit and one implicit – that remove the time-stepping error from the WaveHoltz solution. When either of these time-stepping methods are used the solution to the discrete WaveHoltz method is identical to the solution obtained by directly discretizing the frequency domain equation.

For high frequency, large scale problems, parallel solution of the time harmonic elastic equations is the only feasible option. For a parallel solver to scale well the ratio of communication to

computation should be small. In general, there are two types of communication: a) the local communications between processors to exchange local degrees of freedom needed for stencil operations in the discretization of spatial derivatives, and b) global all-to-all operations such as computing the inner product between two global vectors. The WaveHoltz method has an intrinsic advantage compared to methods that work directly with the frequency domain equation in that the all-to-all communication that is required to update search directions in CG, GMRES etc. only needs to be computed once per $T = 2\pi/\omega$-period. Here we explore the effect of filtering over an additional number of periods to further reduce the number of all-to-all communications.

We believe that the method we propose here is an alternative to previously proposed methods. In particular, El WaveHoltz is easily implemented if an elastic wave equation solver is already available. As we show in the numerical experiments section, El WaveHoltz can be one to two orders of magnitude faster compared to an algebraic multigrid (AMG) preconditioned GMRES solver for the frequency domain equation when using the symmetric interior penalty discontinuous Galerkin implementation available in MFEM [4]. There are, of course, many other solvers available; the question of which method will be most efficient will (most likely) depend on the details of the problem to be solved. We now review some of the methods available in the literature.

One of the most common preconditioners for acoustic problems is the shifted Laplacian preconditioner (SLP), a more thorough review of which can be found in the review article by Erlangga [44]. Perhaps one of the first extensions of the damping preconditioner to elastic media was introduced by Airaksinen et al. [3], in which a finite element spatial discretization for the damped operator is inverted by AMG. A more traditional finite-difference multigrid SLP with line-relaxations was considered by Rizzuti and Mulder [96]. For both of these previous approaches, the effectiveness of a straightforward SLP is degraded for nearly incompressible media as the prolongation operators struggle to approximate the nullspace of the grad-div operator. To address this, a more recent extension was done by Treister [102] in which a mixed-formulation of the elastic Helmholtz equation is considered. While nearly incompressible media could be handled by the methods of [102], this comes at the cost of doubling the number of unknowns as well as additional storage requirements

for precomputing the inverse of relaxation operators.

Another important class of methods for the solution of the Helmholtz equation are domain decomposition (DD) methods, for which we refer the reader to [50] for a review. In the short article [30], it was shown that a classic Schwarz DD with overlap for elastic problems converges for high frequencies, diverges for medium frequencies, and stagnates for small frequencies. Moreover, overlapping DD as a preconditioner for a GMRES accelerated solver exhibits convergence behavior that depends strongly on the frequency ω with degrading performance for increasing frequency. To remedy this, Brunet et al. introduced more general transmission conditions at the boundaries of overlapping domains in [31]. These transmission conditions, together with a sufficiently large enough overlap, yield convergence of the DD method for all frequencies with the exception of $\{\omega/C_{\mathrm{s}}, \omega/C_{\mathrm{p}}\}$, where C_{s} and C_{p} are the shear and pressure wave speeds, respectively.

For unbounded problems one of the most promising classes of preconditioners for the Helmholtz equation are the so-called sweeping preconditioners by Engquist and Ying [40, 41]. These preconditioners construct an LDL^T decomposition by sweeping through the domain layer-by-layer, with the key observation that the application of the Schur complement matrices found in the block diagonal matrix D is equivalent to solving a quasi-1D(2D) problem in 2D(3D). In contrast to the acoustic case, however, the sweeping preconditioner for time harmonic elastic waves, [103], exhibited an increase in the number of iterations with frequency for a heterogeneous media as the moving perfectly matched layer (PML) does not approximate Green's function as well. We note that the stable construction of PML for many elastic problems is still considered an open question [21, 11]. Similar to the sweeping preconditioner, Belonosov et al. [22] construct a preconditioner in 3D with damping that sweeps through the domain along a coordinate axis while additionally homogenizing the medium in each layer. The preconditioner of [22] is inverted using FFT's and is accelerated with BiCGSTAB in the outer loop. As with the sweeping preconditioner, the choice of sweeping direction is important. Thus for problems where heterogeneity is present in all directions this preconditioner is less effective. Yet another solver with a sweeping nature is an extension of the Gordon and Gordon [60] CARP-CG method for Helmholtz problems to elastic media [80]. Despite

its simplicity this method requires a large number of iterations, especially for heterogeneous media or problems with higher Poisson ratios. It should be emphasized that, although successful for unbounded problems, the efficiency of sweeping methods for energy conserving boundary conditions has largely not been demonstrated and their parallel implementation remains cumbersome.

Instead of the LDL^T decomposition used by the sweeping preconditioner, other approaches constructing LU/ILU factorizations and preconditioners are available. In [37] an ILU preconditioner based on wavelet transforms with Gibbs reordering is used in a GMRES accelerated solver (with restarts) for time harmonic elastic waves. Wang et al. introduced a structured multifrontal algorithm using nested dissection based domain decomposition, together with hierarchical semi-separable (HSS) compression for frontal matrices with low off-diagonal ranks in [108]. The use of multilevel sequentially semi-separable (MSSS) matrix structure of the discretized elastic wave equation on Cartesian grids was leveraged in [19] inside of an induced dimension reduction (IDR) accelerated ILU preconditioner. The drawback of LU/ILU methods for elastic Helmholtz problems is the growth in memory and storage requirements.

The rest of this chapter is organized as follows. In Section 4.1 we present the elastic Helmholtz and wave equations. In Section 4.2 we introduce the WaveHoltz iteration applied to elastic problems with Dirichlet and/or free surface boundary conditions. In Section 4.3 we outline the numerical methods used to solve the elastic wave equation and present new results on time-stepping and Krylov acceleration. Numerical examples are presented in Section 4.4. Finally, we summarize and conclude in Section 4.5.

4.1 Governing Equations

4.1.1 The Time Harmonic Elastic Wave Equation

For a linear isentropic elastic media the frequency domain equation is

$$\rho\omega^2\mathbf{v} + \nabla\cdot\mathcal{T}(\mathbf{v}) = \mathbf{f}(\mathbf{x}), \quad \mathbf{x}\in\Omega. \tag{4.2}$$

For notational convenience we will refer to this as the elastic Helmholtz or, when there is no ambiguity, simply the Helmholtz equation. This equation can be obtained by making the ansatz $\mathbf{u}(\mathbf{x},t) = e^{i\omega t}\mathbf{v}(\mathbf{x})$ and inserting it into the elastic wave equation (discussed below). We note that, in general, the Helmholtz solution $\mathbf{v}$ is complex-valued. However, for boundary conditions that conserve the energy (such as Dirichlet and conditions on the normal stress) the corresponding elastic wave equation solution $\mathbf{v}$ becomes real-valued. For real-valued solutions, the ansatz then simplifies to $\mathbf{u}(\mathbf{x},t) = \cos(\omega t)\mathbf{v}(\mathbf{x})$. The El WaveHoltz method can be used to find the solution $\mathbf{v}$ in both cases, but as we exclusively consider the energy conserving case here we primarily describe the method for that case.

4.1.2 The Elastic Wave Equation

The linear elastic wave equation in an isentropic material described by the density $\rho(\mathbf{x},t)$, the Lamé parameters $\mu(\mathbf{x}) > 0$ and $\lambda(\mathbf{x}) > 0$, and with a time harmonic forcing takes the form

$$\rho\mathbf{u}_{tt} = \nabla\cdot\mathcal{T}(\mathbf{u}) - \cos(\omega t)\mathbf{f}(\mathbf{x}), \quad \mathbf{x}\in\Omega, \quad 0\le t\le T. \tag{4.3}$$

Here $\mathbf{u} = (u(\mathbf{x},t), v(\mathbf{x},t), w(\mathbf{x},t))$ is the displacement vector, $\mathbf{x} = (x,y,z)^T$ is the Cartesian coordinate and t is time. The stress tensor $\mathcal{T}(\mathbf{u})$ can be decomposed into

$$\mathcal{T}(\mathbf{u}) = \lambda(\nabla\cdot\mathbf{u})I + 2\mu\mathcal{D}(\mathbf{u}), \tag{4.4}$$

where $\mathcal{D}(\mathbf{u})$ is the symmetric part of the displacement gradient

$$\mathcal{D}(\mathbf{u}) = \frac{1}{2}\begin{pmatrix} 2u_x & u_y+v_x & u_z+w_x \\ u_y+v_x & 2v_y & v_z+w_y \\ u_z+w_x & v_z+w_y & 2w_z \end{pmatrix}. \tag{4.5}$$

The equation (4.3) is closed by boundary conditions specifying the displacement

$$\mathbf{u}(\mathbf{x},t) = \cos(\omega t)\mathbf{g}(\mathbf{x}), \quad x\in\partial\Omega_{\mathrm{D}}, \tag{4.6}$$

or the normal stress

$$\mathcal{T}(\mathbf{u})\mathbf{n} = \cos(\omega t)\mathbf{h}(\mathbf{x}), \quad x\in\partial\Omega_{\mathrm{S}}, \tag{4.7}$$

along with initial conditions

$$\mathbf{u}(\mathbf{x},0)=\mathbf{u}_0(\mathbf{x}),\quad \frac{\partial \mathbf{u}(\mathbf{x},0)}{\partial t}=\mathbf{u}_1(\mathbf{x}). \tag{4.8}$$

Multiplying (4.3) by $\mathbf{u}^T$, integrating over Ω and invoking the divergence theorem yields the energy estimate

$$\frac{1}{2}\frac{d}{dt}\left(\|\sqrt{(}\rho)\mathbf{u}_t\|^2+\int_\Omega \lambda(\nabla\cdot\mathbf{u})I+2\mu(\mathcal{D}:\mathcal{D})d\mathbf{x}\right)=-\int_\Omega \cos(\omega t)\mathbf{u}^T\mathbf{f}(\mathbf{x})d\mathbf{x}+\int_{\partial\Omega}\mathbf{u}_t^T\mathcal{T}(\mathbf{u})\mathbf{n}\,dS. \tag{4.9}$$

Here $\mathbf{n}$ is the outward unit normal and the notation $(\mathcal{A}:\mathcal{B})=\sum_{i=1}^d\sum_{j=1}^d a_{i,j}b_{i,j}$ is the standard tensor contraction over two indices.

Thus, when there is no forcing, $\mathbf{f}(\mathbf{x})=0$, the energy is conserved in time as long as $\mathbf{u}_t^T\mathcal{T}(\mathbf{u})\mathbf{n}=0$ on the boundary $\partial\Omega$. The condition $\mathcal{T}(\mathbf{u})\mathbf{n}=0$ indicates that the boundary is stress free or free of traction. The Dirichlet condition on the velocity $\mathbf{u}_t=0$ also holds if the displacement vanishes for all time on the boundary, i.e. $\mathbf{u}=0$.

Note that if the initial data, $\mathbf{u}_0(\mathbf{x})$, gives rise to a solution of the form $\mathbf{u}(\mathbf{x},t)=\cos(\omega t)\mathbf{v}(\mathbf{x})$ then that solution coincides with the elastic Helmholtz solution to (4.2).

Remark 1. *In the rest of this chapter, unless otherwise noted, we will assume that the equations have been non-dimensionalized and that* $\rho=1$.

4.2 The El WaveHoltz Iteration

The El WaveHoltz iteration is a direct generalization of the WaveHoltz iteration introduced and and analyzed in [14]. Precisely, if we consider the energy conserving case, applying the Wave-Holtz operator component wise to the initial displacement vector $\mathbf{u}_0$ defines the El WaveHoltz operator

$$\Pi\mathbf{u}_0=\frac{2}{T}\int_0^T\left(\cos(\omega t)-\frac{1}{4}\right)\mathbf{u}(x,t)\;dt. \tag{4.10}$$

Here $T=\frac{2\pi}{\omega}$ and $\mathbf{u}(\mathbf{x},t)$ is the solution to (4.3) with the initial data $\mathbf{u}_0$ (recall that for the energy conserving case we always have $\mathbf{u}_1=\frac{\mathbf{u}(\mathbf{x},0)}{\partial t}=0$).

As the analysis of this operator is the same as that for the scalar operator analyzed in [14], we will not repeat the analysis in detail here. Instead, we now highlight its most important properties. The first thing to note is that if $\mathbf{u}(\mathbf{x},t) = \cos(\omega t)\mathbf{v}(\mathbf{x})$ (and thus $\mathbf{u}_0(\mathbf{x}) = \mathbf{v}(\mathbf{x})$), then the integral in (4.10) can trivially be evaluated

$$\Pi\mathbf{v}(x) = \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right)\cos(\omega t)\mathbf{v}(x)\ dt = \mathbf{v}(x), \tag{4.11}$$

showing that the elastic Helmholtz solution is a fixed point of the operator. Further, if we let $(\lambda_j^2, \boldsymbol{\phi}_j)$ be the eigendecomposition satisfying $\lambda^2\boldsymbol{\phi}_j = \nabla\cdot\mathcal{T}(\boldsymbol{\phi}_j)$, then for a general initial displacement the solution will be on the form $\sum_{j=0}^{\infty} d_j\cos(\lambda_j t)\boldsymbol{\phi}_j$. Let

$$\beta(\lambda) \equiv \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right)\cos(\lambda t)dt.$$

Then as in [14] we can define the operator $\mathcal{S}$ as

$$\mathcal{S}\sum_{j=0}^{\infty} d_j\boldsymbol{\phi}_j \equiv \sum_{j=0}^{\infty}\beta(\lambda_j)d_j\boldsymbol{\phi}_j,$$

which gives the filtered solution to the elastic wave equation when $\mathbf{f} = 0$. If $\omega \neq \lambda$ then the spectral radius $\max|\beta| < 1$ (see Lemma 2.1 in [14]) so the iteration will converge. Since the operator is linear, we may find the fixed point (or equivalently the elastic Helmholtz solution) by solving the equation $(\mathcal{I} - \mathcal{S})\mathbf{v} \equiv \mathcal{A}\mathbf{v} = \mathbf{b} \equiv \Pi\mathbf{0}$. As is the case for the scalar Helmholtz equation, the eigenvalues of $\mathcal{A}$ lie in $(0, 3/2)$ and the condition number scales with the frequency as $\text{cond}(\mathcal{A}) \sim \omega^{2d}$ in d dimensions.

We emphasize that here $\mathcal{A}$ is a self-adjoint, positive definite and bounded operator. Thus once $\mathcal{A}$ is discretized it will be possible to apply the conjugate gradient method. Moreover, as the condition number *does not* depend on the discretization size, the number of iterations are not expected to increase as the solution becomes more accurate due to grid refinement. We also note that since $\text{cond}(\mathcal{A}) \sim \omega^{2d}$ the conjugate gradient method is expected to converge to a fixed tolerance in ω^d iterations.

Finally, as mentioned above it is possible to define the iteration as the integral over multiple periods in order to reduce the number of all-to-all communication in the Krylov iteration. For example, if the number of periods is K then we can define the filtering as

$$\Pi_K \mathbf{u}_0 = \frac{2}{KT} \int_0^{KT} \left(\cos(\omega t) - \frac{1}{4} \right) \mathbf{u} \; dt, \; T = \frac{2\pi}{\omega}. \tag{4.12}$$

Remark 2. *For general boundary conditions (e.g. non-reflecting or impedance),* $\frac{\boldsymbol{u}(\boldsymbol{x},0)}{\partial t} = \boldsymbol{u}_1(\boldsymbol{x})$ *will not be zero and we must seek the initial data* $\boldsymbol{u}_0$ *and* $\boldsymbol{u}_1$ *simultaneously. The El WaveHoltz operator then is*

$$\Pi \begin{bmatrix} \boldsymbol{u}_0 \\ \\ \boldsymbol{u}_1 \end{bmatrix} = \frac{2}{T} \int_0^{T} \left(\cos(\omega t) - \frac{1}{4} \right) \begin{bmatrix} \boldsymbol{u} \\ \\ \boldsymbol{u}_t \end{bmatrix} \; dt, \; T = \frac{2\pi}{\omega}.$$

This operator is more difficult to analyze [51] but in practice the iteration converges much faster, typically in $\sim \omega$ *iterations independent of dimension.*

4.3 Numerical Methods and Discrete Analysis

An attractive feature of El WaveHoltz is that it can be used together with any convergent discretization of the elastic wave equation. Here we consider the conservative curvilinear finite difference method from [12] and the symmetric interior penalty discontinuous Galerkin method [35, 62]. We give a very brief description of these below methods and refer the reader to [12, 35] for details.

Although highly non-intrusive, the one additional discretizational detail necessitated by El WaveHoltz is how to discretize the integral in (4.10). As the integrand is periodic (once converged) we always use the trapezoidal rule.

4.3.1 El WaveHoltz by Finite Differences

To discretize the elastic wave equation (4.1.2) in a general non-Cartesian geometry we write (4.1.2) in a curvilinear coordinate system that conforms with the boundaries of the domain but that can be mapped back to the unit square (cube). Thus, we assume that there is a one-to-one

mapping

$$x = x(q,r), \quad y = y(q,r), \quad (q,r) \in [0,1]^2,$$

from the unit square to the domain of interest. Then the two dimensional version of (4.1.2) becomes

$$J\rho\frac{\partial^2 u}{\partial t^2} = \frac{\partial}{\partial q}\Big[Jq_x\big[\,(2\mu+\lambda)\,(q_x\partial_q + r_x\partial_r)\,u + \lambda\,(q_y\partial_q + r_y\partial_r)\,v\big] + Jq_y\big[\mu\,((q_x\partial_q + r_x\partial_r)\,v + (q_y\partial_q + r_y\partial_r)\,u)\,\big]\Big]$$
$$+\frac{\partial}{\partial r}\Big[Jr_x\big[\,(2\mu+\lambda)\,(q_x\partial_q + r_x\partial_r)\,u + \lambda\,(q_y\partial_q + r_y\partial_r)\,v\big] + Jr_y\big[\mu\,((q_x\partial_q + r_x\partial_r)\,v + (q_y\partial_q + r_y\partial_r)\,u)\,\big]\Big],$$

$$J\rho\frac{\partial^2 v}{\partial t^2} = \frac{\partial}{\partial q}\Big[Jq_x\big[\mu\,((q_x\partial_q + r_x\partial_r)\,v + (q_y\partial_q + r_y\partial_r)\,u)\,\big] + Jq_y\big[\,(2\mu+\lambda)\,(q_y\partial_q + r_y\partial_r)\,v + \lambda\,(q_x\partial_q + r_x\partial_r)\,u\big]\Big]$$
$$+\frac{\partial}{\partial r}\Big[Jr_x\big[\mu\,((q_x\partial_q + r_x\partial_r)\,v + (q_y\partial_q + r_y\partial_r)\,u)\,\big] + Jr_y\big[\,(2\mu+\lambda)\,(q_y\partial_q + r_y\partial_r)\,v + \lambda\,(q_x\partial_q + r_x\partial_r)\,u\big]\Big].$$

Here $J = x_q y_r - x_r y_q$ is the Jacobian of the mapping. Also note that we have considered the case without forcing for brevity.

We discretize the unit square $(q,r) \in [0,1]^2$ by a uniform grid on which we introduce real valued grid functions $[u_{i,j}(t), v_{i,j}(t)] = [u(q_i, r_j, t), v(q_i, r_j, t)]$. On this grid we apply the an energy stable discretization

$$\rho J\frac{\partial^2 u_h}{\partial t^2} = L^{(u)}(u_h, v_h), \qquad \rho J\frac{\partial^2 v_h}{\partial t^2} = L^{(v)}(u_h, v_h). \tag{4.13}$$

Here ρJ is a diagonal matrix containing the metric information and u_h, v_h are vectors containing all the grid function values. The (lengthy) exact definitions of $L^{(u)}(u_h, v_h), L^{(v)}(u_h, v_h)$ can be found in [12].

Suppose we are to impose a free surface boundary condition at $q = 0$. We then use a modified stencil for which the method is stable in a modified inner product. Let w_h and u_h be real valued grid functions and $(w_h, u_h)_h$ be the discrete inner product

$$(w_h, u_h)_h = h_q h_r \sum_{j=1}^{N_r}\left(\frac{1}{2}w_{1,j}u_{1,j} + \sum_{i=2}^{N_q} w_{i,j}u_{i,j}\right),$$

with corresponding norm $\|w_h\|_h^2 = (w_h, w_h)_h$. In this inner product, the discretization (of the PDE and boundary conditions) is self adjoint. That is, for all real-valued grid functions (u^*, v^*), $(u^\dagger, v^\dagger)$

satisfying the discrete boundary conditions, we have

$$(u^*, L^{(u)}(u^\dagger, v^\dagger))_h + (v^*, L^{(v)}(u^\dagger, v^\dagger))_h = (u^\dagger, L^{(u)}(u^*, v^*))_h + (v^\dagger, L^{(v)}(u^*, v^*))_h. \tag{4.14}$$

To discretize the equations in time we either use the standard second order accurate centered differences, or one of the time-corrected schemes discussed below. For the standard second order accurate centered difference approximation in time, the fully discrete equations take the form

$$\begin{aligned}(\rho J)(u_h^{n+1} - 2u_h^n + u_h^{n-1}) &= \Delta t^2 L^{(u)}(u_h^n, v_h^n),\\ (\rho J)(v_h^{n+1} - 2v_h^n + v_h^{n-1}) &= \Delta t^2 L^{(v)}(u_h^n, v_h^n).\end{aligned} \tag{4.15}$$

Then, if $(u, v)_{\rho J}$ is the weighted inner product defined by $(f, (\rho J)^{-1} g)_{\rho J} = (f, g)_h$, and $C_e(t^{n+1})$ is the discrete energy

$$C_e(t^{n+1}) = \|D_+^t u^n\|_{\rho J}^2 + \|D_+^t v^n\|_{\rho J}^2 - (u^{n+1}, (\rho J)^{-1} L^{(u)}(u^n, v^n))_{\rho J} - (v^{n+1}, (\rho J)^{-1} L^{(v)}(u^n, v^n))_{\rho J}, \tag{4.16}$$

one can show that this discrete energy is conserved [12].

Note that (4.15) is slightly non-symmetric and needs to be diagonally scaled to become symmetric. Here we scale by 2 along sides with free surface boundary conditions, and by 4 in corners where free surfaces meet. Incorporating this scaling through the multiplication by a scaling matrix Λ, the method can be formally written as

$$M(\mathbf{u}_h^{n+1} - 2\mathbf{u}_h^n + \mathbf{u}_h^{n-1}) = \Delta t^2 L_h \mathbf{u}_h. \tag{4.17}$$

Here $M = \mathrm{diag}(\Lambda\rho J, \Lambda\rho J)$ and $L_h = \mathrm{diag}(\Lambda L^{(u)}, \Lambda L^{(v)})$ are symmetric and M is diagonal. However, as $M^{-1}L_h$ is not in general symmetric, the iteration (4.10) will produce a symmetrizable but not symmetric operator. We will show below that this necessitates a minor modification of the conjugate gradient algorithm when used together with the iteration (4.10).

Remark 3. *Here we only consider domains that can be discretized by a single logically Cartesian grid, but note that by using the overset grid version of the method, [7], more complex geometry could be handled.*

4.3.2 El WaveHoltz by Symmetric Interior Penalty Discontinuous Galerkin Method

As an alternative to the finite difference method outlined above, we will also consider the Symmetric Interior Penalty Discontinuous Galerkin (SIPDG) method [35, 62]. Let Ω_h be a finite element partition of the computational domain Ω, with Γ_h the set of all faces. Then (4.3) can be reformulated into the interior-penalty weak formulation: Find $\mathbf{u}_h \in (0,T) \times V_h$ such that

$$\sum_{E\in\Omega_h} (\rho \frac{d^2\mathbf{u}_h}{dt^2}, \mathbf{v})_E + \sum_{E\in\Omega_h} B_E(\mathbf{u}_h, \mathbf{v}) + \sum_{\gamma\in\Gamma_h} J_\gamma(\mathbf{u}_h, \mathbf{v}; S, R) = -\cos(\omega t) \sum_{E\in\Omega_h} (\mathbf{f}, \mathbf{v})_E, \tag{4.18}$$

for all $\mathbf{v} \in V_h$. Here

$$(\mathbf{u}, \mathbf{v})_E = \int_E \mathbf{u} \cdot \mathbf{v}\, dE,$$

$$B_E(\mathbf{u}, \mathbf{v}) = \int_E \left[\lambda(\nabla\cdot\mathbf{u})(\nabla\cdot\mathbf{v}) + \mu(\nabla\mathbf{u} + \nabla\mathbf{u}^T) : \nabla\mathbf{v}\right]\, dE,$$

$$J_\gamma(\mathbf{u}, \mathbf{v}; S, R) = -\int_\gamma \{\mathcal{T}(\mathbf{u})\mathbf{n}\}\cdot[\mathbf{v}]\, d\gamma + S\int_\gamma \{\mathcal{T}(\mathbf{v})\mathbf{n}\}\cdot[\mathbf{u}]\, d\gamma + R\int_\gamma \{\lambda+2\mu\}[\mathbf{u}]\cdot[\mathbf{v}]\, d\gamma,$$

where $\{\cdot\}$ and $[\cdot]$ denote the average and jump of a function, respectively. The parameter R is the penalty and S determines the particular flavor of IPDG. We thus set $S = -1$, corresponding to the Symmetric IPDG [62]. In this case, J_γ is symmetric with respect to $\mathbf{u}_h$ and $\mathbf{v}$ so that together with the symmetry of B_E we have that the stiffness matrix is symmetric. Thus SIPDG provides a symmetric discretization of the elastic wave equation, which will allow the use of conjugate gradient to accelerate convergence of the El WaveHoltz iteration.

Our solver is implemented in MFEM[1] [4] and is essentially a direct extension of example 17 to the time domain. Depending on the mesh, our choice of finite element space V_h is typically one of two broken spaces. We choose either $\mathcal{P}^p(E)$, the space of polynomials of total degree at most p on triangles, or $\mathcal{Q}^p(E)$, the space of polynomials of at most degree p on quadrilaterals. Unless otherwise noted, for the penalty parameter we make the choice $R = (p+1)(p+2)$.

With the standard second order explicit time discretization, the matrix form of (4.18) becomes

$$M_\rho(\mathbf{u}_h^{n+1} - 2\mathbf{u}_h^n + \mathbf{u}_h^{n-1}) = \Delta t^2 \left[L_h\mathbf{u}_h^n - \cos(\omega t^n)\hat{\mathbf{f}}\right].$$

[1] `www.mfem.org`

As for the finite difference method, $M_\rho^{-1} L_h$ is not (in general) symmetric and this will necessitate a minor modification of the conjugate gradient algorithm when this scheme is used together with the iteration (4.10).

For this explicit time-stepping and the error corrected time-stepping discussed below, we use the CFL condition from [10]

$$\Delta t < \frac{\text{CFL} \cdot h_{\min}}{(p+\frac{3}{2})^2 \sqrt{\frac{2\mu+\lambda}{\rho}}}, \tag{4.19}$$

where $h_{\min}$ is the smallest diameter of the elements and CFL depends on the time-stepper. For the second order centered scheme, we typically choose CFL $\sim$ 0.4–0.8.

4.3.3 Explicit Time-Corrected Scheme

If the elastic Helmholtz equation equation (4.2) is discretized directly, the solution satisfies the equation (in this section we take $\rho = 1$ and for notational clarity we suppress the subscript h)

$$\omega^2 \mathbf{v} + L_h \mathbf{v} = \mathbf{f}(\mathbf{x}). \tag{4.20}$$

As we show in [14, 51], when the elastic wave equation is time marched with e.g. the second order method

$$\mathbf{u}^{n+1} - 2\mathbf{u}^n + \mathbf{u}^{n-1} = \Delta t^2 \left[L_h \mathbf{u}^n - \cos(\omega t_n)\, \mathbf{f} \right], \tag{4.21}$$

started with the initial data

$$\mathbf{u}^0 = \mathbf{u}_0, \qquad \mathbf{u}^{-1} = \mathbf{u}_0 - \frac{\Delta t^2}{2} L_h \left(\mathbf{u}_0 + \mathbf{f} \right).$$

Once El WaveHoltz has converged the initial data $\mathbf{u}_0$ satisfies the elastic Helmholtz equation with a modified frequency

$$\tilde{\omega}^2 \mathbf{u}_0(\mathbf{x}) + L_h \mathbf{u}_0(\mathbf{x}) = \mathbf{f}(\mathbf{x}), \quad \tilde{\omega} = \frac{2\sin(\Delta t \omega/2)}{\Delta t}. \tag{4.22}$$

For this second order time discretization the difference between the final converged $\mathbf{u}_0$ and $\mathbf{v}$ is $\mathcal{O}(\Delta t^2)$. Thus if a high order accurate spatial discretization is used, time discretization errors will

limit the accuracy of the El WaveHoltz solution. To reduce this error, a time discretization which is at least as accurate as the spatial discretization can be used. It is also possible, however, to use the technique proposed by Stolk in [101] to modify the second order time-stepping method and eliminate the error altogether. The corrected scheme in [101], introduced as a time domain preconditioner, is the straightforward modification

$$\mathbf{u}^{n+1} - 2\mathbf{u}^n + \mathbf{u}^{n-1} = \frac{\tilde{\omega}^2}{\omega^2}\Delta t^2 \left[L_h \mathbf{u}^n - \cos(\omega t_n)\,\mathbf{f}\right]. \tag{4.23}$$

As [101] solves the equations in the frequency domain no initial data is needed. Here, as we work in the time domain, we must also modify the computation of $\mathbf{u}^0$ accordingly:

$$\mathbf{u}^0 = \mathbf{u}_0, \qquad \mathbf{u}^{-1} = \mathbf{u}_0 - \frac{\tilde{\omega}^2}{\omega^2}\frac{\Delta t^2}{2} L_h \left(\mathbf{u}_0 + \mathbf{f}\right).$$

4.3.4 Implicit Time-Corrected Scheme

For a DG discretization, the use of an explicit time-stepping scheme for the elastic wave equation requires a CFL condition that shrinks as $\mathcal{O}(p^{-2})$ where p is the polynomial order within an element. For meshes with geometrical stiffness and DG discretizations of high order, it is then particularly desirable to consider the use of an implicit scheme to circumvent a potentially restrictive time-step size demanded by an explicit scheme.

To that end, consider the semi-discrete system

$$\rho \mathbf{u}_{tt} = L_h \mathbf{u} - \cos(\omega t)\mathbf{f}(\mathbf{x}), \quad \mathbf{x} \in \Omega,\, t \geq 0, \tag{4.24}$$

where L_h is a symmetric, positive definite approximation to the continuous operator $\nabla \cdot \mathcal{T}$ including boundary conditions. The values $\nabla \cdot \mathcal{T}(\mathbf{u})$ are then approximated by $L_h \mathbf{u}$. We assume L_h has the eigenmodes $(\lambda_j^2, \boldsymbol{\phi}_j)$, such that $L_h \boldsymbol{\phi}_j = \lambda_j^2 \boldsymbol{\phi}_j$ for $j = 1, \ldots, N$, where all λ_j are strictly positive and ordered as $0 \leq \lambda_1 \leq \cdots \leq \lambda_N$.

We let the discrete Helmholtz solution $\mathbf{v}$ be given by

$$-L_h \mathbf{v} + \omega^2 \mathbf{v} = \mathbf{f}.$$

The numerical approximation of the iteration operator is denoted Π_h, and it is implemented as follows. Given a grid function $\mathbf{u} \in \mathbb{R}^N$, we use the following implicit time-stepping scheme to solve the elastic wave equation as

$$\frac{\mathbf{u}^{n+1} - \alpha\mathbf{u}^n + \mathbf{u}^{n-1}}{\Delta t^2} = \frac{1}{2}L_h(\mathbf{u}^{n+1} + \mathbf{u}^{n-1}) - \mathbf{f}\cos(\omega t_n)\cos(\omega\Delta t), \tag{4.25}$$

where

$$\alpha = \cos(\omega\Delta t)(2 + \omega^2\Delta t^2) \approx 2 - \frac{5(\omega\Delta t)^4}{12} + \mathcal{O}(\Delta t^6). \tag{4.26}$$

For the stability of the method it is necessary to have $|\alpha| < 2$. This choice of the time-step corresponds to a (mild) requirement of at least five time-steps per iteration (See details in Appendix .10).

With a time-step $\Delta t = T/k$ for some integer k, the scheme (4.25) is completed by initial data

$$\mathbf{u}^0 = \mathbf{u}_0, \qquad \mathbf{u}^{-1} = \left(I - \frac{\Delta t^2}{2}L_h\right)^{-1}\left(\frac{\alpha}{2}\mathbf{u}_0 - \frac{\Delta t^2}{2}\cos(\omega\Delta t)\mathbf{f}\right).$$

The trapezoidal rule is then used to compute $\Pi_h\mathbf{u}$,

$$\Pi_h\mathbf{u} = \frac{2\Delta t}{T}\sum_{n=0}^{M}\eta_n\left(\cos(\omega t_n) - \frac{1}{4}\right)\mathbf{u}^n, \qquad \eta_n = \begin{cases} \frac{1}{2}, & n = 0 \text{ or } n = M, \\ 1, & 0 < n < M. \end{cases} \tag{4.27}$$

Define the discrete filter transfer function by

$$\beta_h(\lambda) = \frac{2\Delta t}{T}\sum_{n=0}^{M}\eta_n\cos(\lambda t_n)\left(\cos(\omega t_n) - \frac{1}{4}\right),$$

In Appendix .11 we motivate the following Conjecture (we believe this can be proved but at the time of writing we have not yet done so.)

Conjecture 1. *Let $\Delta t = T/k$ for some integer k with $T = 2\pi/\omega$. The discrete filter transfer function β_h satisfies*

$$\begin{cases} |\beta_h(\lambda)| \geq 1, & \lambda \in [\omega(1 - 0.022\cdot\Delta t^2), \omega], \\ |\beta_h(\lambda)| < 1, & \textit{otherwise.} \end{cases}$$

Under the assumption that Conjecture 1 holds we may then prove the following theorem.

Theorem 1. *Suppose there are no resonances, such that* $\delta_h = \min_j |\lambda_j - \omega|/\omega > 0$. *Moreover, assume that* Δt *satisfies the stability and accuracy requirements*

$$\frac{|\alpha|}{2} = |\cos(\omega\Delta t)(1+\omega^2\Delta t^2/2)| < 1, \qquad \Delta t \le \frac{\cos(2\pi/5)\omega^2\delta_h}{0.044\cdot(1+2(\pi\lambda_N/5)^2)}. \tag{4.28}$$

Further assume that the properties of the discrete filter transfer function in Conjecture 1 hold. Then the fixed point iteration $\boldsymbol{v}^{(k+1)} = \Pi_h \boldsymbol{v}^{(k)}$ *with* $\boldsymbol{v}^{(0)} = 0$ *converges to* $\boldsymbol{v}$ *which is a solution to the discretized Helmholtz equation*

$$-L_h\boldsymbol{v} + \omega^2\boldsymbol{v} = \boldsymbol{f}.$$

Proof. We expand all functions in eigenmodes of L_h,

$$\mathbf{u}^n = \sum_{j=1}^N \hat{u}_j^n \boldsymbol{\phi}_j, \qquad \mathbf{f} = \sum_{j=1}^N \hat{f}_j\boldsymbol{\phi}_j, \qquad \mathbf{v} = \sum_{j=1}^N \hat{v}_j\boldsymbol{\phi}_j.$$

Then the Helmholtz eigenmodes of $\mathbf{v}$ satisfy

$$\hat{v}_j = \frac{\hat{f}_j}{\omega^2 - \lambda_j^2}.$$

The wave solution eigenmodes are given by the difference equation

$$\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\hat{u}_j^{n+1} - \alpha\hat{u}_j^n + \left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\hat{u}_j^{n-1} = -\Delta t^2\hat{f}_j\cos(\omega t_n)\cos(\omega\Delta t), \tag{4.29}$$

with initial data

$$\hat{u}_j^0 = \hat{u}_{0,j}, \qquad \hat{u}_j^{-1} = \left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)^{-1}\left(\frac{\alpha}{2}\hat{u}_{1,j} - \frac{1}{2}\Delta t^2\cos(\omega\Delta t)\hat{f}_j\right).$$

By (4.28),

$$\left|\alpha\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)^{-1}\right| = \left|2\cos(\omega\Delta t)\frac{2+\Delta t^2\omega^2}{2+\Delta t^2\lambda_j^2}\right| < 2,$$

so that the characteristic polynomial for the equation, $r^2 - \alpha(1+\Delta t^2\lambda_j^2/2)^{-1}r + 1$, has two roots on the boundary of the unit circle. The solution is therefore stable and is given by (with a verification in Appendix .9)

$$\hat{u}_j^n = (\hat{u}_{0,j} - \hat{v}_j)\cos(\tilde{\lambda}_j t_n) + \hat{v}_j\cos(\omega t_n), \tag{4.30}$$

where $\tilde{\lambda}_j$ is defined by the relation

$$\cos(\tilde{\lambda}_j \Delta t) = \frac{\alpha}{2}\left(1 + \frac{\Delta t^2}{2}\lambda_j^2\right)^{-1} = \cos(\omega \Delta t)\left(1 + \frac{\omega^2 \Delta t^2}{2}\right)\left(1 + \frac{\Delta t^2}{2}\lambda_j^2\right)^{-1}. \tag{4.31}$$

Now, let

$$\Pi_h \mathbf{u}_0 = \sum_{j=1}^{\infty} \bar{u}_j \boldsymbol{\phi}_j.$$

Then the numerical integration gives

$$\bar{u}_j = \frac{2\Delta t}{T}\sum_{n=0}^{M} \eta_n \left(\cos(\omega t_n) - \frac{1}{4}\right)\Big((\hat{u}_{0,j} - \hat{v}_j)\cos(\tilde{\lambda}_j t_n) + \hat{v}_j \cos(\omega t_n)\Big) = (\hat{u}_{0,j} - \hat{v}_j)\beta_h(\tilde{\lambda}_j) + \hat{v}_j \beta_h(\omega)$$

$$= \hat{u}_{0,j}\beta_h(\tilde{\lambda}_j) + (1 - \beta_h(\tilde{\lambda}_j))\hat{v}_j.$$

Here we used the fact that the trapezoidal rule is exact, and equal to one, when $\lambda = \omega$. (Recall that for periodic functions the trapezoidal rule is exact for all pure trigonometric functions of order less than the number of grid points.)

For the time-step restriction (4.28), we have that $|\tilde{\lambda}_j - \omega| > 0.022 \cdot \Delta t^2$ so that the bound $|\beta_h(\tilde{\lambda}_j)| \leq \rho_h < 1$ in the conjecture is uniform for all j. It follows that $\mathbf{v}^{(k)} \to \mathbf{v}$. This concludes the proof of the theorem. □

Remark 4. *We remark that it is also possible to remove the time discretization error by modifying the weights in the trapezoidal rule as in [93]*

$$\frac{2\Delta t}{T}\sum_{n=0}^{N_t} \frac{\cos(\omega t_n)}{\cos(\frac{2\sin(\Delta t \omega/2)}{\Delta t} t_n)}\left(\cos(\omega t_n) - \frac{1}{4}\right)\boldsymbol{u}^n. \tag{4.32}$$

It should however be noted that there is a risk that the denominator in this expression can become arbitrarily close to zero unless care is taken.

4.3.5 Krylov Solution of the El WaveHoltz Iteration

Let Π_h be the matrix corresponding to a discretization of the El WaveHoltz method using either the finite difference or the SIPDG method. Then the iteration is (in this section a superscript

i denotes iteration and a superscript n denotes time-step)

$$\mathbf{u}_h^0 = \Pi_h \mathbf{0},$$

$$\mathbf{u}_h^{i+1} = \Pi_h \mathbf{u}_h^i, \quad i = 0, 1, \ldots$$

The solution to this fixed point iteration can also be found by solving

$$(I - \Pi_h)\mathbf{u}_h = \mathbf{b} \equiv \Pi_h \mathbf{0}, \tag{4.33}$$

where the action of the matrix $(I - \Pi_h)$ requires (4.3) with $\mathbf{f} = 0$ to be solved for one period, $T = 2\pi/\omega$, and the right hand side is pre computed by solving (4.3) with the $\mathbf{f}$ at hand.

Let $Q = I - \Pi_h$. We know from [14] that the eigenvalues of Q are in the interval $(0, 3/2)$ so that Q is positive definite. We note that the methods for the elastic wave equation we consider here produce solutions $\{\mathbf{u}^1, \mathbf{u}^2, \ldots, \mathbf{u}^{N_\mathrm{t}}\}$ at time instances $0, \Delta t, 2\Delta t, \ldots$, according to the recursion

$$\mathbf{u}^0 = a_0 \mathbf{u} + a_1 M^{-1} S \mathbf{u}^n, \tag{4.34}$$

$$\mathbf{u}^1 = \mathbf{u} \tag{4.35}$$

$$\mathbf{u}^{n+1} = \kappa \mathbf{u}^n - \mathbf{u}^{n-1} + \gamma M^{-1} S \mathbf{u}^n. \tag{4.36}$$

It follows that the matrix

$$Q\mathbf{u} = \sum_{n=1}^{N_\mathrm{t}} \alpha_i \mathbf{u}^n, \tag{4.37}$$

will not be symmetric even if M and S are.

However, as the operator Q can be expressed as a polynomial P_Q of degree $N_\mathrm{t} - 1$ in $M^{-1}S$, we have that MQ is symmetric. Thus rather than applying the conjugate gradient method to (4.33), we instead solve

$$M(I - \Pi_h)\mathbf{u}_h = M\mathbf{b}. \tag{4.38}$$

We note that the main cost of applying the matrix Q is in computing $\Pi_h \mathbf{u}_h$. Since the matrix M is diagonal for the finite difference method and block-diagonal for SIPDG, the difference in cost between applying (4.38) over (4.33) is negligible compared to the advantage of not having to store a Krylov subspace when using the conjugate gradient or conjugate residual method.

Table 4.1: L_1, L_2 and L_∞ errors of the computed solution with corresponding estimated rates of convergence.

n	L_1 error	Convergence	L_2 error	Convergence	L_∞ error	Convergence
20	3.86(-3)	-	3.86(-3)	-	3.86(-3)	-
40	9.21(-4)	2.06	9.21(-4)	2.06	9.21(-4)	2.06
80	2.25(-4)	2.03	2.25(-4)	2.03	2.25(-4)	2.03
160	5.55(-5)	2.02	5.55e(-5)	2.02	5.55(-5)	2.02

Remark 5. *In some of the experiments below we use conjugate residual rather than conjugate gradient. The reason for this is that it has the property that the residual is non-increasing, which we have found gives a predictable and robust iteration count when doing parameter sweeps over ω. When conjugate gradient is used we sometimes observe that the we get "lucky" and converge in very few iterations for a few frequencies. When considering practical applications it is of course good to have such luck, but as we are trying to present the average behavior of our method here we prefer conjugate residual.*

4.4 Numerical Experiments

In this section we present numerical experiments that demonstrate the properties of the method. We start with numerical experiments that demonstrate the spatial accuracy with and without the time-stepping correction for the finite difference and the discontinuous Galerkin method.

4.4.1 Accuracy of the Finite Difference Method

We consider solving the elastic Helmholtz equation with Lamé parameters $\lambda = \mu = 1.0$, where the forcing function is chosen so that the displacements are given by

$$u = v = 16^2 x^2 (x-1)^2 y^2 (y-1)^2. \tag{4.39}$$

We take the frequency to be $\omega = 1.0$ and enforce Dirichlet boundary conditions on the boundary of the unit square $(x, y) \in [0, 1] \times [0, 1]$. To verify accuracy, we set the tolerance to 10^{-15} in the conjugate residual method as the stopping criteria and compute the error in u to the exact solution. We use the finite difference method together with the standard explicit second order time-stepping scheme, and verify the convergence of the method by grid refinement. To that end, we choose the coarsest grid to have $n = 20$ points along each direction and refine by a factor of two up to $n = 160$ points per direction. In Table 4.1 we estimate the rate of convergence and observe second order convergence, as expected.

4.4.2 Verification of Corrected Time-Steppers

We consider solving the elastic Helmholtz equation with $\lambda = \mu = 1.0$ and choose the forcing so that the exact solution is the same as (4.39). We take the frequency $\omega = 1$ and enforce homogeneous Dirichlet conditions on the boundary of the square $(x, y) \in [0, 1] \times [0, 1]$. As the solution is a fourth order polynomial, choosing $p = 4$ should ensure that the solution to the discrete elastic Helmholtz equation is precisely (4.39). We use the conjugate gradient accelerated version of El WaveHoltz with the corrected second order centered time-stepping scheme presented in Section 4.3.3.

We partition the domain into four quadrilaterals of equal side length $h = 0.5$, set the absolute conjugate gradient residual tolerance to 10^{-15}, and consider the error as the time-step size is decreased.

We see from Figure 4.1 that the standard centered scheme leads to a discrete solution that converges to the true solution to second order. The modified scheme, however, maintains the same relative error to the true solution independent of time-step size which indicates that (aside from roundoff errors) the time-stepping errors have been removed. For the remaining numerical examples we use the modified time-stepping scheme to remove time discretization errors.

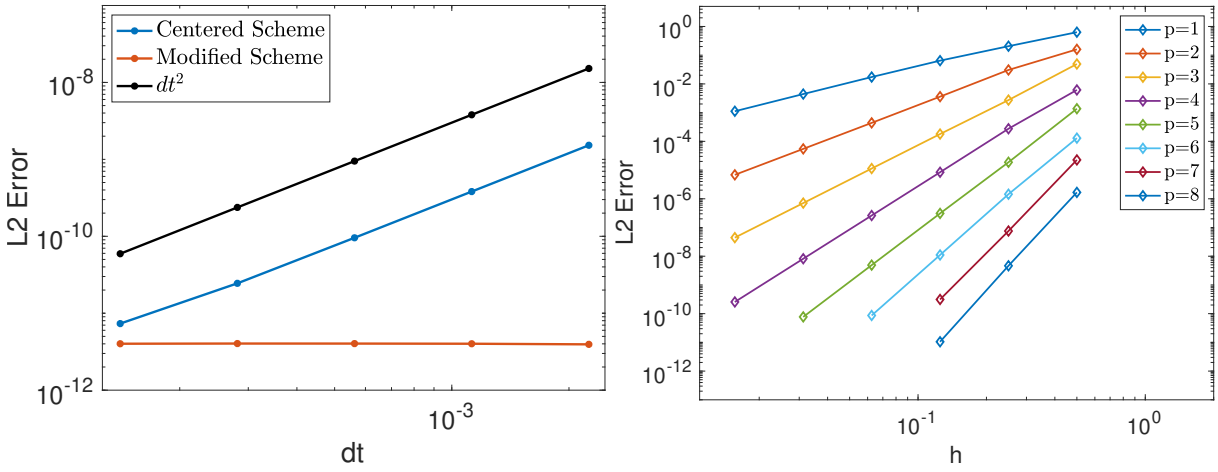


Figure 4.1: (Left) Convergence of the discrete WaveHoltz solution to the true solution of the discrete Helmholtz problem. (Right) Convergence of the discrete WaveHoltz solution to the true solution of the discrete Helmholtz problem for a manufactured solution.

4.4.3 Accuracy of the Symmetric Interior Penalty Discontinuous Galerkin Method

Next we verify the rates of convergence for our symmetric interior penalty DG solver and for non-homogeneous problems using an example taken from [10]. We consider the unit square $S = [0, 1]^2$ and impose Dirichlet conditions on the boundary. The boundary conditions and forcing are chosen so that the Helmholtz solution is

$$u(x, y) = \sin(k_x x + x_0) \sin(k_y y + y_0),$$
$$v(x, y) = -\sin(k_x x + x_0) \sin(k_y y + y_0),$$

where $k_x = 2.5\pi$, $k_x = 2\pi$, $x_0 = 5$, $y_0 = -10$. The mesh used is a uniform discretization of the unit square split into smaller squares of side-length $h = 1/2^n$ for $n = 1, \ldots, 6$.

Table 4.2: Estimated rates of convergence for the spatial discretization.

p	1	2	3	4	5	6	7	8
	1.84	2.94	4.00	4.94	6.01	6.86	8.07	8.64

We set $\omega = 1$ and choose the material parameters to be the constants $\mu = 1$, $\lambda = 2$. Here we use the modified time-stepping scheme of Section 4.3.3 with CFL $= 0.4$.

The errors are plotted in Figure 4.1 as a function of the grid size h. We additionally display estimated rates of convergence calculated using linear least squares in Table 4.2 from which it is clear that the WaveHoltz method converges with optimal rates with the error corrected time-stepper (which is formally only second order accurate in Δt).

4.4.4 Effects on Number of Iterations from Number of Periods and Accuracy

In this section we investigate the efficiency of the filter (4.12), defined over K periods, for various values of K. Let N_T be the number of time-steps for one period. Then we expect the reduction in the number of all-to-all communications to be KN_T when compared to a direct discretization of (4.1.1). Here we consider energy conserving boundary conditions, for which we can

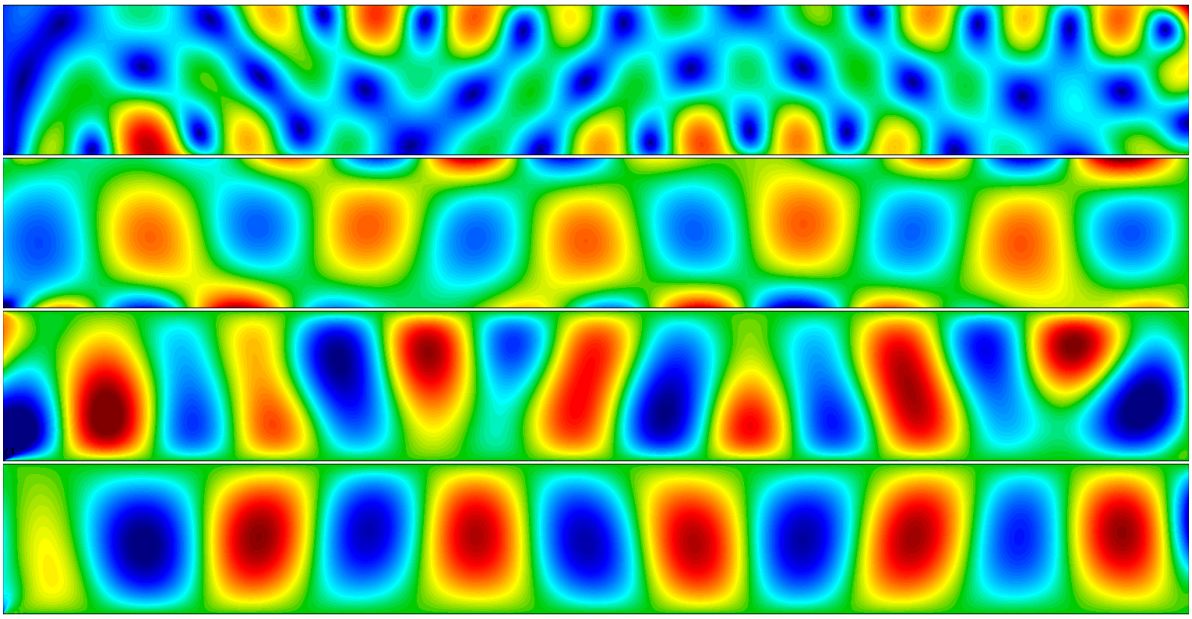


Figure 4.2: From top to bottom: displacement magnitude, σ_{xx}, σ_{xy} and σ_{yy}. The domain is $[0, 8] \times [0, 1]$ and the color scales are $[0, 8]$, $[-50, 50]$, $[-15, 15]$ and $[-40, 40]$ respectively.

Table 4.3: The table displays the number of iterations required and the efficiency of the longer times to reduce the relative residual by a factor 10^{-10} for the two cases (described in the text).

Periods	1	2	3	4	5	10
Case 1 (#iter)	124	69	51	43	39	28
Efficiency	1	0.90	0.81	0.72	0.64	0.44
Case 2 (#iter)	151	96	78	68	62	53
Efficiency	1	0.79	0.65	0.56	0.49	0.29

use the conjugate gradient method and avoid the need to store a Krylov subspace. We note that for problems with impedance or non-reflecting boundary conditions (or with lower order damping terms), El WaveHoltz will still result in a positive definite but non-symmetric system which can be solved e.g. with GMRES. In that case, we also expect that the size of the GMRES Krylov subspace will decrease by a factor KN_T compared to direct discretization, and by a factor K compared to using a single period for the filter procedure.

With these obvious advantages of filtering over K periods, it is natural to ask how the number of iterations are affected by the increased filter time. In this experiment we numerically investigate this. To do so, we use the corrected explicit version of the DG solver and consider the shaking of a bar of (unitless) length 8 and height 1. We impose free surface boundary conditions on the top, bottom, and right of the domain, and on the left we set the boundary conditions to be

$$u(0, y, t) = v(0, y, t) = \cos(\omega t).$$

The base computational mesh uses 8 square elements each with side length 1, which we uniformly refine by dividing each element in 4 parts for some number of refinements. We set $\lambda = 2$, $\mu = 1$, $\omega = 5.123$ and CFL $= 0.8$. We consider 2 cases: Case 1 uses $p = 5$ and refines the base grid 3 times, Case 2 uses $p = 15$ and refines one time. For both cases we use conjugate gradient and count the number of iterations it takes to reduce the relative residual by a factor 10^{-10}. The solution, along with the components of the stress tensor σ_{xx}, σ_{xy} and σ_{yy}, are displayed in Figure 4.2. The number of iteration for the two cases and the relative efficiency are tabulated in Table 4.3. As can be seen,

the efficiency is relatively high when the number of periods are small and can thus be deployed if the all-to-all communications (or the size of a GMRES Krylov space) becomes a limiting factor.

4.4.5 Iteration Count as a Function of Frequency for Rectangles and Annular Sectors

For energy conserving boundary conditions, the theoretical prediction (which is also observed experimentally) is that the number of iterations scales as ω^d in d-dimensions. In this and the next section, we study how the number of iterations depends on the frequency in two and three dimensions. In this section we additionally investigate the dependence of the number of iterations on frequency for different geometries. Here, we study these properties via three different computational domains: a rectangle, a quarter annulus, and a half annulus (all with a characteristic length of 5). We use the finite difference method together with the standard explicit second order time-stepping scheme. For each of the geometries we consider the set of frequencies $\omega = k + \sqrt{2}/10$, with $k = 3, 4, \ldots, 40$.

Let q and r be the coordinates in the (reference) unit square. We set n_q and n_r to be the number of cells in each coordinate direction. The (spatial) step size is given by $h_q = 1/n_q$ and $h_r = 1/n_r$, and our grid on the unit square is given by

$$q_i = ih_q,\ i = 0, \ldots, n_q, \quad r_j = jh_r,\ j = 0, \ldots, n_r.$$

We set the arc length of the outer arc at radii r_{out} of the annular sector to be the length, $L = 5$. Thus for the quarter annulus we have $r_{\text{out}} = \frac{2L}{\pi}$, and for the half-annulus we have $r_{\text{out}} = \frac{L}{\pi}$. For both cases, we take $r_{\text{in}} = r_{\text{out}} - 1$. Precisely, the coordinates of the two grids used for the quarter annular sector and the half annular sector are

$$x_{ij} = (r_{\text{in}} + (r_{\text{out}} - r_{\text{in}})q_i) \cos\left(n_{\text{an}} \frac{\pi}{2} r_j - \frac{\pi}{2}\right),$$
$$y_{ij} = (r_{\text{in}} + (r_{\text{out}} - r_{\text{in}})q_i) \sin\left(n_{\text{an}} \frac{\pi}{2} r_j - \frac{\pi}{2}\right).$$

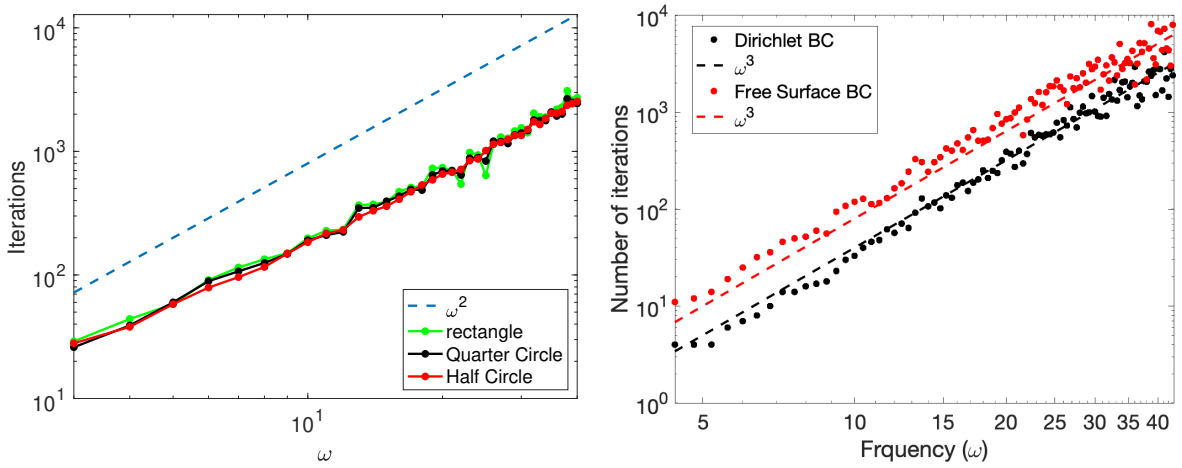


Figure 4.3: The number of iterations as a function of frequency to reach convergence for (Left) a rectangle, quarter circle and half circle, and (Right) the unit cube with Dirichlet or free surface conditions.

Here n_{an} is either 1 or 2 to indicate the quarter or half annulus, respectively. We set $n_r = 4L\omega + 1, n_q = 4\omega + 1$ so that the number of points per shear wavelength is around 20.

For the forcing we use a discrete approximation of the delta function with amplitude $\omega^2 \cos(\omega t)$. We locate this point source at $(x_{i^*j^*}, y_{i^*j^*})$ where $i^* = j^* = (n_q+1)/2$ so that it is close to $(0.5, 0.5)$ in physical space. In Figure 4.3 we display the number of iterations required to reduce the relative residual in the conjugate residual method by a factor 10^{-8}. From Figure 4.3 we see that the results for the three geometries are very similar, indicating that (in this example at least) the geometry has little to no effect on the number of iterations needed. Moreover the number of iterations grow as ω^2, as expected.

4.4.6 Effect of Boundary Conditions in A Cube

In this experiment we consider the unit cube with either Dirichlet boundary conditions on all sides, or with free surface boundary conditions on the top and bottom ($z = 0$ and $z = 1$) with Dirichlet boundary conditions on all other sides. We use the 3D version of the finite difference method described above with the standard explicit time-stepping method. Here we use the forcing

$$f_j(\mathbf{x}, t) = A_j e^{-\frac{\sigma}{2}\|\mathbf{x}-\mathbf{x}_j\|^2},$$

Table 4.4: The effect on iteration count depending on different combinations of λ and μ.

λ	1	2	4	8	16	32	64	1	1
μ	1	1	1	1	1	1	1	1/4	1/16
$h_{\max} \times 10^2$	3.37	3.37	3.37	3.37	3.37	3.37	3.37	1.10	0.852
$h_{\min} \times 10^2$	2.25	2.25	2.25	2.25	2.25	2.25	2.25	1.70	0.547
#Iter.	45	47	35	38	38	56	44	126	247
#Iter. $\times\sqrt{\mu}$	45	47	35	38	38	56	44	63	62

with $A_j \sim \sqrt{\sigma^d}$ and with $\sigma \sim \omega$ so that each of the components of the forcing approaches a delta function as ω grows. We select $\mathbf{x}_j$ sightly different for each j so that both $\nabla \times \mathbf{f} \neq 0$ and $\nabla \cdot \mathbf{f} \neq 0$, resulting in a solution with both shear and pressure waves.

We use the conjugate residual method, keep the product $h\omega = 0.4$ fixed, and report the number of iterations required to reduce the initial residual (starting from zero initial data) by a factor 10^{-9}. The result, which can be found in Figure 4.3, confirm the prediction from [14] that the number of iterations scale as ω^3.

4.4.7 Iteration Count as a Function of Wave Speed Ratio

The length of a domain, when measured in number of wavelengths, will increase both if the physical domain size is increased and if the wave speed is reduced. The compressional and shear wave speeds are $C_{\mathrm{p}} = \sqrt{(2\mu + \lambda)/\rho}$ and $C_{\mathrm{s}} = \sqrt{\mu/\rho}$, respectively. We expect that the number of iterations will depend on the smallest wave speed, but for El WaveHoltz there is no intuitive reason to think that a problem with $C_{\mathrm{p}} \gg C_{\mathrm{s}}$ should be more difficult than a problem with $C_{\mathrm{p}} \approx C_{\mathrm{s}}$. We note, however, that such behavior has been reported in the literature (see e.g. Table 3.1 on page 11 of [102]) for other methods.

To experimentally investigate how well El WaveHoltz works for different combinations of μ and λ, we use the SIPDG solver with the corrected explicit time-stepper for a geometry consisting of the unit square with a circular hole cut out (see Figure 4.4). This is the mesh `square-disc-nurbs.mesh`, which is part of the MFEM distribution. The Lamé parameters are constant in space and we choose

Table 4.5: Comparison of the number of iterations for the three different methods.

		$p=1$	2	3	4	5	6	7	8	9
h	WH	57	102	115	126	132	138	141	144	148
$h/2$	WH	68	108	114	126	130	133	138	143	146
$h/4$	WH	79	107	114	122	130	135	137	142	146
$h/8$	WH	83	108	114	125	130	133	137	142	146
h	IWH	81	182	160	173	213	237	192	235	201
$h/2$	IWH	97	151	157	168	220	263	188	276	196
$h/4$	IWH	112	147	153	163	171	178	264	188	275
$h/8$	IWH	117	147	154	165	171	213	217	N/A	N/A
h	HH	480	25084	18298	37926	80262	144863	204694	230688	500000
$h/2$	HH	15686	32063	57338	106688	256801	347279	500000	500000	500000
$h/4$	HH	46667	79865	184331	334665	500000	500000	500000	500000	500000
$h/8$	HH	500000	304561	500000	500000	500000	500000	500000	N/A	N/A

the number of refinements so that the solution is well resolved (the largest and smallest element size is reported in Table 4.4). We impose the boundary conditions

$$u(0, y, t) = v(0, y, t) = \cos(\omega t),$$

on the outer part of the domain, and let the circular hole be free of traction. For all experiments we set $\omega = 25.12$, CFL $= 0.8$ and we evolve the El WaveHoltz iteration over $K = 3$ periods. We stop the CG iteration when the relative residual falls below 10^{-6}. In Figure 4.4 we display the magnitude of the displacement and the components of the stress tensor σ_{xy}, σ_{xx} and σ_{yy} for the case when $\lambda = 1$ and $\mu = 1/16$.

The results, displayed in Table 4.4, show that El WaveHoltz appears to be robust with respect to the ratio between λ and μ. Moreover, the number of iterations to reach the desired tolerance is primarily a function of the μ, or equivalently, the shear wave speed C_{s}.

4.4.8 Comparison of Explicit and Implicit El WaveHoltz with Direct Discretization of Elastic Helmholtz

In this example we compare the explicit error corrected SIPDG method, the implicit error corrected SIPDG method and the SIPDG method of `example17p` extended to the elastic Helmholtz

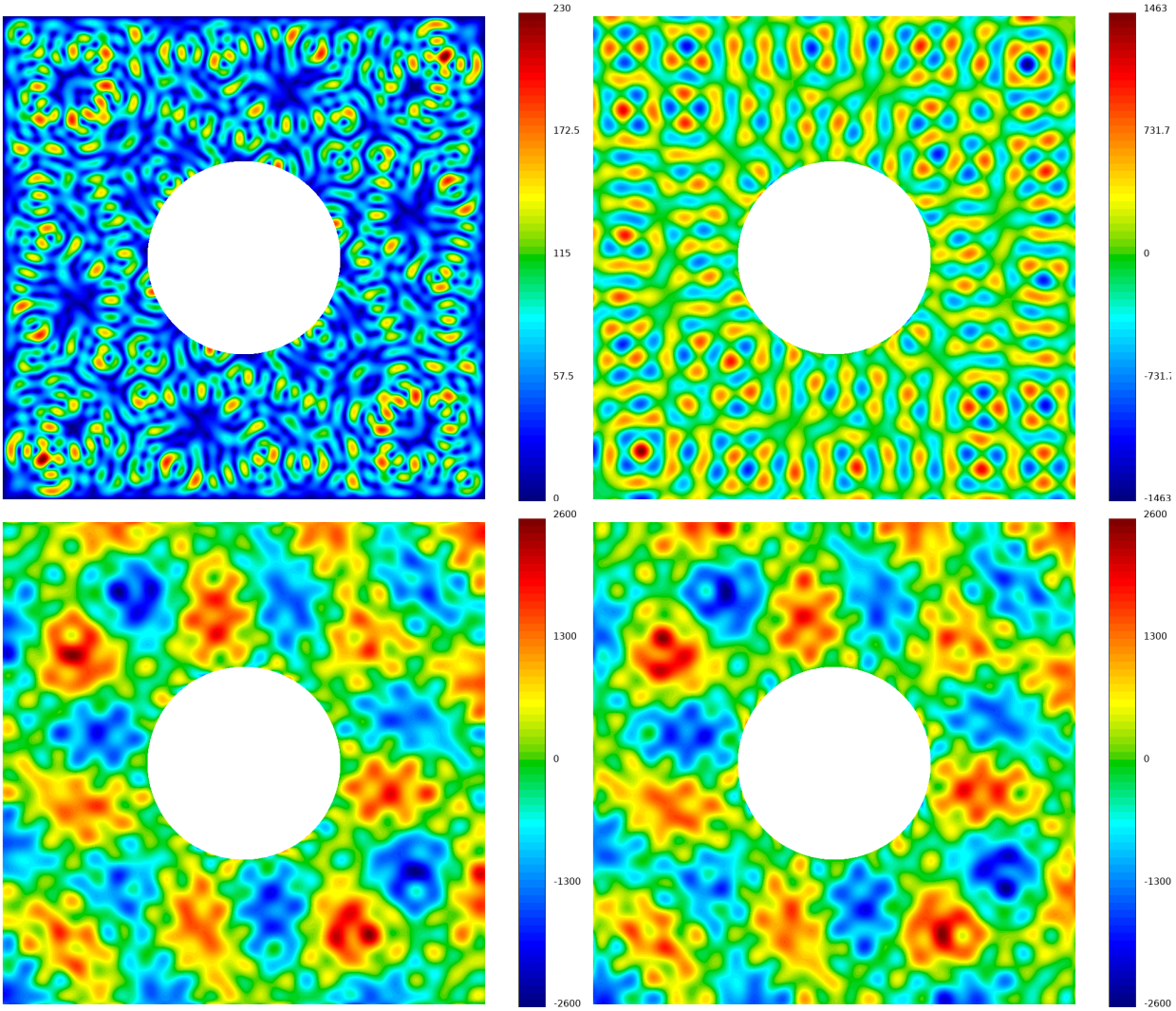


Figure 4.4: From top left to bottom right: displacement magnitude, σ_{xy}, σ_{xx} and σ_{yy}.

problem (4.2). In this section we will refer to these solvers by the abbreviations WH, IWH and HH respectively. For the Helmholtz SIPDG solver we use GMRES preconditioned by the AMG solver provided by `HypreBoomerAMG`, with the elasticity specific options (see also [17]) provided by `SetElasticityOptions`. The GMRES solver is restarted every 100 iterations.

For the implicit solver we must also invert the elasticity operator (but with a shift that preserves its positive definiteness) and we do this by CG preconditioned by the same AMG setup as for the Helmholtz SIPDG solver. We always use 10 time-steps for the implicit solver, and for the explicit solver we use CFL = 1.1. For the WH and IWH solver we solve the El WaveHoltz problem with conjugate gradient. For all three solvers the tolerance is set to be 10^{-10}.

We solve the equations on the unit square with a smooth (but narrow) forcing

$$\mathbf{f} = -\frac{200\omega^2}{\pi} e^{-1.2\omega^2[(x-0.25)^2+(y-0.25)^2]} \begin{pmatrix} -y+0.5 \\ x-0.5 \end{pmatrix},$$

and with free surface boundary conditions on all sides. We consider different refinements and polynomial degrees from 2 to 9 and estimate the error in the solution by computing a reference solution using degree 11 polynomials (note that the solutions are the same up to the tolerance of the iterative solvers since we have eliminated the time errors).

For all of the computations we record the number of iterations (a maximum of 500000) and list them in Table 4.5. It can be seen that the fewest number of iterations are achieved with the WH method. We also note that the number of iterations for WH is insensitive to the mesh resolution and has a weak dependence on the polynomial degree. Again, the latter is due to the fact that the linear system we are solving comes from a bounded operator so that the condition number does not depend on h. HH, which discretizes an unbounded and indefinite operator, behaves radically different with the number of iterations increasing rapidly with decreasing mesh resolution. In addition, the number of iterations for the HH method increases very quickly with the polynomial degree and, as a result, many of the accurate test cases fail to converge. Similar to the WH method, the IWH method has an iteration count that is relatively robust under grid refinement but with a slight increase with order. We note that it does appear the higher order methods have more

variation in iteration count than the lower order methods, and in general the iteration count is larger than for the WH method.

The number of iterations displayed in Table 4.5 are not useful for comparing the different methods as each iteration comes with a different computational cost. In Table 4.6 we instead list the number of right hand side (rhs) evaluations. By a right hand side evaluation we mean a single application of the matrix corresponding to the matrix discretizing the elastic operator. For the explicit method, the total number of rhs evaluations is $N_T N_{\text{iter}}$, where N_T is the number of time-steps needed to evolve the elastic wave equation one period and N_{iter} is the number of iterations. For the IWH method we always take $N_T = 10$ so that the number of rhs evaluations is $10 N_{\text{inner}} N_{\text{iter}}$, where N_{inner} is the number of inner iterations used by the AMG preconditioner (as reported in Table 4.5). For the HH method the number of rhs evaluations is equal to the number of GMRES iterations.

As can be seen in Table 4.6, the WH method is also more efficient with respect to the number of rhs evaluations (note that we report total number of rhs for the WH method and multipliers for the other methods). The advantage of the explicit method over the implicit method appears to be decreasing with increased accuracy - both in terms of decreasing mesh size and increased polynomial order. This is not unexpected as the number of time-steps needed for the explicit method grows linearly with the reciprocal of the mesh size, and quadratically with the polynomial degree while the implicit method maintains the number of time-steps constant. In terms of rhs evaluations, the gap between the WH method and the HH method is smaller than between WH and IWH; though the HH method degrades with increasing mesh refinement.

Finally, in Table 4.7 we report the increase in compute time (as a multiplicative factor) for the IWH and HH methods relative to the time required to solve the same problem with the explicit WH method. Throughout, the WH method is one to two orders of magnitude faster than the other methods. The HH method becomes less attractive (especially when it stops converging) as the accuracy is increased, while the IWH method improves with increased accuracy. It is of note that neither the number of iterations nor the number of rhs is a good predictor of compute time.

Table 4.6: The number of right hand side evaluations (estimated) for the three different methods. The top four rows display the actual number of right hand side evaluations and the rows below indicate how many times more the HH and IWH method evaluates the right hand side. An infinity sign indicates that the computation did not converge.

		$p=1$	2	3	4	5	6	7	8	9
h	WH	1425	4998	9315	15120	22176	30774	40326	51552	64676
$h/2$	WH	3400	10476	18354	30240	43550	59318	78936	102245	127458
$h/4$	WH	7821	20758	36594	58438	86970	120285	156728	202918	254916
$h/8$	WH	16434	41904	73188	119750	173940	236873	313456	405836	509686
h	HH	1	5	2	3	4	5	5	4	∞
$h/2$	HH	5	3	3	4	6	6	∞	∞	∞
$h/4$	HH	6	4	5	6	∞	∞	∞	∞	∞
$h/8$	HH	∞	7	∞	∞	∞	∞	∞	N/A	N/A
h	IWH	81	96	67	63	72	72	53	60	48
$h/2$	IWH	72	67	59	53	69	70	46	59	40
$h/4$	IWH	59	59	51	48	46	41	57	36	49
$h/8$	IWH	50	53	47	43	42	46	42	N/A	N/A

Table 4.7: The table reports how many times longer a computation with the HH and IWH method takes compared to the explicit WH method.

	p /meth	1	2	3	4	5	6	7	8	9
h	HH	15	164	56	56	60	64	62	59	∞
$h/2$	HH	265	157	109	96	117	113	∞	∞	∞
$h/4$	HH	587	311	91	260	∞	∞	∞	∞	∞
$h/8$	HH	∞	224	∞	∞	∞	∞	∞	N/A	N/A
h	IWH	192	197	137	113	103	90	59	73	44
$h/2$	IWH	169	185	125	112	107	121	55	42	29
$h/4$	IWH	261	295	61	152	62	68	71	36	42
$h/8$	IWH	28	30	100	69	62	46	34	N/A	N/A

Possible causes for this discrepancy are, a) that we only counted one right hand side evaluation per iteration and neglected the cost of the AMG preconditioner, and b) that the GMRES solve for the HH method actually has a significantly higher cost than conjugate gradient.

4.4.9 Materials with Spatially Varying Properties

We next consider an example taken from [79] with elastic propagation in a heterogeneous medium. We define the domain as $\Omega = [0, 2] \times [0, 1]$ where there is an embedded inclusion, $\Omega_I = [0.5, 1.5] \times [0.4, 0.6]$, in the middle from a stiffer material. We let Ω_I be a material with $\lambda = 200, \mu = 100$, while the domain $\Omega \setminus \Omega_I$ has $\lambda = 2$ and $\mu = 1$. We impose traction-free boundary conditions at $y = 0, 1$ and at $x = 1$. We additionally have

$$\mathbf{u}(0, y, t) = \begin{pmatrix} 0 \\ \cos(\omega t) \end{pmatrix}, \quad \mathbf{f}(x, y, t) \begin{pmatrix} 0 \\ \delta(|x - x_0| + |y - y_0|) \cos(\omega t) \end{pmatrix},$$

where δ is a delta function centered at $x_0 = 0.1$ and $y_0 = 0.5$.

We use the SIPDG method of Section 4.3.2 with a uniform quadrilateral mesh and polynomial degree $p = 6$. We consider two meshes with element widths of $h_1 = 1/20$ and $h_2 = 1/40$, respectively. We use CFL $= 0.4$ with the (corrected) explicit leapfrog time-stepper of Section 4.3.3, integrate over five periods, and accelerate convergence with conjugate gradient with a relative residual tolerance of 10^{-5}. We choose the frequency $\omega = 100$ so that we have at least one element per

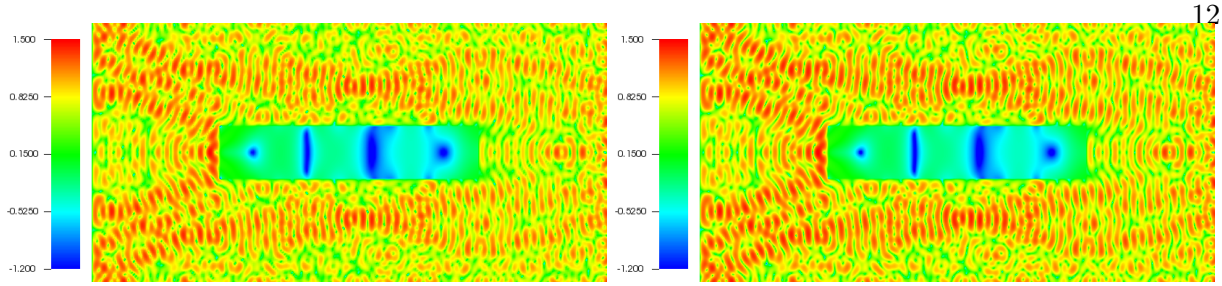


Figure 4.5: The $\log_{10}$ of the magnitude of the displacements for the CG accelerated solution of WH for the inclusion problem using sixth order polynomials within each element. (Left) Solution using a grid resolution of at least one element per wavelength, and (Right) two elements per wavelength.

wavelength when using element widths of h_1, and at least two when the widths are h_2. We plot the $\log_{10}$ of the magnitude of the displacement vector in Figure 4.5.

From Figure 4.5 it is clear that the solution using one element per wavelength is visually quite similar to that of the refined mesh with two elements per wavelength. Thus using one element per wavelength with a higher order polynomial order is sufficient to produce reasonable results, as was similarly seen in [103].

4.4.10 Vibrations of a Toroidal Shell

Finally, as a more realistic example in three dimensions we perform a simulation of a toroidal shell parametrized by

$$x(\theta,\phi,r) = (R + r\cos(\theta))\cos(\phi), \quad y = (\theta,\phi,r)(R + r\cos(\theta))\cos(\phi), \quad z(\theta,\phi,r) = r\sin(\theta).$$

Here we set $R = 4$, and let the partial toroidal shell occupy the volume $1 \leq r \leq 2$, and $0 \leq \phi, \theta \leq \pi$. The surfaces at $r = 1$ and $r = 2$ are free, and we impose homogeneous Dirichlet conditions on all other boundaries.

We force the problem by

$$\mathbf{f} = \frac{\sqrt{\sigma^3}}{20}\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} e^{-\zeta^2}\cos(\omega t), \quad \sigma = 100\omega, \quad \zeta^2 = 0.5\sigma((x-4)^2 + (y-0.5)^2 + (z-1)^2$$

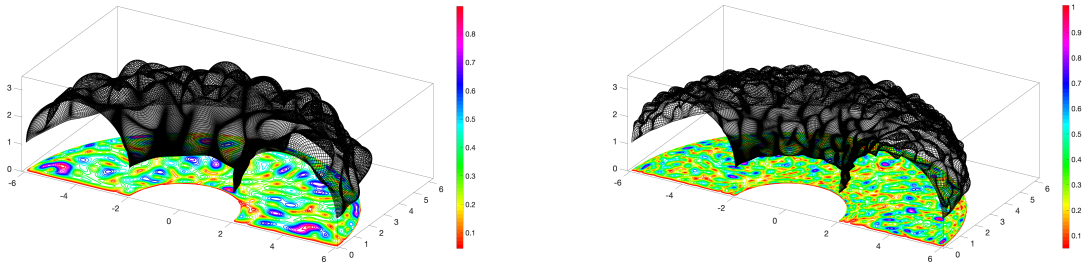


Figure 4.6: The solution in the toroidal shell for (Left) $\omega = 5.1234$, and (Right) $\omega = 10.2468$. The projection onto the xy-plane is the magnitude of the displacement on the outermost free surface $r = 2$. In black we display the (scaled) displaced mesh for $r = 2$.

We consider two cases. In the first case the frequency is $\omega = 5.1234$ and we use a grid that consists of $400 \times 120 \times 40$ points. In the second case the frequency is $\omega = 10.2468$ and we use a grid that consists of $800 \times 240 \times 80$ points.

The converged solution is displayed in Figure 4.6. The projection onto the xy-plane is the magnitude of the displacement $\sqrt{u^2 + v^2 + w^2}$ on the outermost free surface, $r = 2$. The mesh is the grid for that outermost surface with the (scaled) displacements added to the grid coordinates.

4.5 Conclusion

In this chapter we applied the WaveHoltz iteration, a time-domain Krylov accelerated fixed-point iteration, to the solution of the elastic Helmholtz equation for interior problems with Dirichlet and/or free surface boundary conditions. With symmetric discretizations, the iteration results in a positive definite and symmetric matrix, a notable advantage over direct discretizations of the elastic Helmholtz equation which typically lead to highly indefinite systems. In this work we have also introduced a new implicit time-stepping scheme and demonstrated that its use in the WaveHoltz iteration completely removes time discretization errors.

The implicit method did not offer any advantages for the one problem we considered here, but we believe it could be an advantageous approach for problems with anisotropic or refined non-conforming meshes. It could also prove to be a possible way to construct preconditioners.

Finally, here we have only considered the energy conserving problem. In the future, we will revisit the elastic Helmholtz problem with impedance/absorbing boundary conditions which are a hallmark of scattering and seismic applications.

Chapter 5

Optimal Control of Closed Quantum Systems via B-Splines with Carrier Waves

In this chapter we consider the quantum optimal control problem of determining electromagnetic pulses for implementing unitary gates in a quantum computer. A truncated modal expansion of Schrödinger's equation is used to model the quantum system, in which the state of the quantum system is described by the state vector[1] $\boldsymbol{\psi} \in \mathbb{C}^N$. The elements of the state vector are complex probability amplitudes, where the magnitude squared of each element represents the probability that the quantum system occupies the corresponding energy level [90]. Because the probabilities must sum to one, the state vector is normalized such that $\|\boldsymbol{\psi}\|_2^2 = 1$. The evolution of the state vector in the time interval $t \in [0, T]$ is governed by Schrödinger's equation:

$$\frac{d\boldsymbol{\psi}}{dt} + iH(t;\boldsymbol{\alpha})\boldsymbol{\psi} = 0, \quad 0 \le t \le T, \quad \boldsymbol{\psi}(0) = \boldsymbol{g}. \tag{5.1}$$

Here, $i = \sqrt{-1}$ is the imaginary unit and $\boldsymbol{g}$ is the initial state. The Hamiltonian matrix $H(t;\boldsymbol{\alpha}) \in \mathbb{C}^{N\times N}$ (scaled such that Planck's constant becomes $\hbar = 1$) is Hermitian and is assumed to be of the form

$$H(t;\boldsymbol{\alpha}) = H_s + H_c(t;\boldsymbol{\alpha}), \tag{5.2}$$

where H_s and H_c are the system and control Hamiltonians, respectively. The control Hamiltonian models the action of external control fields on the quantum system. The time dependence in the control Hamiltonian is parameterized by the control vector $\boldsymbol{\alpha} \in \mathbb{R}^D$. As a result, the state vector $\boldsymbol{\psi}$ depends implicitly on $\boldsymbol{\alpha}$ through Schrödinger's equation.

[1] This chapter uses conventional matrix-vector notation. For finite-dimensional systems, it is equivalent to the bra-ket notation that often is used in quantum physics.

To justify the truncation of the modal expansion of Schrödinger's equation, we divide the state vector into $E > 0$ "essential" levels and $G \geq 0$ "guard" levels, such that $E + G = N$. The population of the highest guard levels need to be small to minimize coupling to even higher energy levels, which are excluded from the model.

The goal of the quantum optimal control problem is to determine the parameter vector $\boldsymbol{\alpha}$ such that the time-dependence in the Hamiltonian matrix leads to a solution of Schrödinger's equation such that $\boldsymbol{\psi}(T) \approx V_{tg}\boldsymbol{g}$, where V_{tg} is the target gate transformation. The gate transformation should be satisfied for all initial conditions in the essential subspace of the state vector. A basis of this subspace is provided by the matrix $U_0 \in \mathbb{R}^{N\times E}$. The definitions of U_0 and V_{tg} are described in Appendix .12.

To account for any initial condition in the essential subspace, we define the solution operator matrix $U \in \mathbb{C}^{N\times E}$. Each column of this matrix satisfies (5.1), leading to Schrödinger's equation in matrix form,

$$\frac{dU(t)}{dt} + iH(t;\boldsymbol{\alpha})U(t) = 0, \quad 0 \leq t \leq T, \quad U(0) = U_0. \tag{5.3}$$

The overlap between the target gate matrix and the solution operator matrix at the final time is defined by

$$O_{V_{tg}} := \langle U(T;\boldsymbol{\alpha}), V_{tg}\rangle_F \,, \tag{5.4}$$

where $\langle\cdot,\cdot\rangle_F$ denotes the Frobenious matrix scalar product. Because U_0 spans an E-dimensional subspace of initial conditions, we have $|O_T| \leq E$. The difference between $U(T;\boldsymbol{\alpha})$ and V_{tg} can be measured by the target gate infidelity [71, 78, 82, 83, 98],

$$\mathcal{J}_1(U_T(\boldsymbol{\alpha})) := 1 - \frac{1}{E^2}\left|\langle U_T(\boldsymbol{\alpha}), V_{tg}\rangle_F\right|^2, \quad U_T(\boldsymbol{\alpha}) := U(T;\boldsymbol{\alpha}). \tag{5.5}$$

Note that the target gate infidelity is invariant to global phase differences between U_T and V_{tg}. In quantum physics, the global phase of a state is considered irrelevant because it can not be measured.

The leakage of population to the guard states can be measured by the objective function

$$\mathcal{J}_2(U(\cdot;\boldsymbol{\alpha})) = \frac{1}{T}\int_0^T \langle U(t;\boldsymbol{\alpha}), WU(t;\boldsymbol{\alpha})\rangle_F \, dt, \tag{5.6}$$

where W is a diagonal $N \times N$ positive semi-definite weight matrix. The elements in W are zero for all essential states and are positive for the guard states. The elements of W are typically larger for higher energy levels in the model.

For the quantum control problem with guard states, we formulate the optimization problem as

$$\min_{\boldsymbol{\alpha}} \mathcal{G}(\boldsymbol{\alpha}) := \mathcal{J}_1(U_T(\boldsymbol{\alpha})) + \mathcal{J}_2(U(\cdot;\boldsymbol{\alpha})), \tag{5.7}$$

$$\frac{dU}{dt} + iH(t;\boldsymbol{\alpha})U = 0, \quad 0 \le t \le T, \quad U(0;\boldsymbol{\alpha}) = U_0, \tag{5.8}$$

$$\alpha_{min} \le \alpha_q \le \alpha_{max}, \quad q = 1, 2, \ldots, D. \tag{5.9}$$

For a discussion of the solvability of the quantum control problem, see for example Borzi et al. [27].

In the quantum optimal control problem, the Schrödinger (state) equation is a time-dependent Hamiltonian system. To ensure long-time numerical accuracy it is appropriate to discretize it using a symplectic time-integration method [67]. For this purpose we use the Störmer-Verlet method, which can be written as a partitioned Runge-Kutta scheme, based on the trapezoidal and implicit midpoint rules. Our main theoretical contribution is the generalization of Ober-Blöbaum's [92] work to the case of a time-dependent Hamiltonian system. We show that the compatible method for the adjoint state equation resembles a partitioned Runge-Kutta scheme, except that the time-dependent matrices must be evaluated at modified time levels.

Our approach builds upon the works of Hager [65], Sanz-Serna [97] and Ober-Blöbaum [92]. Hager [65] first showed how the Hamiltonian structure in an optimization problem can be utilized to calculate the gradient of the objective function. Hager considered the case in which the state equation is discretized by one Runge-Kutta scheme, with the adjoint state equation discretized by another Runge-Kutta scheme. It was found that the discrete gradient can be calculated exactly if the pair of Runge-Kutta methods satisfy the requirements of a symplectic partitioned Runge-Kutta method. Further details and generalizations are described in the review paper by Sanz-Serna [97]. Ober-Blöbaum [92] extended Hager's approach to the case where the state equation itself is a Hamiltonian system that is discretized by a partitioned Runge-Kutta scheme. For autonomous

state equations, it was shown that the compatible discretization of the adjoint state equation is another partitioned Runge-Kutta scheme.

Several numerical methods for the quantum control problem are based on the GRAPE algorithm [73]. In this case, Schrödinger's equation is discretized in time using the second order accurate Magnus scheme [67], in which the Hamiltonian matrix is evaluated at the midpoint of each time step. A stair-step approximation of the control functions is imposed such that each control function is constant within each time step. Thus, the time step determines both the numerical accuracy of the dynamics of the quantum state **and** the number of control parameters. With Q control functions, M time steps of size h, the control functions are thus described by M times Q parameters $\alpha_{j,k}$. The propagator in the Magnus method during the j^{th} time step is of the form $\exp(-ih(H_0 + \sum_k \alpha_{k,j} H_k))$. In general, the matrices H_0 and H_k do not commute, leading to an integral expression for the derivative of the propagator with respect to the parameters, which is needed for computing the gradient of the objective function. In the original GRAPE method, this integral expression is approximated by the first term in its Taylor series expansion, leading to an approximate gradient that is polluted by an $\mathcal{O}(h^2)$ error. As the gradient becomes smaller during the optimization, the approximation error will eventually dominate the numerical gradient, which may hamper the convergence of the optimization algorithm. A more accurate way of numerically evaluating the derivative of the time-step propagator can be obtained by retaining more terms in the Taylor series expansion, or by using a matrix commutator expansion [36]. More recently, the GRAPE algorithm has been generalized to optimize objective functions that include a combination of the target gate infidelity, integrals penalizing occupation of "forbidden states" and terms for imposing smoothness and amplitude constraints on the control functions. Here, automatic differentiation is used for computing the gradient of the objective function [78]. However, the number of control parameters is still proportional to the number of time steps, which may become very large when the duration of the gate is long, or the quantum state is highly oscillatory.

As an alternative to calculating the gradient of the objective function by solving an adjoint equation backwards in time, the gradient can be calculated by differentiating Schrödinger's equa-

tion with respect to each parameter in the control function, leading to a differential equation for each component of the gradient of the state vector. This approach, implemented in the GOAT algorithm [83], allows the gradient of the objective function to be calculated exactly, but requires $(D + 1)$ Schrödinger systems to be solved when the control functions depend on D parameters. This makes the method computationally expensive when the number of parameters is large.

Using the stair-stepped approximation of the control functions often leads to a large number of control parameters, which may hamper the convergence of the GRAPE algorithm. The total number of parameters can be reduced by instead expanding the control functions in terms of basis functions. By using the chain rule, the gradient from the GRAPE algorithm can then be used to calculate the gradient with respect to the coefficients in the basis function expansion. This approach is implemented in the GRAFS algorithm [82], where the control functions are expanded in terms of Slepian sequences.

Gradient-free optimization methods can also be applied to quantum optimal control problems. These methods do not rely on the gradient to be evaluated and are therefore significantly easier to implement. However, the convergence of these methods is usually much slower than for gradient-based techniques, unless the number of control parameters is very small. One example of a gradient-free methods for quantum optimal control is the CRAB algorithm [32].

Many parameterizations of quantum control functions have been proposed in the literature, for example cubic splines [48], Gaussian pulse cascades [38], Fourier expansions [110] and Slepian sequences [82].

This chapter presents a different approach, based on parameterizing the control functions by B-spline basis functions with carrier waves. Our approach relies on the observation that transitions between the energy levels in a quantum system are triggered by resonance, at frequencies that can be determined by inspection of the system Hamiltonian. The carrier waves are used to specify the frequency spectra of the control functions, while the B-spline functions specify their envelope and phase. We find that this approach allows the number of control parameters to be independent of, and significantly smaller than, the number of time steps for integrating Schrödinger's equation.

The remainder of the chapter is organized as follows. In Section 5.1, we introduce a Hamiltonian model and discuss the resonant frequencies needed to trigger transitions between the states in the system. These resonant frequencies naturally motivate us to parameterize the control functions using B-splines with carrier waves; details of this parameterization are presented in Section 5.2. In Section 5.3, we introduce a real-valued formulation of Schrödinger's equation and present the symplectic Störmer-Verlet scheme that we use for its time-integration. To achieve an exact gradient of the discrete objective function, we apply the "discretize-then-optimize" approach. Based on the Störmer-Verlet scheme, in Section 5.4 we outline the construction of a discrete adjoint time integration method. Section 5.5 presents numerical examples. We illustrate how the proposed technique, combined with the interior point L-BFGS algorithm [91] from the IPOPT package [106], is used to optimize control functions for multi-level qudit gates. We additionally consider a simple noise model and risk-neutral optimization to illustrate the construction of controls that are robust to uncertainty in the Hamiltonian. The proposed scheme is implemented in the Julia [24] programming language, in an open source package called Juqbox.jl [53]. In Section 5.6, we compare the performance of Juqbox.jl and two implementations of the GRAPE algorithm. Concluding remarks are given in Section 5.7.

5.1 Hamiltonian model

Several Hamiltonian models exist for describing the quantum physics of super-conducting circuits [26, 84]. In this paper, we consider a composite quantum system with $Q \geq 1$ subsystems (qubits/qudits) where the system Hamiltonian satisfies:

$$H_s = \sum_{q=1}^{Q} \left(\omega_q a_q^\dagger a_q - \frac{\xi_q}{2} a_q^\dagger a_q^\dagger a_q a_q - \sum_{p>q} \xi_{pq} a_p^\dagger a_p a_q^\dagger a_q \right). \tag{5.10}$$

In this model, ω_q is the ground state transition frequency and ξ_q is the self-Kerr coefficient of subsystem q; the cross-Kerr coefficient between subsystems p and q is denoted ξ_{pq}. Furthermore, subsystem q is assumed to have $n_q \geq 2$ energy levels, with lowering operator a_q. The lowering

operator is constructed using Kronecker products,

$$a_q := I_{n_Q} \otimes \cdots \otimes I_{n_{q+1}} \otimes A_q \otimes I_{n_{q-1}} \otimes \cdots \otimes I_{n_1} \in \mathbb{R}^{N\times N}, \quad N = \prod_{q=1}^{Q} n_q, \tag{5.11}$$

where I_n denotes the $n \times n$ identity matrix and the single-system lowering matrix satisfies

$$A_q := \begin{pmatrix} 0 & \sqrt{1} & & \\ & \ddots & \ddots & \\ & & \ddots & \sqrt{n_q - 1} \\ & & & 0 \end{pmatrix} \in \mathbb{R}^{n_q \times n_q}. \tag{5.12}$$

We consider a control Hamiltonian with real-valued control functions that are parameterized by the control vector $\boldsymbol{\alpha}$,

$$H_c(t;\boldsymbol{\alpha}) = \sum_{q=1}^{Q} f_q(t;\boldsymbol{\alpha})(a_q + a_q^\dagger), \quad f_q(t;\boldsymbol{\alpha}) = 2\,\mathrm{Re}\left(d_q(t;\boldsymbol{\alpha})\, e^{i\omega_{r,q}t}\right). \tag{5.13}$$

2179 where $\omega_{r,q}$ is the drive frequency in subsystem q.

2180 5.1.1 Rotating wave approximation

To slow down the time scales in the state vector, we apply a rotating frame transformation in Schrödinger's equation through the unitary change of variables $\widetilde{\boldsymbol{\psi}}(t) = R(t)\boldsymbol{\psi}(t)$, where

$$R(t) = \bigotimes_{q=Q}^{1} \exp\left(i\omega_{r,q}t\, A_q^\dagger A_q\right), \tag{5.14}$$

and $\otimes_{q=Q}^{1} C_q = C_Q \otimes C_{Q-1} \otimes \ldots \otimes C_1$. Note that we use $\omega_{r,q}$ as the frequency of rotation in subsystem q. The system Hamiltonian transforms into $H_s^{rw} = H_s - \sum \omega_{r,q} a_q^\dagger a_q$. Then, the rotating wave approximation is applied to transform the control Hamiltonian. Here, we substitute the laboratory frame control function $f_q(t;\boldsymbol{\alpha})$ from (5.13) and neglect terms oscillating with frequencies $\pm 2\omega_{r,q}$. As a result, the Hamiltonians (5.10) and (5.13) transform into (see Appendix .13 for details)

$$H_s^{rw} = \sum_{q=1}^{Q} \left(\Delta_q a_q^\dagger a_q - \frac{\xi_q}{2} a_q^\dagger a_q^\dagger a_q a_q - \sum_{p>q} \xi_{qp} a_q^\dagger a_q a_p^\dagger a_p\right), \tag{5.15}$$

$$H_c^{rw}(t;\boldsymbol{\alpha}) = \sum_{q=1}^{Q} \left(d_q(t;\boldsymbol{\alpha})a_q + \bar{d}_q(t;\boldsymbol{\alpha})a_q^\dagger\right), \tag{5.16}$$

where $\Delta_q = \omega_q - \omega_{r,q}$ is called the detuning frequency. The main advantages of the rotating frame approximation are the reduction of the spectral radius in the system Hamiltonian (5.15), and the absence of the highly oscillatory factor $\exp(i\omega_{r,q}t)$ in the control Hamiltonian (5.16). In the following we assume that the rotating wave approximation has already been performed, and the tilde on the state vector will be suppressed. We additionally note that the target unitary V_{tg} is similarly transformed into the rotating frame via $V_{tg}^{rw} = R(T)V_{tg}$.

5.1.2 Resonant frequencies

To simplify the presentation we restrict our analysis to a bipartite quantum system, i.e., $Q = 2$. The system Hamiltonian (5.15) is diagonal and we denote its elements by

$$\{H_s^{rw}\}_{\boldsymbol{j},\boldsymbol{k}} = \begin{cases} \kappa_{\boldsymbol{j}}, & \boldsymbol{j} = \boldsymbol{k}, \\ 0, & \text{otherwise}, \end{cases} \qquad \kappa_{\boldsymbol{j}} = \sum_{q=1}^{2} \left(\Delta_q j_q - \frac{\xi_q}{2} j_q(j_q - 1) \right) - \xi_{12} j_1 j_2, \tag{5.17}$$

for $j_q \in [0, n_q - 1]$ and where $\boldsymbol{j} = (j_2, j_1)$ is a multi-index. Let us consider the case when the control functions $d_k(t)$ oscillate with carrier wave frequencies $\{\Omega_1, \Omega_2\}$, and amplitude ϵ, where $0 < \epsilon \ll 1$. These assumptions give

$$H_c^{rw}(t) = \epsilon H_1(t), \quad H_1(t) = \sum_{k=1}^{2} \left(e^{i\Omega_k t} a_k + e^{-i\Omega_k t} a_k^\dagger \right). \tag{5.18}$$

We make an asymptotic expansion of the solution of Schrödinger's equation (5.1), $\boldsymbol{\psi} = \boldsymbol{\psi}^{(0)} + \epsilon\boldsymbol{\psi}^{(1)} + \mathcal{O}(\epsilon^2)$. The zero'th and first order terms satisfy

$$\frac{d\boldsymbol{\psi}^{(0)}}{dt} + iH_s^{rw}\boldsymbol{\psi}^{(0)} = 0, \qquad \boldsymbol{\psi}^{(0)}(0) = \boldsymbol{g}, \tag{5.19}$$

$$\frac{d\boldsymbol{\psi}^{(1)}}{dt} + iH_s^{rw}\boldsymbol{\psi}^{(1)} = \boldsymbol{f}(t), \quad \boldsymbol{\psi}^{(1)}(0) = \boldsymbol{0}. \tag{5.20}$$

Because the system Hamiltonian is diagonal, (5.19) is a decoupled system of ordinary differential equation that is solved by $\psi_{\boldsymbol{j}}^{(0)}(t) = g_{\boldsymbol{j}} e^{-i\kappa_{\boldsymbol{j}} t}$. The right hand side of (5.20) satisfies $\boldsymbol{f}(t) := -iH_1(t)\boldsymbol{\psi}^{(0)}(t)$, which can be written

$$\boldsymbol{f}(t) = \sum_{k=1}^{Q} \boldsymbol{f}^{(k)}(t), \quad \boldsymbol{f}^{(k)}(t) = -i \left(e^{i\Omega_k t} a_k + e^{-i\Omega_k t} a_k^\dagger \right) \boldsymbol{\psi}^{(0)}(t). \tag{5.21}$$

Because the matrix H_s^{rw} is diagonal, the system for the first order perturbation, (5.20), is also decoupled. We are interested in cases when $\psi_{\boldsymbol{j}}^{(1)}(t)$ grows in time, corresponding to resonance. Let $\boldsymbol{e}_k$ denote the k^{th} unit vector and denote a shifted multi-index by $\boldsymbol{j} \pm \boldsymbol{e}_1 = (j_2, j_1 \pm 1)$ and $\boldsymbol{j} \pm \boldsymbol{e}_2 = (j_2 \pm 1, j_1)$.

Lemma 5.1.1. *The perturbation of the state vector, $\psi_{\boldsymbol{j}}^{(1)}(t)$, grows linearly in time when the carrier wave frequencies and the initial condition satisfy:*

$$\Omega_k = \kappa_{\boldsymbol{j}+\boldsymbol{e}_k} - \kappa_{\boldsymbol{j}}, \quad g_{\boldsymbol{j}+\boldsymbol{e}_k} \neq 0, \quad j_k \in [0, n_k - 2], \tag{5.22}$$

$$\Omega_k = \kappa_{\boldsymbol{j}} - \kappa_{\boldsymbol{j}-\boldsymbol{e}_k}, \quad g_{\boldsymbol{j}-\boldsymbol{e}_k} \neq 0, \quad j_k \in [1, n_k - 1], \tag{5.23}$$

for $k = \{1, 2\}$.

Proof. See Appendix .14. □

We can evaluate the conditions for resonance by inserting the Hamiltonian elements from (5.17) into (5.22) and (5.23). For $k = 1$ and $j_2 \in [0, n_2 - 1]$, resonance occurs in $\psi_{\boldsymbol{j}}^{(1)}(t)$ when

$$\Omega_1 = \begin{cases} \Delta_1 - \xi_1 j_1 - \xi_{12} j_2, & g_{\boldsymbol{j}+\boldsymbol{e}_1} \neq 0,\ j_1 \in [0, n_1 - 2], \\ \Delta_1 - \xi_1 (j_1 - 1) - \xi_{12} j_2, & g_{\boldsymbol{j}-\boldsymbol{e}_1} \neq 0,\ j_1 \in [1, n_1 - 1]. \end{cases} \tag{5.24}$$

For $k = 2$ and $j_1 \in [0, n_1 - 1]$, the resonant cases are

$$\Omega_2 = \begin{cases} \Delta_2 - \xi_2 j_2 - \xi_{12} j_1, & g_{\boldsymbol{j}+\boldsymbol{e}_2} \neq 0,\ j_2 \in [0, n_2 - 2], \\ \Delta_2 - \xi_2 (j_2 - 1) - \xi_{12} j_1, & g_{\boldsymbol{j}-\boldsymbol{e}_2} \neq 0,\ j_2 \in [1, n_2 - 1]. \end{cases} \tag{5.25}$$

For example, when $n_1 = 3$, $n_2 = 3$ and $g_{\boldsymbol{j}} \neq 0\ \forall \boldsymbol{j}$, the carrier wave frequencies:

$$\Omega_1 = \Big[\Delta_1,\ \ \Delta_1 - \xi_{12},\ \ \Delta_1 - 2\xi_{12},\ \ \Delta_1 - \xi_1,\ \ \Delta_1 - \xi_1 - \xi_{12},\ \ \Delta_1 - \xi_1 - 2\xi_{12}\Big],$$
$$\Omega_2 = \Big[\Delta_2,\ \ \Delta_2 - \xi_{12},\ \ \Delta_2 - 2\xi_{12},\ \ \Delta_2 - \xi_2,\ \ \Delta_2 - \xi_2 - \xi_{12},\ \ \Delta_2 - \xi_2 - 2\xi_{12}\Big],$$

lead to resonance.

Since Schrödinger's equation conserves total probability, the linear growth in time only occurs for short times. Thus, each resonant frequency corresponds to the initiation of a transition between two energy levels in the quantum system.

5.2 Quadratic B-splines with carrier waves

Motivated by the results from the previous section, we parameterize the rotating frame control functions using basis functions that act as envelopes for carrier waves with fixed frequencies:

$$d_k(t;\boldsymbol{\alpha}) = \sum_{n=1}^{N_f} d_{k,n}(t;\boldsymbol{\alpha}), \quad d_{k,n}(t;\boldsymbol{\alpha}) = \sum_{b=1}^{N_b} \hat{S}_b(t)\alpha_{b,n}^k\, e^{it\Omega_{k,n}}, \quad k \in [1,Q]. \tag{5.26}$$

Here, $\Omega_{k,n} \in \mathbb{R}$ is the n^{th} carrier wave frequency for system k. These frequencies are chosen to match the resonant frequencies in the system Hamiltonian (5.15), as outlined above. The complex coefficients $\alpha_{b,n}^k = \alpha_{b,n}^{k(r)} + i\alpha_{b,n}^{k(i)}$ are control parameters that are to be determined through optimization, corresponding to a total of $D = 2QN_bN_f$ real-valued parameters. It is convenient to also define the real-valued functions

$$p_{k,n}(t;\boldsymbol{\alpha}) = \sum_{b=1}^{N_b} \hat{S}_b(t)\alpha_{b,n}^{k(r)}, \quad q_{k,n}(t;\boldsymbol{\alpha}) = \sum_{b=1}^{N_b} \hat{S}_b(t)\alpha_{b,n}^{k(i)}, \tag{5.27}$$

such that $d_{k,n}(t;\boldsymbol{\alpha}) = (p_{k,n}(t;\boldsymbol{\alpha}) + iq_{k,n}(t;\boldsymbol{\alpha}))\exp(it\Omega_{k,n})$.

The basis functions $\hat{S}_b(t)$ are chosen to be piece-wise quadratic B-spline wavelets (see Figure 5.1), centered on a uniform grid in time,

$$t_m = (m-1.5)\delta, \quad m = 1,\ldots,D_1, \quad \delta = \frac{T}{D_1-2}. \tag{5.28}$$

Each basis function $\hat{S}_b(t)$ is centered around $t = t_m$ and is easily expressed in terms of the scaled time parameter $\tau_m(t) = (t-t_m)/3\delta$,

$$\hat{S}_b(t) = \widetilde{S}(\tau_m(t)), \quad \widetilde{S}(\tau) = \begin{cases} \frac{9}{8} + \frac{9}{2}\tau + \frac{9}{2}\tau^2, & -\frac{1}{2} \le \tau < -\frac{1}{6}, \\ \frac{3}{4} - 9\tau^2, & -\frac{1}{6} \le \tau < \frac{1}{6}, \\ \frac{9}{8} - \frac{9}{2}\tau + \frac{9}{2}\tau^2, & \frac{1}{6} \le \tau < \frac{1}{2}, \\ 0, & \text{otherwise.} \end{cases} \tag{5.29}$$

The basis function $\hat{S}_b(t)$ has local support for $t \in [t_m - 1.5\delta, t_m + 1.5\delta]$. Thus, for any fixed time t a control function will get contributions from at most three B-spline wavelets, allowing the control functions to be evaluated very efficiently. We also remark that the control function (5.26) can be

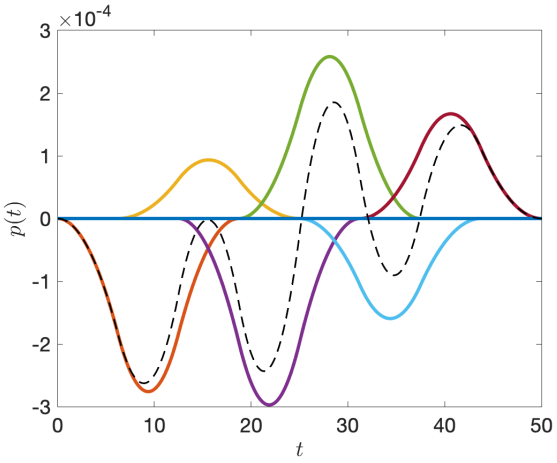


Figure 5.1: The real part of a quadratic B-spline control function, with zero carrier frequency (dashed black). The solid colored lines are the individual B-spline wavelets.

evaluated at any time $t \in [0, T]$. Importantly, this allows the time-integration scheme to be chosen independently of the parameterization of the control function, and allows the number of control parameters to be chosen independently of the number of time steps for integrating Schrödinger's equation.

5.3 Real-valued formulation

A real-valued formulation of Schrödinger's equation (5.1) is given by

$$\begin{bmatrix} \dot{\boldsymbol{u}} \\ \dot{\boldsymbol{v}} \end{bmatrix} = \begin{bmatrix} S(t) & -K(t) \\ K(t) & S(t) \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} =: \begin{bmatrix} f^u(\boldsymbol{u}, \boldsymbol{v}, t) \\ f^v(\boldsymbol{u}, \boldsymbol{v}, t) \end{bmatrix}, \quad \begin{bmatrix} \boldsymbol{u}(0) \\ \boldsymbol{v}(0) \end{bmatrix} = \begin{bmatrix} \boldsymbol{g}^u \\ \boldsymbol{g}^v \end{bmatrix}, \tag{5.30}$$

where,

$$\boldsymbol{u} = \mathrm{Re}(\boldsymbol{\psi}), \quad \boldsymbol{v} = -\mathrm{Im}(\boldsymbol{\psi}), \quad K = \mathrm{Re}\,(H), \quad S = \mathrm{Im}\,(H),$$

Because the matrix H is Hermitian, $K^T = K$ and $S^T = -S$. The real-valued formulation of Schrödinger's equation is a time-dependent Hamiltonian system corresponding to the Hamiltonian functional,

$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{v}, t) = \boldsymbol{u}^T S(t) \boldsymbol{v} + \frac{1}{2} \boldsymbol{u}^T K(t) \boldsymbol{u} + \frac{1}{2} \boldsymbol{v}^T K(t) \boldsymbol{v}. \tag{5.31}$$

In general, $S(t) \neq 0$, which makes the Hamiltonian system non-separable.

In terms of the real-valued formulation, let the columns of the solution operator matrix in (5.3) satisfy $U = [\boldsymbol{u}_1 - i\boldsymbol{v}_1,\ \boldsymbol{u}_2 - i\boldsymbol{v}_2,\ \ldots, \boldsymbol{u}_E - i\boldsymbol{v}_E]$. Here, $(\boldsymbol{u}_j, \boldsymbol{v}_j)$ satisfy (5.30) subject to the initial conditions $\boldsymbol{g}_j^v = \boldsymbol{0}$ and $\boldsymbol{g}_{\boldsymbol{j}}^u = \boldsymbol{e}_{\boldsymbol{j}}$, where $\boldsymbol{j} = (j_Q, j_{Q-1}, \ldots, j_1)$ is a multi-index such that $j_q \in \{0, 1, \ldots, m_q - 1\}$ and m_q is the number of essential levels of subsystem q. The columns in the target gate matrix $V_{tg} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_E]$ correspond to

$$V_{tg} = [\boldsymbol{d}_1^u - i\boldsymbol{d}_1^v,\ \boldsymbol{d}_2^u - i\boldsymbol{d}_2^v,\ \ldots, \boldsymbol{d}_E^u - i\boldsymbol{d}_E^v], \quad \boldsymbol{d}_j^u = \mathrm{Re}(\boldsymbol{d}_j), \quad \boldsymbol{d}_j^v = -\mathrm{Im}(\boldsymbol{d}_j).$$

Using the real-valued notation, the two parts of the objective function (5.7) can be written

$$\mathcal{J}_1(U_T(\boldsymbol{\alpha})) = \left(1 - \frac{1}{E^2}\left|S_V(U_T(\boldsymbol{\alpha}))\right|^2\right), \tag{5.32}$$

$$\mathcal{J}_2(U(\cdot, \boldsymbol{\alpha})) = \frac{1}{T}\sum_{j=0}^{E-1}\int_0^T \langle \boldsymbol{u}_j(t, \boldsymbol{\alpha}) - i\boldsymbol{v}_j(t, \boldsymbol{\alpha}), W(\boldsymbol{u}_j(t, \boldsymbol{\alpha}) - i\boldsymbol{v}_j(t, \boldsymbol{\alpha}))\rangle_2 \, dt, \tag{5.33}$$

where

$$O_V(U_T) = \sum_{j=0}^{E-1} \left\langle \boldsymbol{u}_j(T, \boldsymbol{\alpha}) - i\boldsymbol{v}_j(T, \boldsymbol{\alpha}), \boldsymbol{d}_j^u - i\boldsymbol{d}_j^v \right\rangle_2. \tag{5.34}$$

5.3.1 Time integration

Let $t_n = nh$, for $n = 0, 1, \ldots, M$, be a uniform grid in time where $h = T/M$ is the time step. Also let $\boldsymbol{u}^n \approx \boldsymbol{u}(t_n)$ and $\boldsymbol{v}^n \approx \boldsymbol{v}(t_n)$ denote the numerical solution on the grid. We use a partitioned Runge-Kutta (PRK) scheme [67] to discretize the real-valued formulation of Schrödinger's equation,

$$\boldsymbol{u}^0 = \boldsymbol{g}^u, \qquad \boldsymbol{v}^0 = \boldsymbol{g}^v, \tag{5.35}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + h\sum_{i=1}^s b_i^u \boldsymbol{\kappa}^{n,i}, \qquad \boldsymbol{v}^{n+1} = \boldsymbol{v}^n + h\sum_{i=1}^s b_i^v \boldsymbol{\ell}^{n,i}, \tag{5.36}$$

$$\boldsymbol{\kappa}^{n,i} = f^u(\boldsymbol{U}^{n,i}, \boldsymbol{V}^{n,i}, t_n + c_i^u h), \quad \boldsymbol{\ell}^{n,i} = f^v(\boldsymbol{U}^{n,i}, \boldsymbol{V}^{n,i}, t_n + c_i^v h), \tag{5.37}$$

$$\boldsymbol{U}^{n,i} = \boldsymbol{u}^n + h\sum_{j=1}^s a_{ij}^u \boldsymbol{\kappa}^{n,j}, \qquad \boldsymbol{V}^{n,i} = \boldsymbol{v}^n + h\sum_{j=1}^s a_{ij}^v \boldsymbol{\ell}^{n,j}. \tag{5.38}$$

Here, $s \geq 1$ is the number of stages. The stage variables $\boldsymbol{U}^{n,i}$ and $\boldsymbol{V}^{n,i}$ are set in a bold font to indicate that they are unrelated to the solution operator matrix $U(t, \boldsymbol{\alpha})$ and the target gate matrix V_{tg}.

The Störmer-Verlet scheme is a two-stage PRK method ($s = 2$) that is symplectic, time-reversible and second order accurate [67]. It combines the trapezoidal and the implicit midpoint rules, with Butcher coefficients:

$$a_{11}^u = a_{12}^u = 0, \quad a_{21}^u = a_{22}^u = \frac{1}{2}, \qquad a_{11}^v = a_{21}^v = \frac{1}{2}, \quad a_{12}^v = a_{22}^v = 0, \tag{5.39}$$

$$b_1^u = b_2^u = \frac{1}{2}, \quad c_1^u = 0, \quad c_2^u = 1, \qquad b_1^v = b_2^v = \frac{1}{2}, \quad c_1^v = c_2^v = \frac{1}{2}. \tag{5.40}$$

5.3.2 Time step restrictions for accuracy and stability

The accuracy in the numerical solution of Schrödinger's equation is essentially determined by how well the fastest time scale in the state vector is resolved on the grid in time. The analysis of the time scales in the solution of Schrödinger's equation is most straightforward to perform in the complex-valued formulation (5.1).

There are two fundamental time scales that must be resolved in the solution of Schrödinger's equation. The first corresponds to how quickly the control functions must vary in time to trigger the desired transitions between the energy levels in the quantum system. This time scale is determined by the transition frequencies in the system Hamiltonian, which follow as the difference between its consecutive eigenvalues. In the Hamiltonian model (5.15) and (5.16), the angular transition frequencies between the essential energy levels (with detuning frequency Δ_1) are

$$\Omega_{1,n} = \Delta_1 - n\xi_1, \quad n = 0, \ldots, N_f - 1.$$

The second time scale is due to the harmonic oscillation of the phase in the state vector. It can be estimated by freezing the time-dependent coefficients in the Hamiltonian matrix at some time $t = t_*$ and considering Schrödinger's equation with the time-independent Hamiltonian matrix $H_* =$

$H(t_*)$. The $N \times N$ matrix H_* is Hermitian and can be diagonalized by a unitary transformation,

$$H_* X = X\Gamma, \quad X^\dagger X = I_N, \quad \Gamma = \mathrm{diag}(\gamma_1, \gamma_2, \ldots, \gamma_N),$$

where the eigenvalues γ_k are real. By the change of variables $\widetilde{\boldsymbol{\psi}} = X^\dagger \boldsymbol{\psi}$, the solution of the diagonalized system follows as

$$\widetilde{\psi}_k(t) = e^{-i\gamma_k t}\widetilde{\psi}_k(0),$$

corresponding to the period $\tau_k = 2\pi/|\gamma_k|$. The shortest period thus follows from the spectral radius of H_*, $\rho(H_*) = \max_k |\gamma_k|$.

To estimate the time step for the Störmer-Verlet method, we require that the shortest period in the solution of Schrödinger's equation must be resolved by at least C_P time steps. Taking both time scales into account leads to the time step restriction

$$h \leq \frac{2\pi}{C_P \max\{\rho(H_*), \max_n(|\Omega_{1,n}|)\}}. \tag{5.41}$$

The value of C_P that is needed to obtain a given accuracy in the numerical solution depends on the order of accuracy, the duration of the time integration, as well as the details of the time-stepping scheme. For second order accurate methods such as the Störmer-Verlet method, acceptable accuracy for engineering applications can often achieved with $C_P \approx 40$. With the Störmer-Verlet method, we note that the time-stepping can become unstable if $C_P \leq 2$, corresponding to a sampling rate below the Nyquist limit.

After freezing the coefficients, the Hamiltonian (5.15) and (5.16) becomes

$$H_* = -\frac{\xi_a}{2} a^\dagger a^\dagger a a + p_*(a + a^\dagger) + i q_*(a - a^\dagger), \quad p_* = p(t_*, \boldsymbol{\alpha}), \quad q_* = q(t_*, \boldsymbol{\alpha}).$$

We can estimate the spectral radius of $H_* \in \mathbb{C}^{N\times N}$ using the Gershgorin circle theorem [59]. Because H_* is Hermitian, all its eigenvalues are real. As a result, its spectral radius can be bounded by

$$\rho(H_*) \leq \frac{|\xi_a|}{2}(N-1)(N-2) + 2d_\infty\sqrt{N-1}.$$

Here we have used that the control function is bounded by $d_\infty = \max_t |d_1(t, \boldsymbol{\alpha})|$ for a given parameter vector $\boldsymbol{\alpha}$, in the interval $0 \leq t \leq T$. With this estimate in (5.41) we guarantee that the time-dependent phase in the state vector is resolved by at at least C_P time steps per shortest period.

If the optimization imposes amplitude constraints on the parameter vector, $|\boldsymbol{\alpha}|_\infty \leq \alpha_{max}$, those constraints can be used to estimate the time step before the optimization starts. This allows the same time step to be used throughout the iteration and eliminates the need to recalculate the spectral radius of H_* when $\boldsymbol{\alpha}$ changes.

Our implementation of the Störmer-Verlet scheme was verified to be second order accurate. It was also found to give approximately the same accuracy as the second order Magnus integrator [67] when the same time step was used in both methods.

5.4 Discretizing the objective function and its gradient

In this section, we develop a "discretize-then-optimize" approach in which we first discretize the objective function and then derive a compatible scheme for discretizing the adjoint state equation, which is used for computing the gradient of the objective function. As was outlined in the introduction, our approach builds upon the works of Hager [65], Sanz-Serna [97] and Ober-Blöbaum [92].

5.4.1 Discretizing the objective function

The Störmer-Verlet scheme can be written in terms of the stage variables $(\boldsymbol{U}^{n,i}, \boldsymbol{V}^{n,i})$ by substituting $(\boldsymbol{\kappa}^{n,i}, \boldsymbol{\ell}^{n,i})$ from (5.37) into (5.36),

$$\boldsymbol{u}^0 = \boldsymbol{g}^u, \quad \boldsymbol{v}^0 = \boldsymbol{g}^v, \tag{5.42}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \frac{h}{2}\left(S_n \boldsymbol{U}^{n,1} + S_{n+1}\boldsymbol{U}^{n,2} - K_n \boldsymbol{V}^{n,1} - K_{n+1}\boldsymbol{V}^{n,2}\right), \tag{5.43}$$

$$\boldsymbol{v}^{n+1} = \boldsymbol{v}^n + \frac{h}{2}\left(K_{n+1/2}\left(\boldsymbol{U}^{n,1} + \boldsymbol{U}^{n,2}\right) + S_{n+1/2}(\boldsymbol{V}^{n,1} + \boldsymbol{V}^{n,2})\right), \tag{5.44}$$

and into (5.38),

$$\boldsymbol{U}^{n,1} = \boldsymbol{u}^n, \tag{5.45}$$

$$\boldsymbol{U}^{n,2} = \boldsymbol{u}^n + \frac{h}{2}\left(S_n\boldsymbol{U}^{n,1} + S_{n+1}\boldsymbol{U}^{n,2} - K_n\boldsymbol{V}^{n,1} - K_{n+1}\boldsymbol{V}^{n,2}\right), \tag{5.46}$$

$$\boldsymbol{V}^{n,1} = \boldsymbol{v}^n + \frac{h}{2}\left(K_{n+1/2}\boldsymbol{U}^{n,1} + S_{n+1/2}\boldsymbol{V}^{n,1}\right), \tag{5.47}$$

$$\boldsymbol{V}^{n,2} = \boldsymbol{v}^n + \frac{h}{2}\left(K_{n+1/2}\boldsymbol{U}^{n,1} + S_{n+1/2}\boldsymbol{V}^{n,1}\right). \tag{5.48}$$

Here, $S_n = S(t_n)$, $S_{n+1/2} = S(t_n + 0.5h)$, etc. Because $S(t) \neq 0$, the scheme is block implicit. Note that $\boldsymbol{u}^{n+1} = \boldsymbol{U}^{n,2}$ and $\boldsymbol{V}^{n,1} = \boldsymbol{V}^{n,2} = \boldsymbol{v}(t_{n+1/2}) + \mathcal{O}(h^2)$.

The numerical solution at the final time step provides a second order accurate approximation of the continuous solution operator matrix U_T, which we denote U_{Th}. It is used to approximate the matrix overlap function O_T in (5.4),

$$O_{Vh}(U_{Th}) = \sum_{j=0}^{E-1}\left(\left\langle \boldsymbol{u}_j^M, \boldsymbol{d}_j^u\right\rangle_2 + \left\langle \boldsymbol{v}_j^M, \boldsymbol{d}_j^v\right\rangle_2\right) + i\sum_{j=0}^{E-1}\left(\left\langle \boldsymbol{v}_j^M, \boldsymbol{d}_j^u\right\rangle_2 - \left\langle \boldsymbol{u}_j^M, \boldsymbol{d}_j^v\right\rangle_2\right), \tag{5.49}$$

which is then used as the first part of the discrete objective function,

$$\mathcal{J}_1^h(U_{Th}) = \left(1 - \frac{1}{E^2}|O_{Vh}(U_{Th})|^2\right). \tag{5.50}$$

The integral in the objective function (5.6) can be discretized to second order accuracy by using the Runge-Kutta stage variables,

$$\mathcal{J}_2^h(\boldsymbol{U},\boldsymbol{V}) = \frac{h}{T}\sum_{j=0}^{E-1}\sum_{n=0}^{M-1}\left(\frac{1}{2}\left\langle \boldsymbol{U}_j^{n,1}, W\boldsymbol{U}_j^{n,1}\right\rangle_2 + \frac{1}{2}\left\langle \boldsymbol{U}_j^{n,2}, W\boldsymbol{U}_j^{n,2}\right\rangle_2 + \left\langle \boldsymbol{V}_j^{n,1}, W\boldsymbol{V}_j^{n,1}\right\rangle_2\right). \tag{5.51}$$

Based on the above formulas we discretize the objective function (5.7) according to

$$\mathcal{G}^h(\boldsymbol{\alpha}) = \mathcal{J}^h(U_{Th}^{\alpha}, \boldsymbol{U}^{\alpha}, \boldsymbol{V}^{\alpha}), \quad \mathcal{J}^h(U_{Th}, \boldsymbol{U}, \boldsymbol{V}) := \mathcal{J}_1^h(U_{Th}) + \mathcal{J}_2^h(\boldsymbol{U}, \boldsymbol{V}). \tag{5.52}$$

Here, U_{Th}^{α}, $\boldsymbol{U}^{\alpha}$ and $\boldsymbol{V}^{\alpha}$ represent the time-discrete solution of the Störmer-Verlet scheme for a given parameter vector $\boldsymbol{\alpha}$. We note that $\mathcal{G}^h(\boldsymbol{\alpha})$ can be evaluated by accumulation during the time-stepping of the Störmer-Verlet scheme.

5.4.2 The discrete adjoint approach

The gradient of the discretized objective function can be derived from first order optimality conditions of the corresponding discrete Lagrangian. In this approach, let $(\boldsymbol{\mu}_j^n, \boldsymbol{\nu}_j^n)$ be the adjoint variables and let $(\boldsymbol{M}_j^{n,i}, \boldsymbol{N}_j^{n,i})$ be Lagrange multipliers. We define the discrete Lagrangian by

$$\mathcal{L}^h(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\mu}, \boldsymbol{\nu}, \boldsymbol{M}, \boldsymbol{N}, \boldsymbol{\alpha}) = \mathcal{J}^h(U_{Th}, \boldsymbol{U}, \boldsymbol{V}) - \sum_{j=0}^{E-1} \left(\left\langle \boldsymbol{u}_j^0 - \boldsymbol{g}_j^u, \boldsymbol{\mu}_j^0 \right\rangle_2 + \left\langle \boldsymbol{v}_j^0 - \boldsymbol{g}_j^v, \boldsymbol{\nu}_j^0 \right\rangle_2 + \sum_{k=1}^{6} T_j^k \right). \quad (5.53)$$

The first two terms in the sum enforce the initial conditions (5.42). The terms T_j^1 and T_j^2 enforce the time-stepping update formulas (5.43)-(5.44) in the Störmer-Verlet scheme,

$$T_j^1 = \sum_{n=0}^{M-1} \left\langle \boldsymbol{u}_j^{n+1} - \boldsymbol{u}_j^n - \frac{h}{2} \left(S_n \boldsymbol{U}_j^{n,1} + S_{n+1} \boldsymbol{U}_j^{n,2} - K_n \boldsymbol{V}_j^{n,1} - K_{n+1} \boldsymbol{V}_j^{n,2} \right), \boldsymbol{\mu}_j^{n+1} \right\rangle_2, \quad (5.54)$$

$$T_j^2 = \sum_{n=0}^{M-1} \left\langle \boldsymbol{v}_j^{n+1} - \boldsymbol{v}_j^n - \frac{h}{2} \left(K_{n+1/2} \left(\boldsymbol{U}_j^{n,1} + \boldsymbol{U}_j^{n,2} \right) + S_{n+1/2} (\boldsymbol{V}_j^{n,1} + \boldsymbol{V}_j^{n,2}) \right), \boldsymbol{\nu}_j^{n+1} \right\rangle_2. \quad (5.55)$$

The terms T_j^3 to T_j^6 enforce the relations between the stage variables (5.45)-(5.48) using the Lagrange multipliers ($\boldsymbol{M}_j^{n,i}$ and $\boldsymbol{N}_j^{n,i}$), see Appendix .15 for details.

To derive the discrete adjoint scheme, we note that the discrete Lagrangian (5.53) has a saddle point if

$$\frac{\partial \mathcal{L}^h}{\partial \boldsymbol{\mu}_j^n} = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{\nu}_j^n} = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{N}_j^{n,i}} = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{M}_j^{n,i}} = 0, \quad (5.56)$$

$$\frac{\partial \mathcal{L}^h}{\partial \boldsymbol{u}_j^n} = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{v}_j^n} = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{U}_j^{n,i}} = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{V}_j^{n,i}} = 0, \quad (5.57)$$

for $n = 0, 1, \ldots, M$, $i = 1, 2$ and $j = 0, 1, \ldots, E-1$. Here, the set of conditions in (5.56) result in the Störmer-Verlet scheme (5.42)-(5.48) for evolving $(\boldsymbol{u}_j^n, \boldsymbol{v}_j^n, \boldsymbol{U}_j^{n,i}, \boldsymbol{V}_j^{n,i})$ forwards in time. The set of conditions in (5.57) result in a time-stepping scheme for evolving the adjoint variables $(\boldsymbol{\mu}_j^n, \boldsymbol{\nu}_j^n)$ backwards in time, as is made precise in the following lemma.

Lemma 5.4.1. *Let $\mathcal{L}^h$ be the discrete Lagrangian defined by* (5.53)*. Furthermore, let $(\boldsymbol{u}_j^n, \boldsymbol{v}_j^n, \boldsymbol{U}_j^{n,i}, \boldsymbol{V}_j^{n,i})$ satisfy the Störmer-Verlet scheme* (5.42)-(5.48) *for a given parameter vector $\boldsymbol{\alpha}$. Then, the set of*

saddle-point conditions (5.57) *are satisfied if the Lagrange multipliers* $(\boldsymbol{\mu}_j^n, \boldsymbol{\nu}_j^n)$ *are calculated according to the reversed time-stepping scheme,*

$$\boldsymbol{\mu}_j^M = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{u}_j^M}, \quad \boldsymbol{\nu}_j^M = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{v}_j^M}, \tag{5.58}$$

$$\boldsymbol{\mu}_j^n = \boldsymbol{\mu}_j^{n+1} - \frac{h}{2}\left(\boldsymbol{\kappa}_j^{n,1} + \boldsymbol{\kappa}_j^{n,2}\right), \tag{5.59}$$

$$\boldsymbol{\nu}_j^n = \boldsymbol{\nu}_j^{n+1} - \frac{h}{2}\left(\boldsymbol{\ell}_j^{n,1} + \boldsymbol{\ell}_j^{n,2}\right), \tag{5.60}$$

for $n = M-1, M-2, \ldots 0$. *Because* $S^T = -S$ *and* $K^T = K$, *the slopes satisfy*

$$\boldsymbol{\kappa}_j^{n,1} = S_n \boldsymbol{X}_j^n - K_{n+1/2}\boldsymbol{Y}_j^{n,1} - \frac{2}{h}\frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}}, \tag{5.61}$$

$$\boldsymbol{\kappa}_j^{n,2} = S_{n+1} \boldsymbol{X}_j^n - K_{n+1/2}\boldsymbol{Y}_j^{n,2} - \frac{2}{h}\frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}}, \tag{5.62}$$

$$\boldsymbol{\ell}_j^{n,1} = K_n \boldsymbol{X}_j^n + S_{n+1/2}\boldsymbol{Y}_j^{n,1} - \frac{2}{h}\frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}}, \tag{5.63}$$

$$\boldsymbol{\ell}_j^{n,2} = K_{n+1} \boldsymbol{X}_j^n + S_{n+1/2}\boldsymbol{Y}_j^{n,2} - \frac{2}{h}\frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}}, \tag{5.64}$$

where the stage variables are given by

$$\boldsymbol{X}_j^n = \boldsymbol{\mu}_j^{n+1} - \frac{h}{2}\boldsymbol{\kappa}_j^{n,2}, \tag{5.65}$$

$$\boldsymbol{Y}_j^{n,2} = \boldsymbol{\nu}_j^{n+1}, \tag{5.66}$$

$$\boldsymbol{Y}_j^{n,1} = \boldsymbol{\nu}_j^{n+1} - \frac{h}{2}\left(\boldsymbol{\ell}_j^{n,1} + \boldsymbol{\ell}_j^{n,2}\right). \tag{5.67}$$

Proof. The lemma follows after a somewhat tedious but straightforward calculation shown in detail in Appendix .15. □

Corresponding to the continuous Schrödinger equation (5.30), the adjoint state equation (without forcing) is

$$\begin{bmatrix} \dot{\boldsymbol{\mu}} \\ \dot{\boldsymbol{\nu}} \end{bmatrix} = \begin{bmatrix} S(t) & -K(t) \\ K(t) & S(t) \end{bmatrix} \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\nu} \end{bmatrix} =: \begin{bmatrix} \boldsymbol{f}^{\mu}(\boldsymbol{\mu}, \boldsymbol{\nu}, t) \\ \boldsymbol{f}^{\nu}(\boldsymbol{\mu}, \boldsymbol{\nu}, t) \end{bmatrix}, \tag{5.68}$$

where we used that $S^T = -S$ and $K^T = K$.

Corollary 5.4.1.1. *The time-stepping scheme* (5.59)-(5.67) *(without forcing) is a consistent approximation of the continuous adjoint state equation* (5.68)*. It can be written as a modified partitioned Runge-Kutta method, where the Butcher coefficients are*

$$a^{\mu}_{11} = a^{\mu}_{21} = 1/2, \quad a^{\mu}_{12} = a^{\mu}_{22} = 0, \qquad a^{\nu}_{11} = a^{\nu}_{12} = 0, \quad a^{\nu}_{21} = a^{\nu}_{22} = 1/2, \tag{5.69}$$

$$b^{\mu}_1 = b^{\mu}_2 = \frac{1}{2}, \qquad b^{\nu}_1 = b^{\nu}_2 = \frac{1}{2}, \tag{5.70}$$

corresponding to the implicit midpoint rule for the $\boldsymbol{\mu}$*-equation and the trapezoidal rule for the* $\boldsymbol{\nu}$*-equation in* (5.68)*. The modifications to the partitioned Runge-Kutta scheme concerns the formulae for the slopes,* (5.61)-(5.64)*. Because of the time-levels at which the matrices* K *and* S *are evaluated, it is* ***not*** *possible to define Butcher coefficients* c^{μ}_i *and* c^{ν}_i *such that*

$$\boldsymbol{\kappa}^{n,i}_j = \boldsymbol{f}^{\mu}(\boldsymbol{X}^{n,i}_j, \boldsymbol{Y}^{n,i}_j, t_n + c^{\mu}_i h),$$
$$\boldsymbol{\ell}^{n,i}_j = \boldsymbol{f}^{\nu}(\boldsymbol{X}^{n,i}_j, \boldsymbol{Y}^{n,i}_j, t_n + c^{\nu}_i h).$$

Proof. See Appendix .16. □

Only the matrices K and S depend explicitly on $\boldsymbol{\alpha}$ in the discrete Lagrangian. When the saddle point conditions (5.56) and (5.57) are satisfied, we can therefore calculate the gradient of $\mathcal{G}^h$ by differentiating (5.53),

$$\frac{\partial \mathcal{G}^h}{\partial \alpha_r} = \frac{\partial \mathcal{L}^h}{\partial \alpha_r}, \quad r = 0, 1, \ldots, E-1.$$

This relation leads to the following lemma.

Lemma 5.4.2. *Let* $\mathcal{L}^h$ *be the discrete Lagrangian defined by* (5.53)*. Assume that* $(\boldsymbol{u}^n_j, \boldsymbol{v}^n_j, \boldsymbol{U}^{n,i}_j, \boldsymbol{V}^{n,i}_j)$ *are calculated according to the Störmer-Verlet scheme for a given parameter vector* $\boldsymbol{\alpha}$*. Furthermore, assume that* $(\boldsymbol{\mu}^n_j, \boldsymbol{\nu}^n_j, \boldsymbol{X}^n_j, \boldsymbol{Y}^{n,i}_j)$ *satisfy the adjoint time-stepping scheme in Lemma 5.4.1, subject to the terminal conditions*

$$\boldsymbol{\mu}^M_j = -\frac{2}{E^2}\left(Re(O_{Vh})\boldsymbol{d}^u_j - Im(O_{Vh})\boldsymbol{d}^v_j\right), \quad \boldsymbol{\nu}^M_j = -\frac{2}{E^2}\left(Re(O_{Vh})\boldsymbol{d}^v_j + Im(O_{Vh})\boldsymbol{d}^u_j\right),$$

and the forcing functions

$$\frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}} = \frac{h}{T} W \boldsymbol{U}_j^{n,1}, \quad \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}} = \frac{h}{T} W \boldsymbol{U}_j^{n,2}$$
$$\frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}} = \frac{h}{T} W \boldsymbol{V}_j^{n,1}, \quad \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}} = 0.$$

Then, the saddle-point conditions (5.56) *and* (5.57) *are satisfied and the gradient of the objective function* (5.52) *is given by*

$$\frac{\partial \mathcal{G}^h}{\partial \alpha_r} = \frac{h}{2} \sum_{j=0}^{E-1} \sum_{n=0}^{M-1} \Big\{ \Big\langle S_n' \boldsymbol{U}_j^{n,1} + S_{n+1}' \boldsymbol{U}_j^{n,2} - (K_n' + K_{n+1}') \boldsymbol{V}_j^{n,1}, \boldsymbol{X}_j^n \Big\rangle_2 + \Big\langle K_{n+1/2}' \boldsymbol{U}_j^{n,1} + S_{n+1/2}' \boldsymbol{V}_j^{n,1}, \boldsymbol{Y}_j^{n,1} \Big\rangle_2 + \Big\langle K_{n+1/2}' \boldsymbol{U}_j^{n,2} + S_{n+1/2}' \boldsymbol{V}_j^{n,1}, \boldsymbol{Y}_j^{n,2} \Big\rangle_2 \Big\}, \tag{5.71}$$

where $S_n' = \partial S / \partial \alpha_r(t_n)$, $K_{n+1/2}' = \partial K / \partial \alpha_r(t_{n+1/2})$, *etc.*

Proof. See Appendix .17. □

As a result of Lemma 5.4.2, all components of the gradient can be calculated from $(\boldsymbol{u}_j^n, \boldsymbol{v}_j^n, \boldsymbol{U}_j^{n,i}, \boldsymbol{V}_j^{n,1})$ and the adjoint variables $(\boldsymbol{\mu}_j^n, \boldsymbol{\nu}_j^n, \boldsymbol{X}_j^n, \boldsymbol{Y}_j^{n,i})$. The first set of variables are obtained from time-stepping the Störmer-Verlet scheme forward in time, while the second set of variables follow from time-stepping the adjoint scheme backward in time.

We can avoid storing the time-history of $(\boldsymbol{u}_j^n, \boldsymbol{v}_j^n, \boldsymbol{U}_j^{n,i}, \boldsymbol{V}_j^{n,1})$ by using the time-reversibility of the Störmer-Verlet scheme. However, in order to do so, we must first calculate the terminal conditions $(\boldsymbol{u}_j^M, \boldsymbol{v}_j^M)$ by evolving (5.42)-(5.48) forwards in time. The time-stepping can then be reversed and the gradient of the objective function (5.71) can be accumulated by simultaneously time-stepping the adjoint system (5.59)-(5.67) backwards in time.

5.5 Numerical optimization

Our numerical solution of the optimal control problem is based on the general purpose interior-point optimization package IPOPT [106]. This open-source library implements a primal-dual barrier approach for solving large-scale nonlinear programming problems, i.e., it minimizes an objective function subject to inequality (barrier) constraints on the parameter vector. Because

the Hessian of the objective function is costly to calculate, we use the L-BFGS algorithm [91] in IPOPT, which only relies on the objective function and its gradient to be evaluated. Inequality constraints that limit the amplitude of the parameter vector $\boldsymbol{\alpha}$ are enforced internally by IPOPT.

The routines for evaluating the objective function and its gradient are implemented in the Julia programming language [24], which provides a convenient interface to IPOPT. Given a parameter vector $\boldsymbol{\alpha}$, the routine for evaluating the objective function solves the Schrödinger equation with the Störmer-Verlet scheme and evaluates $\mathcal{G}^h(\boldsymbol{\alpha})$ by accumulation. The routine for evaluating the gradient first applies the Störmer-Verlet scheme to calculate terminal conditions for the state variables. It then proceeds by accumulating the gradient $\nabla_\alpha \mathcal{G}^h$ by simultaneous reversed time-stepping of the discrete adjoint scheme and the Störmer-Verlet scheme. These two fundamental routines, together with functions for setting up the Hamiltonians, estimating the time step, setting up constraints on the parameter vector, post-processing and plotting of the results have been implemented in the software package Juqbox, which was used to generate the numerical results below.

The adjoint gradient implementation has been verified against a centered finite difference approximation of the discrete objective function by perturbing each component of the parameter vector. To further verify our implementation, we also calculated the discrete gradient by differentiating the Störmer-Verlet scheme with respect to each component of the parameter vector. This gradient agreed with the adjoint gradient to within 11-12 digits.

5.5.1 A CNOT gate on a single qudit with guard levels

To test our methods on a quantum optimal control problem, we consider realizing a CNOT gate on a single qudit with four essential energy levels and two guard levels. The qudit is modeled in the rrotating frame of reference (with detuning frequencies $\Delta_1 = \Delta_2 = 0$) using the system and control Hamiltonians (5.15) and (5.16), respectively. We set the fundamental frequency $\omega_1/2\pi =$ 4.10336 GHz and self-Kerr coefficient $\xi_1/2\pi = 0.2198$ GHz. We parameterize the two control functions using B-splines with carrier waves and choose the frequencies to be $\Omega_1 = 0$, $\Omega_2 = -\xi_1$ and $\Omega_3 = -2\xi_1$. In the rotating frame, these frequencies correspond to transitions between the

ground state and the first exited state, the first and second excited states and the second and third excited states. We discourage population of the fourth and fifth excited states using the weight matrix $W = \mathrm{diag}[0, 0, 0, 0, 0.1, 1.0]$ in $\mathcal{J}_2^h$, see (5.6). We use $D_1 = 10$ basis functions per frequency and control function, resulting in a total of $D = 60$ parameters. The amplitudes of the control functions are limited by the constraint

$$\|\boldsymbol{\alpha}\|_\infty := \max_{1 \le r \le D} |\alpha_r| \le \alpha_{max}. \tag{5.72}$$

We set the gate duration to $T = 100$ ns and estimate the time step using the technique in Section 5.3.2. To guarantee at least $C_P = 40$ time steps per period, we use $M = 8,796$ time steps, corresponding to $h \approx 1.136 \cdot 10^{-2}$ ns.

As initial guess for the elements of the parameter vector, we use a random number generator with a uniform distribution in $[-0.01, 0.01]$. In Figure 5.2 we present the convergence history with the two parameter thresholds $\alpha_{max}/2\pi = 4$ MHz and 3 MHz, respectively. We show the objective function $\mathcal{G}$, decomposed into $\mathcal{J}_1^h$ and $\mathcal{J}_2^h$, together with the norm of the dual infeasibility, $\|\nabla_\alpha \mathcal{G} - z\|_\infty$, that IPOPT uses to monitor convergence, see [106] for details. For the case with $\alpha_{max}/2\pi = 3$ MHz, IPOPT converges well and needs 126 iteration to reduce the dual infeasibility to 10^{-5}, which was used as convergence criteria. However, when the parameter constraint is relaxed to $\alpha_{max}/2\pi = 4$ MHz, the convergence of IPOPT stalls after about 100 iterations and is terminated after 200 iterations.

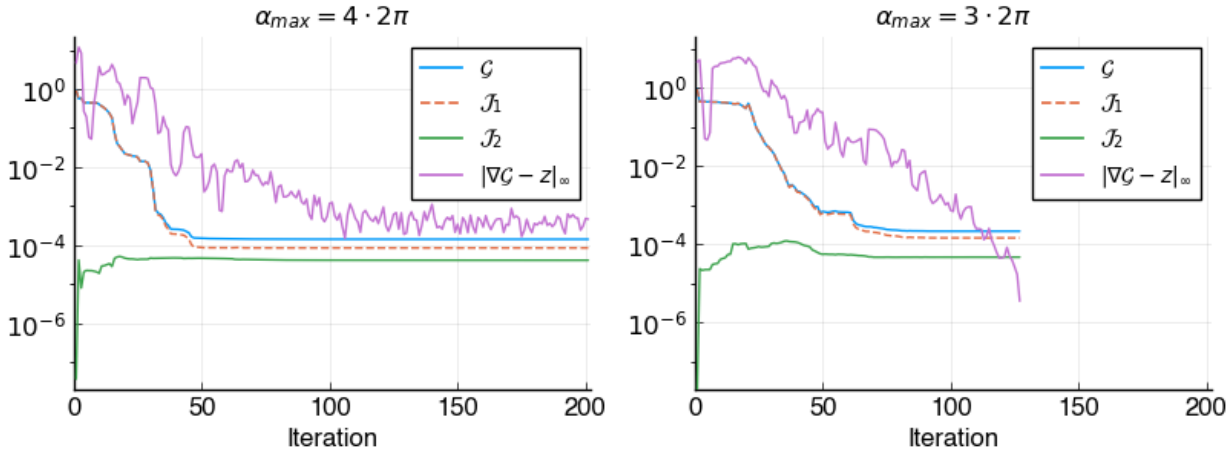


Figure 5.2: Convergence of the IPOPT iteration for the CNOT gate with the parameter constraint $\|\boldsymbol{\alpha}\|_\infty \leq \alpha_{max}$. Here, $\alpha_{max}/2\pi = 4$ MHz (left) and $\alpha_{max}/2\pi = 3$ MHz (right).

For the converged solution with parameter constraint $\alpha_{max}/2\pi = 3$ MHz, the two parts of the objective function are $\mathcal{J}_1^h \approx 1.47 \cdot 10^{-4}$ and $\mathcal{J}_2^h \approx 4.72 \cdot 10^{-5}$, corresponding to a trace fidelity greater than 0.9998. The population of the guard states remains small for all times and initial conditions. In particular, the "forbidden" state $|5\rangle$ has a population that remains below $4.04 \cdot 10^{-7}$, see Figure 5.3.

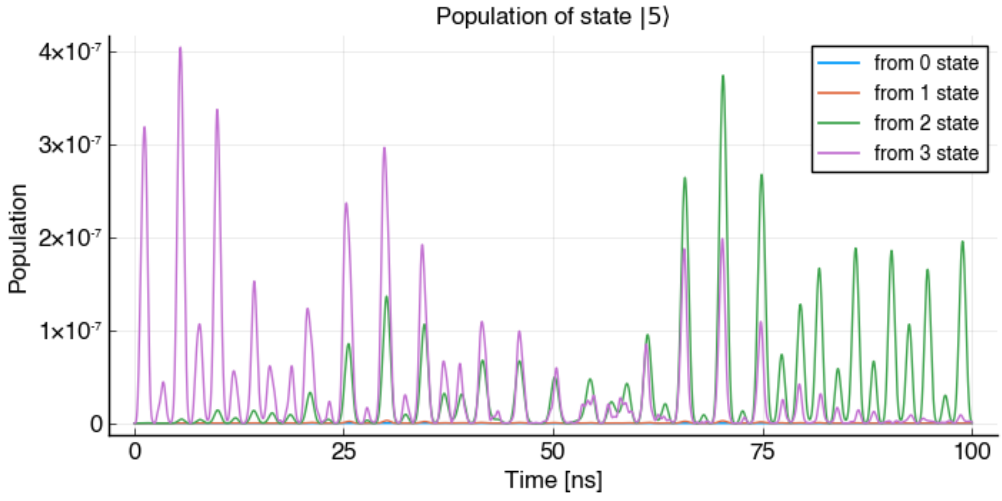


Figure 5.3: The population of the "forbidden" state $|5\rangle$ as function of time for the four initial conditions of the CNOT gate. Here, $\alpha_{max}/2\pi = 3$ MHz.

The optimized control functions are shown in Figure 5.4 and the population of the essential states, corresponding to the four initial conditions of the CNOT gate, are presented in Figure 5.5.

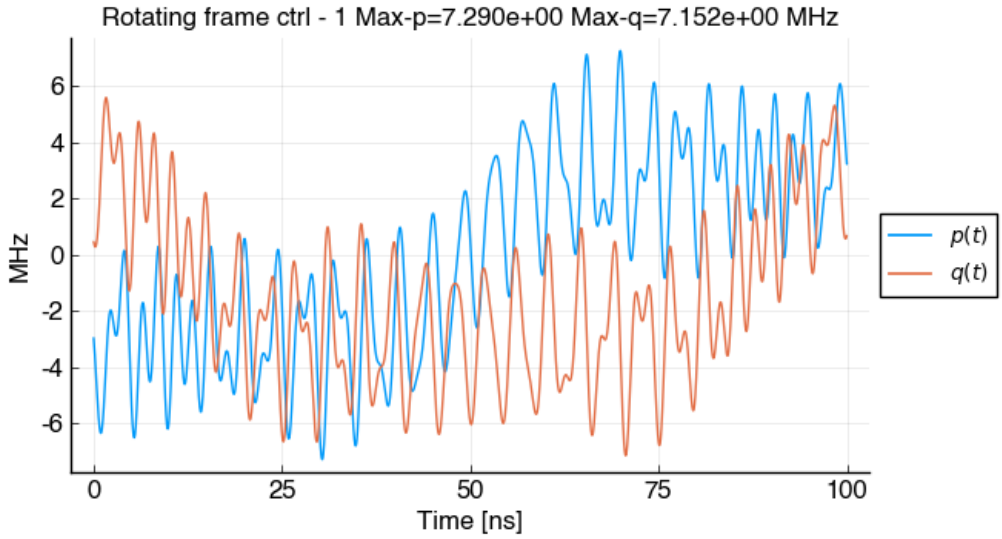


Figure 5.4: The rotating frame control functions $p(t)$ (blue) and $q(t)$ (orange) for realizing a CNOT gate with $D_1 = 10$ basis function per carrier wave and three carrier wave frequencies. Here, $\alpha_{max}/2\pi = 3$ MHz.

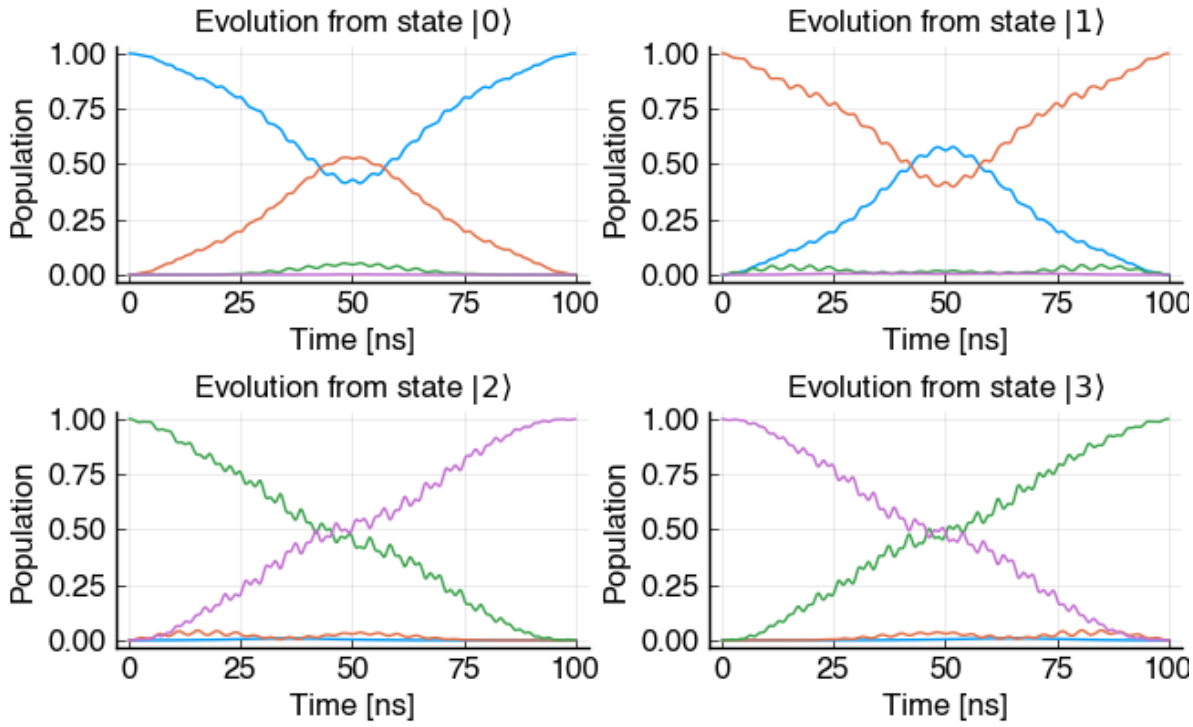


Figure 5.5: The population of the states $|0\rangle$ (blue), $|1\rangle$ (orange), $|2\rangle$ (green) and $|3\rangle$ (purple), as function of time, for each initial condition of the CNOT gate. Here, $\alpha_{max}/2\pi = 3$ MHz.

Even though the dual infidelity does not reach the convergence criteria with the parameter threshold $\alpha_{max}/2\pi = 4$ MHz, the resulting control functions give a very small objective function. Here, $\mathcal{J}_1^h \approx 8.56 \cdot 10^{-5}$ and $\mathcal{J}_2^h \approx 4.15 \cdot 10^{-5}$, corresponding to a trace fidelity greater than 0.9999. The population of the "forbidden" state $|5\rangle$ has a population that remains below $3.39 \cdot 10^{-7}$.

5.5.2 The Hessian of the objective function

The numerical results shown in Figure 5.2 illustrate that the convergence properties of the optimization algorithm depend on the parameter constraints. To gain clarity into the local landscape of the optima we study the Hessian of the objective function. Let the optima correspond to the parameter vector $\boldsymbol{\alpha}^*$. Based on the adjoint scheme for calculating the gradient, we can approximate the elements of the Hessian matrix using a centered finite difference approximation,

$$\frac{\partial^2 \mathcal{G}^h(\boldsymbol{\alpha}^*)}{\partial \alpha_j \partial \alpha_k} \approx \frac{1}{2\varepsilon}\left(\frac{\partial \mathcal{G}^h}{\partial \alpha_j}(\boldsymbol{\alpha}^* + \varepsilon \boldsymbol{e}_k) - \frac{\partial \mathcal{G}^h}{\partial \alpha_j}(\boldsymbol{\alpha}^* - \varepsilon \boldsymbol{e}_k)\right) := L_{j,k}, \tag{5.73}$$

ε	$\|0.5(L + L^T)\|_F$	$\|0.5(L - L^T)\|_F$
10^{-4}	$4.95 \cdot 10^3$	$1.99 \cdot 10^{-4}$
10^{-5}	$4.95 \cdot 10^3$	$2.01 \cdot 10^{-6}$
10^{-6}	$4.95 \cdot 10^3$	$1.46 \cdot 10^{-6}$
10^{-7}	$4.95 \cdot 10^3$	$1.47 \cdot 10^{-5}$

Table 5.1: The Frobenius norm of the symmetric and asymmetric parts of the approximate Hessian, L, for the case $\alpha_{max}/2\pi = 3.0$ MHz.

for $j, k = 1, 2, \ldots, D$. To perform this calculation, the gradient must be evaluated for the $2D$ parameter vectors ($\boldsymbol{\alpha}^* \pm \varepsilon \boldsymbol{e}_k$). Because the objective function and the parameter vector are real-valued, the gradient and the Hessian are also real-valued. Due to the finite difference approximation, the matrix L is only approximately equal to the Hessian. The accuracy in L is estimated in Table 5.1 by studying the norm of its asymmetric part, which is zero for the Hessian. Based on this experiment we infer that $\varepsilon = 10^{-6}$ is appropriate to use for approximating the Hessian in (5.73). To eliminate spurious effects from the asymmetry in the L matrix, we study the spectrum of its symmetric part, $L_s = 0.5(L + L^T)$. Because it is real and symmetric, it has a complete set of eigenvectors and all eigenvalues are real.

The eigenvalues of the Hessian are shown in Figure 5.6 for both values of the parameter threshold, α_{max}.

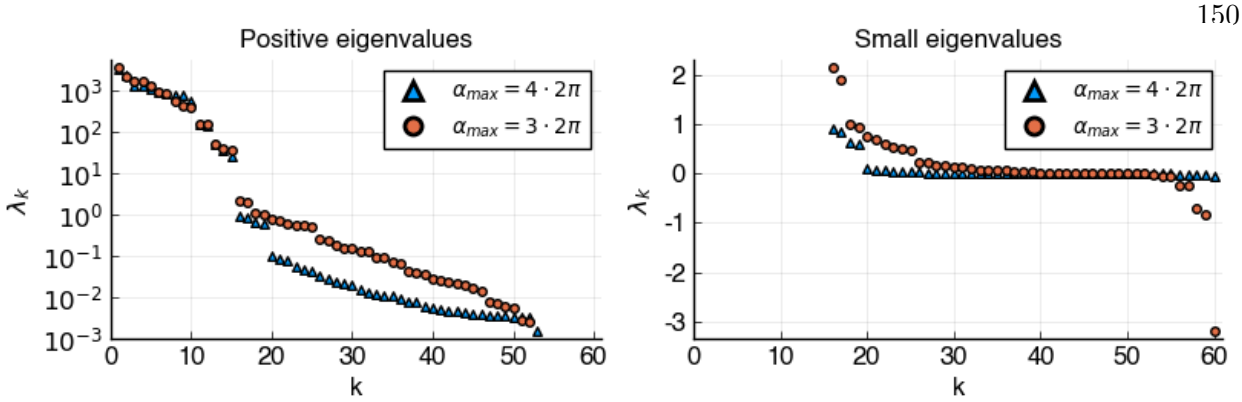


Figure 5.6: The eigenvalues of the symmetric part of the approximate Hessian, $0.5(L + L^T)$, evaluated at the optima for the parameter thresholds $\alpha_{max}/2\pi = 4$ MHz (blue triangles) and $\alpha_{max}/2\pi = 3$ MHz (orange circles). The positive eigenvalues are shown on a log-scale on the left and the small eigenvalues are shown on a linear scale on the right.

Two properties of the spectra are noteworthy. First, a few eigenvalues are negative. This may be an artifact related to the elements of the parameter vector that are close to their bounds. As a result the landscape of the objective function may not be accurately represented by the corresponding components of the Hessian. The second interesting property is that the 15 largest eigenvalues are significantly larger than the rest. This indicates that the control functions are essentially described by the 15 eigenvectors associated with those eigenvalues. As a result, the objective function varies much faster in those directions than in the directions of the remaining 45 eigenvectors and this may hamper the convergence of the optimization algorithm in that subspace. However, most of those 45 eigenvalues become larger when the parameter threshold is reduced from $\alpha_{max}/2\pi = 4$ MHz to $\alpha_{max}/2\pi = 3$ MHz. This indicates that the constraints on the parameter vector have a regularizing effect on the optimization problem and may explain why the latter case converges better (see Figure 5.2).

5.5.3 Risk-neutral controls

In practice the entries of the Hamiltonian may have some uncertainty, especially for higher energy levels, and it is desirable to design control pulses that are more robust to noise. There

are several ways to design noise resilient controls, including robust optimization methods, in which a min-max problem is solved [54], or risk-neutral/averse optimization approaches that minimize the expectation of a utility function based on the original objective function subject to uncertain parameters [55].

In this section we consider a risk-neutral strategy to design a $|0\rangle \leftrightarrow |2\rangle$ SWAP gate on a single qubit ($Q = 1$), with three essential levels and one guard level. Let $\epsilon \sim \text{Unif}(-\epsilon_{\max}, \epsilon_{\max})$ be a uniform random variable for some $\epsilon_{\max} > 0$. As a simple example, we consider the uncertain system Hamiltonian $H_s^u(\epsilon) = H_s^{rw} + H'(\epsilon)$ where H_s^{rw} is given by (5.15), and $H'(\epsilon)$ is a diagonal perturbation:

$$\frac{H'(\epsilon)}{2\pi} = \begin{pmatrix} 0 & & & \\ & \epsilon/100 & & \\ & & \epsilon/10 & \\ & & & \epsilon \end{pmatrix}.$$

Here, no perturbation is imposed on the control Hamiltonian (5.16). From these assumptions follow that the uncertain system Hamiltonian has expectation $\mathbb{E}[H_s^u(\epsilon)] = H_s^{rw}$. We may correspondingly update the original objective function, $\mathcal{G}(\boldsymbol{\alpha}, H_s^{rw})$, to the risk-neutral utility function $\widetilde{\mathcal{G}}(\boldsymbol{\alpha}) = \mathbb{E}[\mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon))]$. Given the simple form of the random variable ϵ, we may compute $\widetilde{\mathcal{G}}$ by quadrature:

$$\mathbb{E}[\mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon))] = \int_{-\epsilon_{\max}}^{\epsilon_{\max}} \mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon))\, d\epsilon \approx \sum_{k=1}^{M} w_k \mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon_k)), \tag{5.74}$$

where w_k and ϵ_k are the weights and collocation points of a quadrature rule.

For the following example, we compare the optimal control obtained using the standard optimization procedure (no noise) and a risk-neutral control, in which the utility function (5.74) is computed using the Gauss-Legendre quadrature with $N = 9$ collocation points and $\epsilon_{\max} = 10$ MHz. We set the gate duration to $T = 300$ ns, the maximum allowable amplitude to $\alpha_{\max}/2\pi = 12$ MHz, the fundamental frequency to $\omega_1/2\pi = 4.10336$ GHz, with detuning frequency $\Delta_1 = 0$, and the self-Kerr coefficient to $\xi_1/2\pi = 0.2198$ GHz.

The control functions are constructed using two carrier waves with frequencies $\Omega_{1,1} = 0$

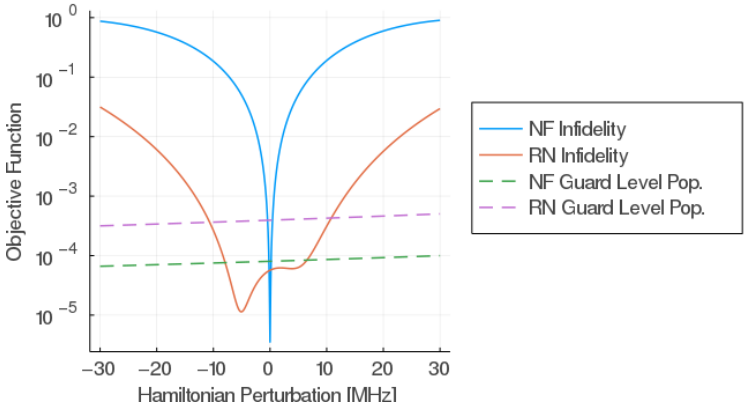


Figure 5.7: Infidelity objective ($\mathcal{J}_1$) and guard level objective ($\mathcal{J}_2$) as function of ε in $H_s^u(\epsilon)$. Here 'NF' and 'RN' correspond to the "Noise-Free" and "Risk-Neutral" cases.

and $\Omega_{1,2} = -\xi_1$ for both the "noise-free" (NF) and "risk-neutral" (RN) cases. In each case we use $D_1 = 12$ splines per control and carrier wave frequency for a total of $D = 48$ splines. We additionally constrain the controls to start and end at zero. We set the tolerance for L-BFGS to 10^{-5}, the maximum iteration count to 150, and use a maximum of five previous iterates to approximate the Hessian at each iteration. For the noise-free and risk-neutral optimized control functions, we use the perturbed Hamiltonian $H_s^u(\epsilon)$ to evaluate the objective function $\mathcal{G}$, for evenly spaced ϵ in the range $[-30, 30]$ MHz. The results are shown in Figure 5.7. From Figure 5.7 we note that the optimal control corresponding to the noise-free approach obtains the smallest infidelity for $\epsilon = 0$, but it grows rapidly for $|\epsilon| > 0$. By comparison, the optimal control found with the risk-neutral approach is much less sensitive to noise. We plot the control functions for both cases in Figure 5.8. Note that the risk-neutral controls (bottom panel) have larger amplitudes compared to the noise-free controls (top panel), indicating a potential drawback of the risk-neutral approach. However, a more systematic study of this issue is needed and left for future work.

5.6 Comparing Juqbox with QuTiP/pulse_optim and Grape-TF

The QuTiP/pulse_optim package is part of the QuTiP [71] framework and implements the GRAPE algorithm in the Python language. The Grape-TF code (TF is short for TensorFlow [2]) is

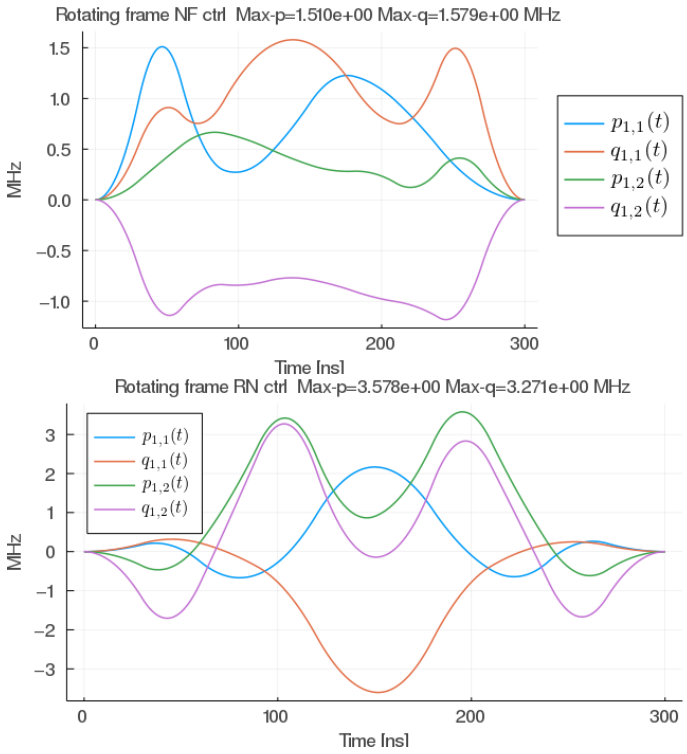


Figure 5.8: Control functions (without carrier waves) for the cases: "noise-free" (top), and "risk-neutral" (bottom). Here, $p_{k,n}(t)$ and $q_{k,n}(t)$ are defined in (5.27).

also implemented in Python and provides an enhanced implementation of the GRAPE algorithm, as described by Leung et al. [78]. It is callable from QuTiP and shares a similar problem setup with the pulse_optim function.

To compare the Juqbox code with pulse_optim and Grape-TF, we consider a set of SWAP gates. These gates transform the ground state $|0\rangle$ to excited state $|d\rangle$, and vice versa. The transformation can be described by the unitary matrix

$$V_g = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathbb{C}^{(d+1)\times(d+1)}, \tag{5.75}$$

which involves $E = d + 1$ essential states. To evaluate how much leakage occurs to higher energy levels, we add one guard (forbidden) level ($G = 1$) and evolve a total of $N = d + 2$ states in Schrödinger's equation. As before, the guard level is left unspecified in the target gate transformation. We consider implementing the SWAP gates on a multi-level qudit that can be described by the fundamental frequency $\omega_1/2\pi = 4.8$ GHz and the self-Kerr coefficient $\xi_1/2\pi = 0.22$ GHz. We apply the rotating wave approximation, where the angular frequency of the rotation is ω_1, resulting in the Hamiltonian model (5.15) and (5.16). As a realistic model for current superconducting quantum devices, we impose the control amplitude restrictions

$$\max_t |d(t;\boldsymbol{\alpha})| \leq c_\infty, \quad \frac{c_\infty}{2\pi} = 9\,\text{MHz}, \tag{5.76}$$

in the rotating frame of reference.

5.6.1 Setup of simulation codes

QuTiP/pulse_optim can minimize the target gate fidelity, $\mathcal{J}_1$, but does not suppress occupation of higher energy states. Thus, it does **not** minimize terms of the type $\mathcal{J}_2$. As a proxy for $\mathcal{J}_2$, we append one additional energy level to the simulation and measure its occupation as an estimate

of leakage to higher energy states. In pulse_optim, the control functions are discretized on the same grid in time as Schrödinger's equation and no smoothness conditions are imposed. In our tests, we use a random initial guess for the parameter vector.

Grape-TF discretizes the control functions on the same grid in time as Schrödinger's equation. It minimizes an objective function that consists of a number of user-configurable parts. In our test, we minimize the gate infidelity ($\mathcal{J}_1$) and the occupation of one guard (forbidden) energy level (similar to $\mathcal{J}_2$). To smooth the control functions in time, the objective function also contains additional terms to minimize their first and second time derivatives. The various parts of the objective function are weighted together by user-specified coefficients. The gradient of the objective function is calculated using the automatic differentiation (AD) technique, as implemented in the TensorFlow package. In our tests, we use a random initial guess for the control vector.

In Juqbox, we trigger the first d transition frequencies in the system Hamiltonian by using d carrier waves in the control functions, with frequencies

$$\Omega_{1,k} = (k-1)(-\xi_1), \quad k = 1, 2, \ldots, N_f, \quad N_f = d.$$

Similar to pulse_optim and Grape-TF, a pseudo-random number generator is used to construct the initial guess for the parameter vector.

The pulse_optim and Juqbox simulations were run on a Macbook Pro with a 2.6 GHz Intel iCore-7 processor. To utilize the GPU acceleration in TensorFlow, the Grape-TF simulations were run on one node of the Pascal machine at Livermore Computing, where each node has an Intel XEON E5-2695 v4 processor with two NVIDIA P-100 GPUs.

5.6.2 Numerical results

A SWAP gate where the control functions meet the control amplitude bounds (5.76) can only be realized if the gate duration is sufficiently long. Furthermore, the minimum gate duration increases with d. For each value of d, we used numerical experiments to determine a duration T_d such that at least two of the three simulation codes could find a solution with a small gate

infidelity. For Juqbox, we used the technique in Section 5.3.2 with $C_P = 80$ to obtain the number of time steps. The number of control parameters follow from $D = 2N_f D_1$, where $N_f = d$ equals the number of carrier wave frequencies and D_1 is the number of B-splines per control functions. Here, $D_1 = 10$ for $d = 3, 4, 5$ and $D_1 = 20$ for $d = 6$. For pulse_optim and Grape-TF, we calculate the number of time steps based on the shortest transition period, corresponding to the highest transition frequency in the system. We then use 40 time steps per shortest transition period to resolve the control functions. For both GRAPE methods there are 2 control parameters per time step. The main simulation parameters are given in Table 5.2.

		# time steps		# parameters	
d	T_d [ns]	Juqbox	GRAPE	Juqbox	GRAPE
3	140	14,787	4,480	60	8,960
4	215	37,843	7,568	80	15,136
5	265	69,962	11,661	100	23,322
6	425	157,082	22,441	240	44,882

Table 5.2: Gate duration, number of time steps (M) and total number of control parameters (D) in the $|0\rangle \leftrightarrow |d\rangle$ SWAP gate simulations. The number of time steps and control parameters are the same for pulse_optim and Grape-TF.

Optimization results for the pulse_optim, Grape-TF and Juqbox codes are presented in Tables 5.3, 5.4 and 5.5. The pulse_optim code generates piecewise constant control functions that are very noisy and may therefore be hard to realize experimentally. To obtain a realistic estimate of the resulting dynamics, we interpolate the optimized control functions on a grid with 20 times smaller time step and use the `mesolve()` function in QuTiP to calculate the evolution of the system from each initial state. We then evaluate the gate infidelity using the evolved states at the final time, denoted by $\mathcal{J}_1^*$ in Table 5.3. Since the control functions from Grape-TF and Juqbox are significantly smoother, we report the target gate fidelities as calculated by those codes.

d	$\mathcal{J}_1^*$	$\lvert\psi^{(d+1)}\rvert^2_\infty$	$\lvert p\rvert_\infty$ [MHz]	$\lvert q\rvert_\infty$ [MHz]	# iter	CPU [s]
3	4.35e-6	9.41e-3	9.00	9.00	38	30
4	3.91e-5	1.20e-2	9.00	9.00	93	108
5	1.57e-4	8.77e-3	9.00	9.00	215	385
6	**1.76e-3**	**4.48e-2**	9.00	9.00	246	894

Table 5.3: QuTiP/pulse_optim results for $|0\rangle \leftrightarrow |d\rangle$ SWAP gates. Note the larger infidelity and guard state population for $d = 6$.

d	$\mathcal{J}_1$	$\lvert\psi^{(d+1)}\rvert^2_\infty$	$\lvert p\rvert_\infty$ [MHz]	$\lvert q\rvert_\infty$ [MHz]	# iter	CPU [s]
3	8.76e-6	4.03e-3	6.98	8.83	78	2,062
4	1.52e-5	3.39e-3	6.87	6.54	128	10,601
5	2.80e-5	1.78e-3	7.21	7.62	161	28,366
6	**4.89e-1**	2.33e-5	0.73	0.74	93	81,765

Table 5.4: Grape-TF results for $|0\rangle \leftrightarrow |d\rangle$ SWAP gates. Note the very large infidelity for $d = 6$. These simulations used two NVIDIA P-100 GPUs to accelerate TensorFlow.

d	$\mathcal{J}_1$	$\lvert\psi^{(d+1)}\rvert^2_\infty$	$\lvert p\rvert_\infty$ [MHz]	$\lvert q\rvert_\infty$ [MHz]	# iter	CPU
3	2.71e-5	1.92e-3	7.59	8.99	177	55
4	4.91e-5	1.23e-3	7.78	5.33	166	151
5	4.95e-5	1.25e-3	7.42	7.24	173	291
6	7.41e-6	4.41e-3	4.55	5.39	229	1255

Table 5.5: Juqbox results for $|0\rangle \leftrightarrow |d\rangle$ SWAP gates.

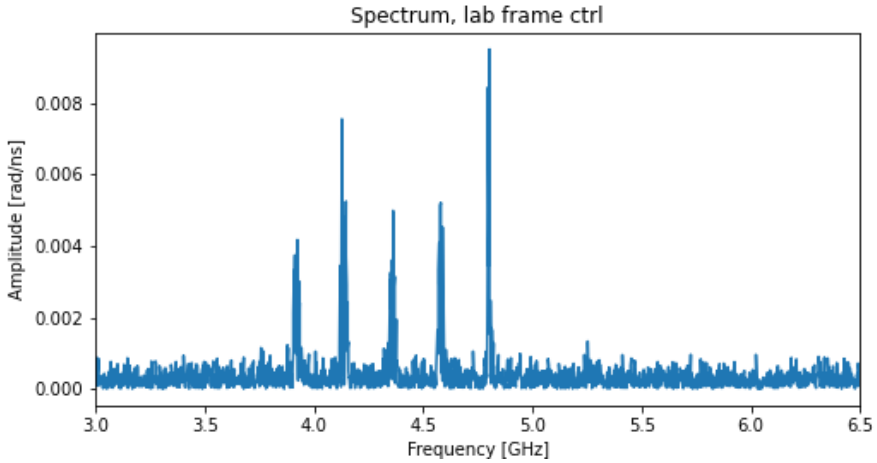
For the $|0\rangle \leftrightarrow |3\rangle$, $|0\rangle \leftrightarrow |4\rangle$ and $|0\rangle \leftrightarrow |5\rangle$ SWAP gates, all three codes produce control functions with very small gate infidelities. We note that the population of the guard level, $|\psi^{(d+1)}|^2$, is about an order of magnitude larger with pulse_optim than with Juqbox; the guard level population from Grape-TF are somewhere in between. The most significant difference between the results occur for the $d = 6$ SWAP gate. Here, the Grape-TF code fails to produce a small gate infidelity after running for almost 23 hours and the pulse_optim code results in a gate fidelity that is about 2 orders of magnitude larger than Juqbox.

While pulse_optim and Juqbox require comparable amounts of CPU time to converge, the Grape-TF code is between 50-100 times slower, despite the GPU acceleration.

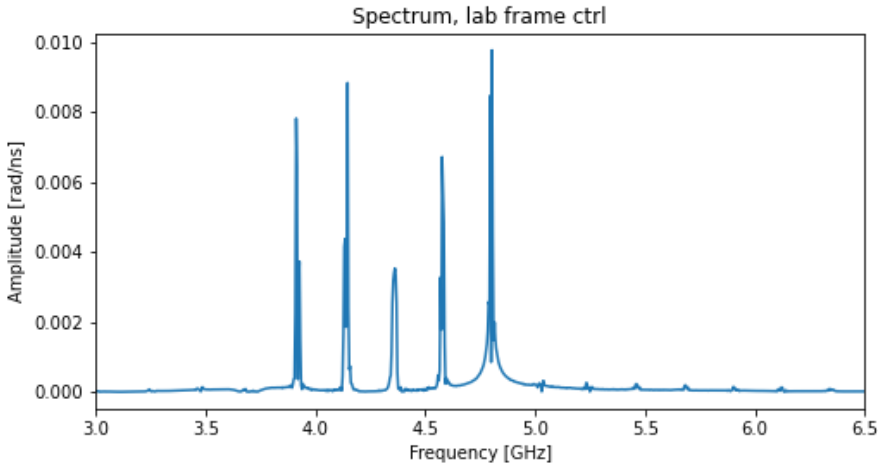
We proceed by analyzing the optimized control functions and take the $|0\rangle \leftrightarrow |5\rangle$ SWAP gate as a representative example. In this case, the relevant transition frequencies in the laboratory frame of reference are

$$f_k = \frac{1}{2\pi}\left(\omega_1 - k\xi_1\right), \quad k = 0, 1, 2, 3, 4. \tag{5.77}$$

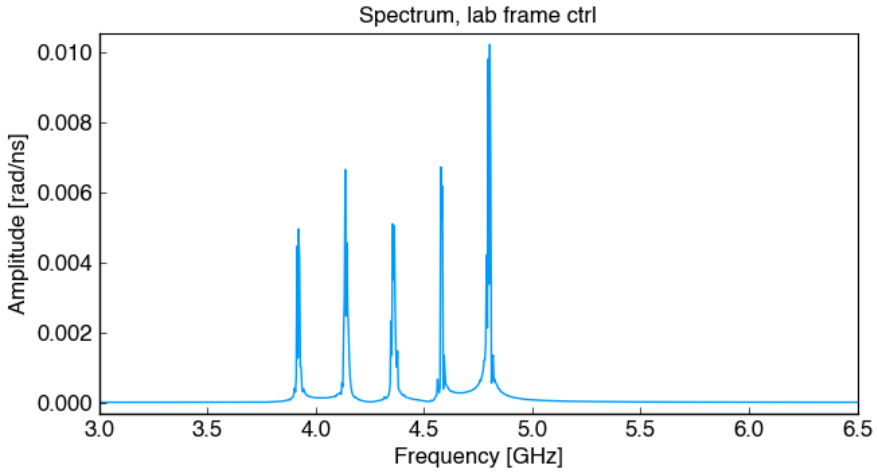
To compare the smoothness of the optimized control functions, we study the Fourier spectra of the laboratory frame control functions, see Figure 5.9. We first note that pulse_optim produces a significantly noisier control function compared to the other two codes. The control function from Grape-TF is significantly smoother, even though its spectrum includes some noticeable peaks at frequencies that do not correspond to transition frequencies in the system. The Juqbox simulation results in a laboratory frame control function where each peak in the spectrum corresponds to a transition frequency in the Hamiltonian.



(a) QuTiP/pulse_optim



(b) Grape-TF



(c) Juqbox

Figure 5.9: Magnitude of the Fourier spectrum of the laboratory frame control function for the $|0\rangle \leftrightarrow |5\rangle$ SWAP gate.

5.7 Conclusions

In this chapter we developed numerical methods for optimizing control functions for realizing logical gates in closed quantum systems where the state is governed by Schrödinger's equation. By asymptotic expansion, we calculated the resonant frequencies in the system Hamiltonian, corresponding to transitions between energy levels in the state vector. We introduced a novel parameterization of the control functions using B-spline basis functions that act as envelopes for carrier waves, with frequencies that match the transition frequencies. This approach allows the number of control parameters to be independent of, and significantly smaller than, the number of time steps for integrating Schrödinger's equation.

The objective function in the optimal control problem consists of two parts: the infidelity of the final gate transformation and a time-integral for evaluating leakage to higher energy levels. We apply a "discretize-then-optimize" approach and outline the derivation of the discrete adjoint equation that is solved to efficiently calculate the gradient of the objective function.

To demonstrate our approach, we optimized the control functions for a CNOT gate with two guard states, resulting in a gate fidelity exceeding 99.99%. Having a moderate number of control parameters enabled us to study the spectrum of the Hessian of the objective function at an optima. We found that imposing tighter bounds on the parameter vector results in a Hessian with larger eigenvalues and thus improves the convergence of the optimization algorithm.

Based on a simple noise model, we also generalized the proposed method to calculate risk-neutral controls that are resilient to uncertainties in the Hamiltonian model. The results are promising and indicate that a more systematic study of optimization under uncertainty can yield controls that are robust to noise in quantum systems. We finally compared the performance of the proposed method, implemented in the Juqbox package [53], and two implementations of the GRAPE algorithm: the pulse_optim method in QuTiP [71] and Grape-TensorFlow [78]. The codes were compared on a set of SWAP gates on a single qudit. Here, Juqbox was found to run 50-100 times faster than Grape-TensorFlow and produce control functions that are significantly smoother

than pulse_optim.

In future work, we intend to generalize our approach to solve optimal control problems for larger quantum systems.

Chapter 6

IBM Open Science Prize – SWAP Gate Challenge

On November 30, 2020, IBM announced an open science prize competition aimed at improving the fidelity of a SWAP gate between qubits 5 and 6 on their Quantum device named "Casablanca." The details of the SWAP gate problem was outlined in a Jupyter notebook, which utilizes the IBM developed open source package Qiskit. The participants were allowed to work in teams with up to five people and restricted to only use open source software in solving the problem. The competition concluded on April 16, 2021, and on June 14, 2021, it was announced that there were no winners as no team achieved the desired 50% reduction in error of the SWAP gate.

As a case study in the use of the optimal control techniques outlined in Chapter 5, in this (brief) chapter we outline an approach for the SWAP gate challenge. The approach in this chapter is based on the quantum optimal control techniques implemented in the open source packages Juqbox.jl (the methods for which were outlined in Chapter 5) and Quandary [64]. The results from the optimal control approach are only as good as the accuracy in the description of the quantum system dynamics, characterized by a Hamiltonian model that was provided by IBM. Based on the calibrated control pulses that IBM provide for their standard gate set, we describe a reverse engineering approach to calibrate our computational model, including effects of cross-talk. Techniques were developed to translate between Qiskit's pulse representation and the B-spline formulation used in Juqbox and Quandary. The fidelity of the optimized pulse sequences were finally estimated using Qiskit's randomized benchmarking (RB) techniques.

6.1 Hamiltonian model

The Hamiltonian model used in this study is based on work by Magesan and Gambetta [84]. In [84], the authors considered a system of two transmons coupled by a bus resonator. The bus resonator is modeled as a harmonic oscillator with fundamental frequency ω^r and each transmon is coupled to the bus resonator by a Jaynes-Cummings Hamiltonian with coupling strength g_j. Let the $|01\rangle$ transition frequencies of the transmons be ω_j. It is assumed that the coupling is in the dispersive regime, which means that resonator frequency is sufficiently detuned from the $|01\rangle$ transition frequencies to make $|\omega_j - \omega^r| \gg |g_j|$.

After transforming the Hamiltonian to a frame rotating with frequency ω^r in all three subsystems (two transmons and a bus resonator), the system and control Hamiltonians become (for notational convenience we set $\hbar = 1$)

$$H_{sys} = \sum_{j=1}^{2} \left((\omega_j - \omega^r)\, b_j^\dagger b_j + \frac{\Delta_j}{2} b_j^\dagger b_j^\dagger b_j b_j \right) + g_j \left(b_j^\dagger c + b_j c^\dagger \right), \tag{6.1}$$

$$H_{ctrl}(t) = \sum_{j=1}^{2} \mathrm{Re}(e^{i\omega^r t} d_j(t)) \left(e^{-i\omega^r t} b_j + e^{i\omega^r t} b_j^\dagger \right), \tag{6.2}$$

where b_j is the lowering operator for the j-th transmon and c is the lowering operator for the bus resonator. The above model is an ideal starting point for an optimal control approach for designing a SWAP gate. In practice, unfortunately, the resonator frequency ω^r and the coupling coefficients g_j are not readily available from the IBM backend description of the Casablanca system.

Magesan and Gambetta [84] proceed by deriving a simplified Hamiltonian model using the following steps: 1) reorder the state vector into blocks of increasing transmon excitation number, 2) adiabatically eliminate the terms that couple the blocks, 3) project the Hamiltonian onto the zero-excitation subspace of the bus resonator. These steps results in an effective Hamiltonian for the two transmon system given in Equation (2.12) of [84]. This model is completely specified for the Casablanca system as all parameters can be accessed through the Qiskit interface. For these reasons, it is used as a starting point in our modeling effort. Based on this model, the lab frame

system Hamiltonian for qubits 5 and 6 of Casablanca is

$$H_{sys} = \sum_{j=5}^{6} \left(\tilde{\omega}_j \, b_j^\dagger b_j + \frac{\Delta_j}{2} b_j^\dagger b_j^\dagger b_j b_j \right) + j_{56} \left(b_5^\dagger b_6 + b_5 b_6^\dagger \right). \tag{6.3}$$

Here $\tilde{\omega}_j$ is the dressed frequency of the j-th transmon, $b_5 = I \otimes a$, and $b_6 = a \otimes I$, where the lowering matrix for a single system is denoted by a. For simplicity, in the following we assume we use the dressed frequencies only and the tildes on the frequencies will be suppressed. Based on Equation (2.14) in [84] and the Hamiltonian entry in the backend of the Casablanca system, the lab frame control Hamiltonian is

$$H_{ctrl}(t) = \Omega_{d,5} \left(D_5(t) + U_{10}^{(5,6)}(t) \right) (b_5 + b_5^\dagger) + \Omega_{d,6} \left(D_6(t) + U_{11}^{(6,5)}(t) \right) (b_6 + b_6^\dagger), \tag{6.4}$$

where

$$D_5(t) = \mathrm{Re}\left(e^{i\omega_5 t} d_5(t) \right), \quad U_{10}^{(5,6)}(t) = \mathrm{Re}\left(e^{i\omega_6 t} u_{10}(t) \right), \tag{6.5}$$

$$D_6(t) = \mathrm{Re}\left(e^{i\omega_6 t} d_6(t) \right), \quad U_{11}^{(5,6)}(t) = \mathrm{Re}\left(e^{i\omega_5 t} u_{11}(t) \right). \tag{6.6}$$

For conciseness we have absorbed the phase factors $e^{i\phi}$ into the normalized control functions d_5 through u_{11}. Note that U_{10} is applied to qubit 5 but uses qubit 6's transition frequency. Correspondingly, U_{11} is applied to qubit 6, but uses qubit 5's transition frequency. Because $\mathrm{Re}(z) = 0.5(z + \bar{z})$ for $z \in \mathbb{C}$, the control Hamiltonian can also be written

$$\begin{aligned} H_c(t) =& \frac{1}{2} \Omega_{d,5} \left(e^{i\omega_5 t} d_5(t) + e^{-i\omega_5 t} \bar{d}_5(t) + e^{i\omega_6 t} u_{10}(t) + e^{-i\omega_6 t} \bar{u}_{10}(t) \right) (b_5 + b_5^\dagger) \\ &+ \frac{1}{2} \Omega_{d,6} \left(e^{i\omega_6 t} d_6(t) + e^{-i\omega_6 t} \bar{d}_6(t) + e^{i\omega_5 t} u_{11}(t) + e^{-i\omega_5 t} \bar{u}_{11}(t) \right) (b_6 + b_6^\dagger). \end{aligned} \tag{6.7}$$

We apply a rotating frame transformation using the same frequency of rotation for both sub-systems (e.g. $\omega_{rot} = \omega_5$). In this frame, the system Hamiltonian becomes

$$\widetilde{H}_{sys} = R H_{sys} R^\dagger + i R^\dagger \dot{R}, \tag{6.8}$$

$$= \sum_{j=5}^{6} \left(\delta_j \, b_j^\dagger b_j + \frac{\Delta_j}{2} b_j^\dagger b_j^\dagger b_j b_j \right) + j_{56} \left(b_5^\dagger b_6 + b_5 b_6^\dagger \right), \tag{6.9}$$

where we have defined $\delta_j = \omega_j - \omega_{rot}$ for $j = 5, 6$. After applying the rotating wave approximation, the control Hamiltonian becomes

$$\widetilde{H}_c(t) \approx \frac{\Omega_{d,5}}{2}\left(e^{i\delta_5 t} d_5(t) + e^{i\delta_6 t} u_{10}(t)\right) b_5 + \frac{\Omega_{d,6}}{2}\left(e^{i\delta_6 t} d_6(t) + e^{i\delta_5 t} u_{11}(t)\right) b_6 + \text{H.c.}, \tag{6.10}$$

where H.c. stands for the Hermitian conjugate.

6.2 Optimal control with Juqbox.jl and Quandary

We use numerical optimization in the open source packages Juqbox.jl [52] and Quandary [64] to determine the control functions $d_5(t)$, $d_6(t)$, $u_{10}(t)$, and $u_{11}(t)$ for realizing the SWAP gate transformation,

$$V_{SW} = \begin{bmatrix} 1 & & & \\ & 0 & 1 & \\ & 1 & 0 & \\ & & & 1 \end{bmatrix}. \tag{6.11}$$

Since higher energy levels play an important role in a cross-resonance gate, we model each transmon with 4 energy levels leading to a state vector with $N = 16$ elements. In the closed system setting, the time-evolution of the quantum system is unitary, yielding the transformation $\psi(t) = U(t, \boldsymbol{\alpha})\psi(0)$ for any initial quantum state $\psi(0)$. Here, $\boldsymbol{\alpha} \in \mathbb{C}^D$ is the vector of control parameters and $U(t, \boldsymbol{\alpha}) \in \mathcal{C}^{N\times N}$ is the unitary solution matrix, which solves Schrödinger's equation

$$\dot{U}(t, \boldsymbol{\alpha}) = -i\widetilde{H}(t, \boldsymbol{\alpha})U(t, \boldsymbol{\alpha}) \quad 0 < t \leq T, \quad \text{with} \quad U(0) = I_N, \tag{6.12}$$

where I_N is the $N \times N$ identity matrix and $\widetilde{H}(t, \boldsymbol{\alpha}) = \widetilde{H}_{sys} + \widetilde{H}_c(t, \boldsymbol{\alpha})$ denotes the Hamiltonian in the rotating frame.

The main target of the optimization is to find the vector of control parameters $\boldsymbol{\alpha}$ that minimizes the difference between the target SWAP gate matrix, V_{SW}, and the final-time solution operator, $U(T, \boldsymbol{\alpha})$, projected onto the two lowest energy levels of each transmon. The difference

between the matrices is measured in terms of the trace infidelity

$$\mathcal{J}_1(\boldsymbol{\alpha}) = 1 - \frac{1}{E^2}\left|\mathrm{Tr}\left(V_{SW}^\dagger \widetilde{U}(T,\boldsymbol{\alpha})\right)\right|^2, \quad \widetilde{U} = PUP^\dagger. \tag{6.13}$$

Here, $U(t,\boldsymbol{\alpha})$ solves Schrödinger's equation (6.12) and P extracts the first two energy levels of each transmon from the full state vector. Thus, only $E = 4$ columns of the solution matrix U are used for evaluating the trace infidelity. The time-averaged population of the highest energy level in each transmon, $\mathcal{J}_2(\boldsymbol{\alpha})$, is used to discourage leakage to non-computational levels of the transmons. The optimization then minimizes the total objective function $\mathcal{J}_1 + \mathcal{J}_2$, subject to constraints on the amplitude of the control functions, as outlined in Chapter 5.

Both Juqbox and Quandary represent the control Hamiltonian in terms of their real and imaginary components,

$$\widetilde{H}_{ctrl}(t) = p_5(t)(b_5 + b_5^\dagger) + iq_5(t)(b_5 - b_5^\dagger) + p_6(t)(b_6 + b_6^\dagger) + iq_6(t)(b_6 - b_6^\dagger). \tag{6.14}$$

To identify the relation between the real-valued control functions (p_k, q_k) and the complex-valued functions d_5, d_6, u_{10} and u_{11} in (6.10), it is convenient to first introduce the complex-valued functions $\zeta_k(t)$,

$$\zeta_k(t) = \sum_{\mathcal{L}=1}^{N_s} \alpha_{k,\mathcal{L}} \hat{B}_{\mathcal{L}}(t), \quad \alpha_{k,\mathcal{L}} \in \mathbb{C}. \tag{6.15}$$

In Juqbox and Quandary, $\hat{B}_{\mathcal{L}}(t)$, $\mathcal{L} = 1, 2, \ldots, N_s$, are quadratic B-spline wavelets, uniformly spaced in time. We define,

$$p_5(t) + iq_5(t) := e^{i\delta_5 t}\zeta_5(t) + e^{i\delta_6 t}\zeta_{10}(t), \tag{6.16}$$

$$p_6(t) + iq_6(t) := e^{i\delta_6 t}\zeta_6(t) + e^{i\delta_5 t}\zeta_{11}(t). \tag{6.17}$$

As there are four control functions, the total number of control parameters becomes $D = 4N_s$. In the following, 30 B-spline coefficients were used to parameterize each control function. We remark that the number of control parameters can be chosen independently of (and usually much smaller than) the number of time steps for integrating Schrödinger's equation (In this study we used $N_T = 105,625$ time steps to integrate Schrödinger's equation to time $T = 668.4$ ns).

The functions $\zeta_k(t)$ allow the control Hamiltonian (6.14) to be written as

$$\widetilde{H}_{c,c}(t) = \left(e^{i\delta_5 t}\zeta_5(t) + e^{i\delta_6 t}\zeta_{10}(t)\right) b_5 + \left(e^{i\delta_6 t}\zeta_6(t) + e^{i\delta_5 t}\zeta_{11}(t)\right) b_6 + \text{H.c.} \tag{6.18}$$

The uncalibrated relation between the control functions in Qiskit and Juqbox is found by comparing (6.18) and (6.10),

$$d_5(t) = \frac{2}{\Omega_{d,5}}\zeta_5(t), \tag{6.19}$$

$$d_6(t) = \frac{2}{\Omega_{d,6}}\zeta_6(t), \tag{6.20}$$

$$u_{10}(t) = \frac{2}{\Omega_{d,5}}\zeta_{10}(t), \tag{6.21}$$

$$u_{11}(t) = \frac{2}{\Omega_{d,6}}\zeta_{11}(t). \tag{6.22}$$

6.2.1 Open system optimal control

To account for system-environment interactions, the numerical optimization with Juqbox can be used as a starting point for optimal control with the Quandary code [64]. Quandary describes open quantum systems using a density matrix $\rho \in \mathbb{C}^{N\times N}$. The time evolution of $\rho(t)$ is modeled by Lindblad's master equation

$$\dot{\rho}(t) = -\,i\left(H(t)\rho(t) - \rho(t)H(t)\right) + L\left(\rho(t)\right). \tag{6.23}$$

Both decay and dephasing processes are modeled using the Lindblad terms:

$$L(\rho) = \sum_{k=5}^{6}\sum_{l=1}^{2} \mathcal{L}_{lk}\rho\mathcal{L}_{lk}^{\dagger} - \frac{1}{2}\left(\mathcal{L}_{lk}^{\dagger}\mathcal{L}_{lk}\rho + \rho\mathcal{L}_{lk}^{\dagger}\mathcal{L}_{lk}\right), \tag{6.24}$$

where the collapse operators satisfy $\mathcal{L}_{1k} := \frac{1}{\sqrt{T_1^k}}a_k$ (decay) and $\mathcal{L}_{2k} := \frac{1}{\sqrt{T_2^k}}a_k^{\dagger}a_k$ (dephasing). Here, T_1^k and T_2^k correspond to the decay and dephasing times for system k.

Quandary solves Lindblad's master equation numerically by applying the implicit midpoint time-stepping method, which is a symplectic, second-order time-integration scheme of Runge-Kutta type. In order to derive the discrete adjoint equations, techniques from Algorithmic Differentiation are applied to yield consistent and exact gradients of the objective function at costs that are independent of the number of control parameters. The optimization problem is then solved iteratively

using gradient updates, preconditioned by the L-BFGS algorithm to incorporate Hessian information. Constraints on the maximum amplitudes of the control parameters are incorporated using a line-search procedure that projects the gradient onto the linear box-constraints of maximally allowed control amplitudes.

6.3 Rabi pulse calibrations

Qiskit support two types of channels that accept custom pulses: drive and control channels. Pulses on the drive channels are specified in the following way:

$$D_i(t_j) = \mathrm{Re}\left\{\exp(i2\pi f j dt + \phi) d_j\right\}. \tag{6.25}$$

Here f is a frequency that can be chosen by the user; it defaults to the qubit transition frequency for drive channels, and the frequency of the linked qubit for control channels. We additionally have that ϕ is a phase, the time step is set to $dt = 2/9$ ns for the Casablanca system, and d_j is the non-dimensional, complex-valued control amplitude at time $t_j = j\,dt$. For each channel, Qiskit specifies the maximum allowable amplitude signal, $\Omega_{d,i}$. Absorbing the phase into the dimensionless amplitude via $\tilde{d}_j = e^{i\phi} d_j$, then

$$\Omega_{d,i} D_i(t_j) = \Omega_{d,i}\left(\cos(2\pi f j dt)\mathrm{Re}\{\tilde{d}_j\} - \sin(2\pi f j dt)\mathrm{Im}\{\tilde{d}_j\}\right),$$

which gives the basic mapping between Qiskit and Juqbox:

$$p(t_j) = \frac{\mathrm{Re}\{\tilde{d}_j\}}{2\Omega_{d,i}}, \quad q(t_j) = \frac{\mathrm{Im}\{\tilde{d}_j\}}{2\Omega_{d,i}}. \tag{6.26}$$

To verify this mapping, we considered a Rabi pulse for a single qubit. With a single carrier-wave with zero frequency in the rotating frame and two energy levels, with constant $p = A_J, q = 0$, half a Rabi oscillation occurs in Juqbox simulations for the pulse duration

$$\tau_p = \frac{\pi}{|A_J|} \implies |A_J| = \frac{\pi}{\tau_p}.$$

Using Qiskit's amplitude convention, the corresponding relation is

$$A_Q = \mathrm{Re}\{\tilde{d}_j\} = \frac{2\pi}{\tau_p}\Omega_{d,i}.$$

The ratio between the amplitudes of the corresponding lab frame control signals in Qiskit and Juqbox becomes

$$\frac{|A_Q|}{2|A_J|} = \Omega_{d,i}.$$

This implies that the amplitudes should be proportional to each other. Results from a Rabi experiment on the Casablanca hardware are shown in Figure 6.1.

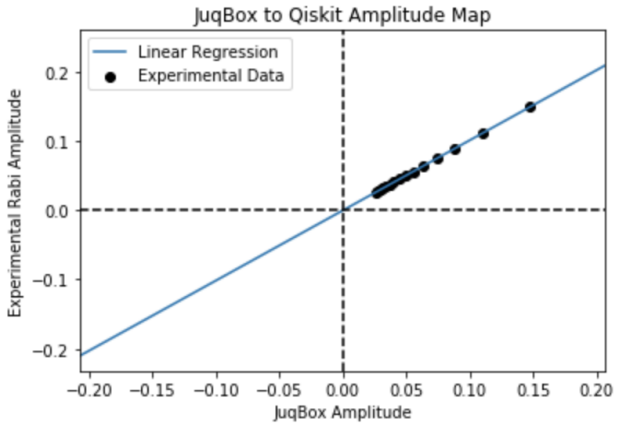


Figure 6.1: Results of the Rabi experiment on qubit 5 of the Casablanca hardware.

For the Casablanca hardware, the fitted line $y = 1.014x - 1.9 \cdot 10^{-4}$ has a slope that is close to $\Omega_{d,5} \approx 1.084$. This study confirms that the drive channel amplitude $\Omega_{d,5}$ is almost perfectly calibrated.

6.4 Gaussian square and DRAG pulses in Qiskit

We start by defining the zeroed Gaussian function, centered at time $T/2$:

$$g_z(t; A, T, \sigma) = \begin{cases} A\left(\exp\left(-\frac{(t-T/2)^2}{2\sigma^2}\right) - g_0(T,\sigma)\right), & t \in [0, T], \\ 0, & \text{otherwise.} \end{cases}$$

Here A is a complex-valued amplitude, T is the duration of the zeroed Gaussian, and σ is its standard deviation. Programmatically, however, the tails of the Gaussian are truncated by subtracting

out the constant g_0, defined by

$$g_0(T,\sigma) = \exp\left(-\frac{(T/2+dt)^2}{2\sigma^2}\right), \quad \text{where } dt = \tfrac{2}{9} \text{ ns.} \tag{6.27}$$

Based on the zeroed Gaussian pulse, we can now define the Derivative Removal by Adiabatic Gate (DRAG) pulse:

$$f(t;A,T,\sigma,\beta) = g_z(t;A,T,\sigma) + i\beta \underbrace{\left(-\frac{(t-T/2)}{\sigma^2}\right) g(t;A,T,\sigma)}_{g_z'(t)},$$

where β is a correction amplitude. Finally, the Gaussian square pulse is defined by its amplitude A, total duration T, and the duration of its constant part, w. Let the duration of the leading and trailing ramp be $r = (T-w)/2 > 0$. Then,

$$s(t;A,T,\sigma,w) = \begin{cases} \dfrac{g_z(t;A,2r,\sigma)}{1-g_0(2r,\sigma)}, & 0 \le t \le r, \\ A, & r \le t \le r+w, \\ \dfrac{g_z(T-t;A,2r,\sigma)}{1-g_0(2r,\sigma)}, & r+w \le t \le T. \end{cases}$$

6.5 Converting Qiskit pulses to B-splines with carrier waves

For many basic gates, Qiskit provides parametric representations of the pulse schedule required to realize the chosen gate. These provided pulses take the form (6.25). For simplicity, suppose we have the signal D_0 and wish to represent it in Juqbox/Quandary via B-splines with carrier waves. Since each signal in Qiskit is associated with a single frequency we have

$$\Omega_{d,0} D_0(t_j) = \Omega_{d,0}\left(\cos(2\pi f j dt)\mathrm{Re}\{\tilde{d}_j\} - \sin(2\pi f j dt)\mathrm{Im}\{\tilde{d}_j\}\right) = 2p(t_j)\cos(\omega t_j) - 2q(t_j)\sin(\omega t_j).$$

The approximation of the real and imaginary parts of d_j can be done independently, so that in the following we focus on approximating the real part. This becomes a classical interpolation problem

$$\frac{\Omega_{d,0}}{2}\mathrm{Re}\{d_k\} = p(t_k) = \sum_{j=1}^{D_1} \alpha_j B_j(t_k), \quad \forall k = 1,2,\ldots,N,$$

where N is the number of samples d_k. This gives us the Vandermonde system

$$\underbrace{\begin{pmatrix} B_1(t_1) & B_2(t_1) & B_3(t_1) & \cdots & B_{D_1}(t_1) \\ B_1(t_2) & B_2(t_2) & B_3(t_2) & \cdots & B_{D_1}(t_2) \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ B_1(t_N) & B_2(t_N) & B_3(t_N) & \cdots & B_{D_1}(t_N) \end{pmatrix}}_{R^{N\times D_1}} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{D_1} \end{pmatrix} = \frac{\Omega_{d,0}}{2} \begin{pmatrix} \mathrm{Re}\{d_1\} \\ \mathrm{Re}\{d_2\} \\ \vdots \\ \mathrm{Re}\{d_N\} \end{pmatrix},$$

where D_1 is the number of B-splines used in the approximation. Given that the time points t_k are distinct and $D_1 = N$ the above system is uniquely solvable. We also note that at a given time t_k only three B-splines are non-zero so that for a large number of splines, D_1, the above system is sparse. Moreover, if we choose the interpolation points t_k to be the centers of each B-spline and pick $D_1 = N$ then the above is the tridiagonal system

$$\begin{pmatrix} 3/4 & 1/8 & & & \\ 1/8 & 3/4 & 1/8 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1/8 \\ & & & 1/8 & 3/4 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{D_1-1} \\ \alpha_{D_1} \end{pmatrix} = \frac{\Omega_{d,0}}{2} \begin{pmatrix} \mathrm{Re}\{d_1\} \\ \mathrm{Re}\{d_2\} \\ \vdots \\ \mathrm{Re}\{d_{D_1-1}\} \\ \mathrm{Re}\{d_{D_1}\} \end{pmatrix},$$

which can be solved in $\mathcal{O}(D_1)$ time.

6.6 Reverse model calibration using X- and Cx-gates

The Casablanca system uses DRAG pulses to implement X-gates and a combination of Gaussian square and DRAG pulses to implement Cx gates. As the pulse coefficients are updated during each calibration of the system hardware, they can be used to calibrate the computational model for the Casablanca hardware. Given the high fidelity of the corresponding gates, the pulses can be used to engineer a mapping between the computational control functions we optimize with Juqbox / Quandary and the physical control functions that must be applied to the hardware.

6.6.1 X-gates

The X-gate for qubit 5 is defined by the single DRAG pulse

$$d_5(t) = f(t; A_1, T, \sigma, \beta_1),$$

for $0 \leq t \leq 160 \cdot dt$ where $dt = 2/9$ ns. The X-gate for qubit 6 is also defined by the single DRAG pulse

$$d_6(t) = f(t; A_2, T, \sigma, \beta_2),$$

for $0 \leq t \leq 160 \cdot dt$. The parameters defining each DRAG pulse can be accessed using Qiskit by querying the backend for Casablanca. The coefficients change slightly after each system calibration, but the duration of the DRAG pulses appears to be fixed at $160 \cdot dt$. At some point during the spring of 2021, the coefficients were

$$A_1 = 0.17545065110530234,$$

$$A_2 = 0.20674287767710134,$$

$$T = 160,$$

$$\sigma = 40,$$

$$\beta_1 = 0.47609887200679674,$$

$$\beta_2 = 1.9314472856919194.$$

The pulse schedules for the above X-gates are shown in Figure 6.2.

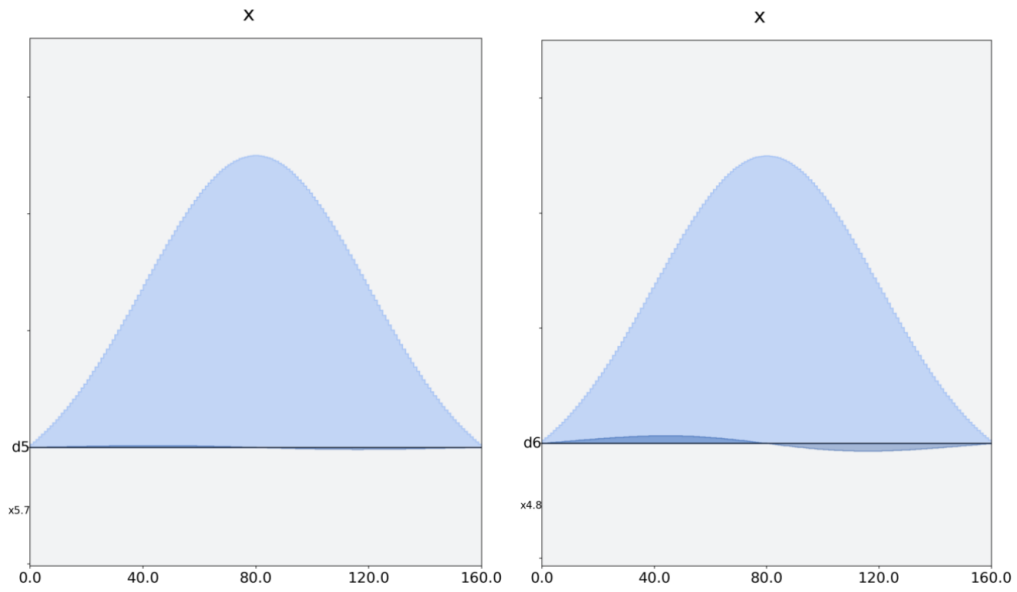


Figure 6.2: The pulse schedule for an X gate on Casablanca for qubit 5 (left), and qubit 6 (right).

Only the $d_5(t)$ function is active during the X5-gate and only the $d_6(t)$ function is active during the X6-gate, which allows the pulses to be calibrated independently. We want to improve the calculated average gate fidelity by modifying the control Hamiltonian (6.10) according to

$$d_5(t) \to \xi_5 d_5(t), \tag{6.28}$$

$$d_6(t) \to \xi_6 d_6(t), \tag{6.29}$$

where ξ_5 and ξ_6 are calibration factors. We were unable to achieve small trace gate infidelities based on (6.13). Upon closer examination, significant relative phase differences occurred between the target unitary and the simulated unitary evolution, which could not be explained by the rotating wave transformation. However, very good agreement in population was obtained. For this reason, we use the averaged gate fidelity, defined by

$$F_{avg} = \frac{1}{E}\sum_{j=1}^{N} \mathrm{Tr}|V^{\dagger}\widetilde{U}(T,\boldsymbol{\alpha})|, \tag{6.30}$$

in this calibration. Here V is the target gate unitary and $\widetilde{U}$ is the projected solution matrix for Schrödinger's equation at final time T. The highest X-gate fidelities were obtained for the

coefficients

$$\xi_5 = 0.9927, \quad \xi_6 = 0.9909, \tag{6.31}$$

indicating that the drive channels on the Casablanca system are very well calibrated. We remark that $\xi_5\Omega_{d,5} \approx 1.08$ is close to the slope 1.014 that was found in the Rabi calibration experiment in Section 4.

6.6.2 Calibrating a cross-talk model using the Cx gates

The Casablanca backend holds calibrated pulse sequences for two CX-gates that involve qubits 5 and 6: CX-56 (5 controls 6) and CX-65 (6 controls 5). These gates are implemented with a combination of DRAG and Gaussian square pulses and use the control functions d_5, d_6 and u_{11} (but not u_{10}), see Figure 6.3. Since the control function u_{11} acts on qubit 6 but uses qubit 5's frequency, the corresponding signal may be subject to cross-talk. Following [84], we can account for this effect through the simple model

$$u_{11}(t)b_6 \to \xi_{11}u_{11}(t)b_6 + A_c e^{i\phi_c}u_{11}(t)b_5, \tag{6.32}$$

where ξ_{11} and A_c are cross-talk coefficients and ϕ_c is a phase shift that compensates for the physical distance between qubits 5 and 6 on the chip. After a parameter space sweep, we found that

$$\xi_{11} = 2.02, \quad A_c = 0.0583, \quad \phi_c = 1.2189, \tag{6.33}$$

result in gate fidelities of 0.994 for both the CX-56 gate and the CX-65 gate.

To complete our modeling we also need to handle potential cross-talk from control function u_{10}. In lieu of calibrated pulses, we use a symmetry argument to motivate the same cross-talk model for u_{10} as for u_{11}, i.e.,

$$u_{10}(t)b_5 \to \xi_{10}u_{10}(t)b_5 + A_c e^{i\phi_c}u_{10}(t)b_6, \quad \xi_{10} = \xi_{11}. \tag{6.34}$$

As a result of the above calibrations, the modified Hamiltonian model becomes $\widetilde{H}_{comp}(t) =$

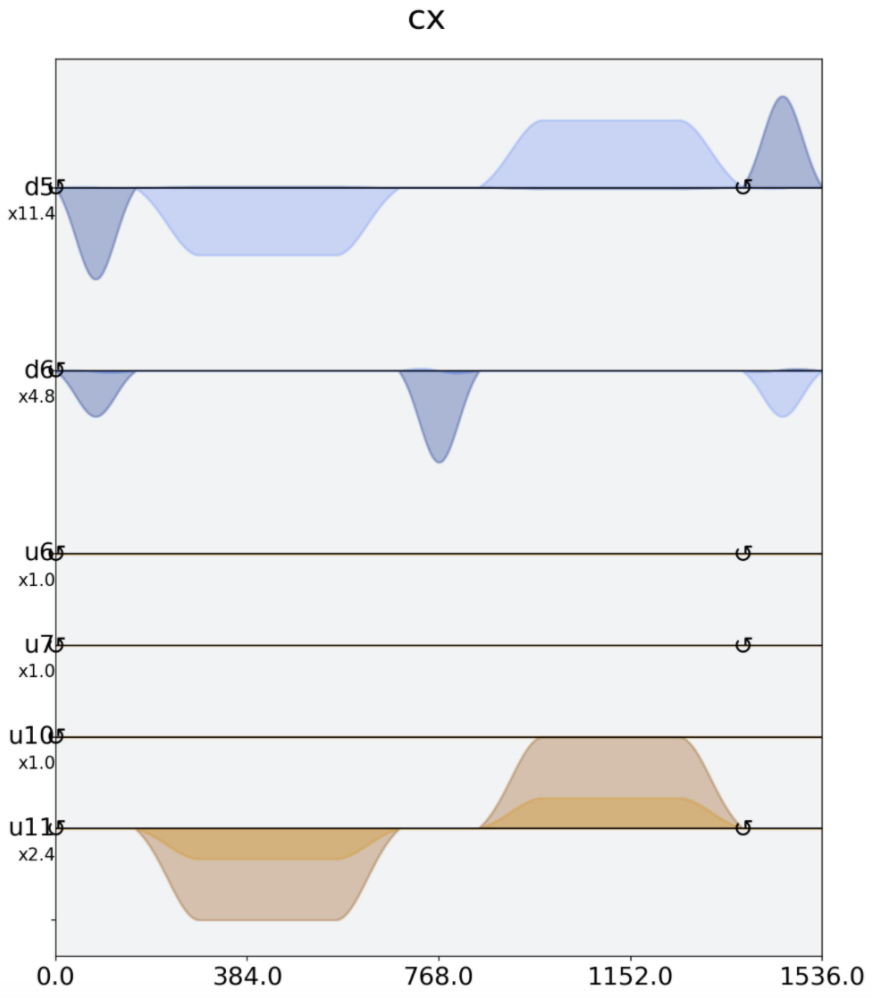


Figure 6.3: The pulse schedule for a CNOT gate on Casablanca where qubit 5 is the control qubit and qubit 6 is the target.

$\widetilde{H}_{sys} + \widetilde{H}_{c,c}(t)$, where

$$\widetilde{H}_{c,c}(t) = \frac{\Omega_{d,5}}{2}\left(e^{i\delta_5 t}\xi_5 d_5(t) b_5 + e^{i\delta_6 t} u_{10}(t)(\xi_{10} b_5 + A_c e^{i\phi_c} b_6)\right) + \frac{\Omega_{d,6}}{2}\left(e^{i\delta_6 t}\xi_6 d_6(t) b_6 + e^{i\delta_5 t} u_{11}(t)(\xi_{11} b_6 + A_c e^{i\phi_c} b_5)\right) + \text{H.c.} \tag{6.35}$$

The Juqbox code represents the control Hamiltonian in terms of its real and imaginary components, as described above. By identifying the coefficients between (6.18) and (6.35), we arrive at the calibrated conversion

$$\zeta_5(t) = \frac{\Omega_{d,5}}{2}\xi_5 d_5(t) + \frac{\Omega_{d,6}}{2} A_c e^{i\phi_c} u_{11}(t), \tag{6.36}$$

$$\zeta_6(t) = \frac{\Omega_{d,6}}{2}\xi_6 d_6(t) + \frac{\Omega_{d,5}}{2} A_c e^{i\phi_c} u_{10}(t), \tag{6.37}$$

$$\zeta_{10}(t) = \frac{\Omega_{d,5}}{2}\xi_{10} u_{10}(t), \tag{6.38}$$

$$\zeta_{11}(t) = \frac{\Omega_{d,6}}{2}\xi_{11} u_{11}(t). \tag{6.39}$$

Thus, control functions $(\zeta_5, \zeta_6, \zeta_{10}, \zeta_{11})$ that are optimized with Juqbox should be converted to Qiskit according to

$$d_5(t) = \frac{2}{\Omega_{d,5}\xi_5}\left(\zeta_5(t) - A_c e^{i\phi_c}\frac{1}{\xi_{11}}\zeta_{11}(t)\right), \tag{6.40}$$

$$d_6(t) = \frac{2}{\Omega_{d,6}\xi_6}\left(\zeta_6(t) - A_c e^{i\phi_c}\frac{1}{\xi_{10}}\zeta_{10}(t)\right), \tag{6.41}$$

$$u_{10}(t) = \frac{2}{\Omega_{d,5}\xi_{10}}\zeta_{10}(t), \tag{6.42}$$

$$u_{11}(t) = \frac{2}{\Omega_{d,6}\xi_{11}}\zeta_{11}(t). \tag{6.43}$$

The control pulses used by Qiskit are defined by inserting the above functions into (6.5)-(6.6).

6.7 Implementation of custom pulses in Qiskit

In Juqbox/Quandary, each control signal is represented by a continuous approximation of B-splines with carrier waves which we may simply evaluate at equispaced time points, $t_k = k\,dt$ where $dt = 2/9$ ns. As the Juqbox/Quandary samples have units of rad/ns, we use the mapping (6.26) to obtain the dimensionless amplitude samples $\tilde{d}_j$ that are required by Qiskit. Once we have

a set of dimensionless amplitude samples, we need to make Qiskit aware of our custom pulse. Our chosen approach is to simply pass a non-basis element Clifford gate (such as the SWAP gate) to `randomized_benchmarking_seq` as follows:

```
# Use standard swap gate for interleaved circuit
circ = QuantumCircuit(2)
circ.swap(0,1)
interleaved_elem = [circ]

# generate the RB circuit parameters
length_vector = np.arange(1,200,20)
nseeds = 5

rb_pattern = [[5,6]]

# Generate the RB circuits
_,_circs = randomized_benchmarking_seq(length_vector=length_vector,
                                       nseeds=nseeds,
                                       rb_pattern=rb_pattern,
                                       interleaved_elem=interleaved_elem)
```

With the interleaved circuits built, we then manually add our custom pulse to each element as follows:

```
for circuits in circs:
    for circuit in circuits:
        circuit.add_calibration("swap", qubits=[5, 6], schedule=sched)
```

where `sched` is a `Schedule` object containing our custom pulse. This approach allows us to avoid issues with `randomized_benchmarking_seq` decomposing a custom gate while providing a custom pulse that the transpiler respects when executing the sequence on the actual device. When downloading the associated JSON files with each RB job on IBM Quantum using this approach, it can be seen that each custom pulse is assigned to a 64 bit hexadecimal string indicating the signal is being used on the actual device.

A final practical matter with this approach is that directly adding the full custom waveform as a single instruction when building a schedule can result in the error: `Waveform memory exceeds the maximum amount of memory currently available # [8018]`. To avoid this error, we have built the full schedule by splitting each custom waveform in smaller chunks of (at most) 160 complex-valued amplitudes.

6.8 Randomized benchmarking results

To reduce influence of state preparation and measurement (SPAM) errors, the fidelity of the SWAP gate is estimated using randomized benchmarking. This functionality is provided by the routine `randomized_benchmarking_seq()` in Qiskit. This routine constructs two sets of randomized circuits for testing purposes. The first is composed of standard gates and is used as a reference. In the second set, the circuits are augmented by one or more interleaved gates, which in this case corresponds to a custom SWAP gate. As described above, the custom pulse schedule is explicitly inserted by calling `add_calibration()`, before transpiling each interleaved circuit. An example of a short interleaved pulse schedule is shown in Figure 6.4.

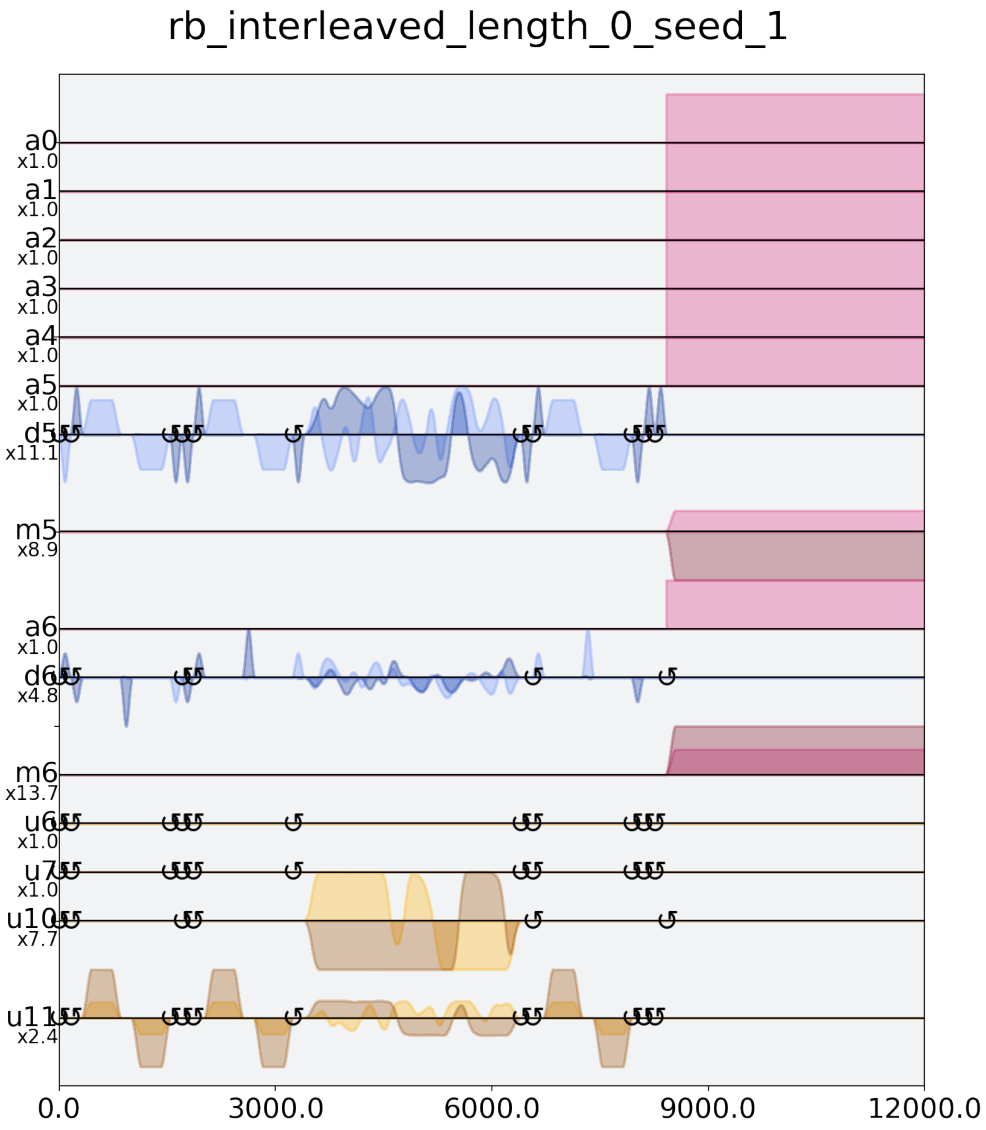


Figure 6.4: The pulse schedule for the first interleaved RB circuit.

The results of the non-interleaved and interleaved randomized benchmarking are processed by the Qiskit function `InterleavedRBFitter()`. It fits the data for each case to an exponential function and estimates the SWAP gate error as the difference in exponential decay between the fitted functions. With 5 randomized samples per case, 1000 shots per circuit, and considering circuit lengths in the range of 1-181, results from the QASM simulator are shown in Figure 6.5. In this case the estimated SWAP gate error was 2.36%.

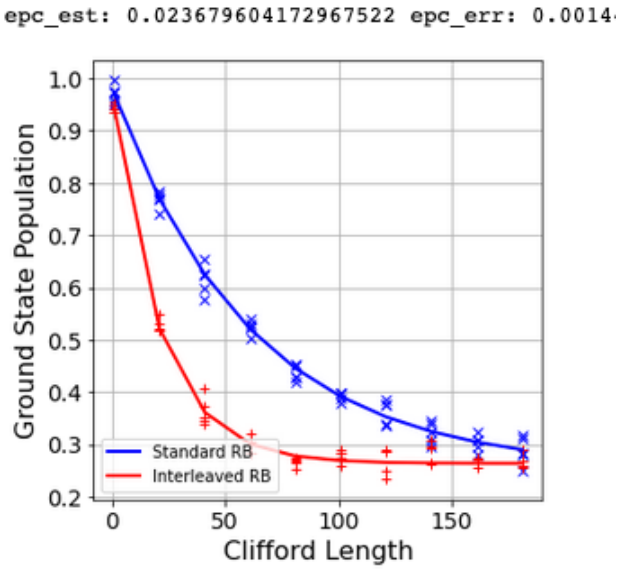


Figure 6.5: Simulated results. The observed ground state population as function of the length of the Clifford circuit. Here, the non-interleaved circuits are shown in blue and interleaved ones in red. The estimated error per Clifford (EPC) is 2.36%.

To test the custom SWAP gate pulse sequence on physical hardware, we consider a randomized benchmarking sequence consisting of 5 sets of interleaved and 5 sets of non-interleaved circuits. For each circuit, we take the average of 1000 separate shots/experiments. By inspecting the resulting state output histograms for the shortest interleaved circuits, we observe that the custom gate suffers from some severe accuracy problems, see Figure 6.6. Due to the large spread in the ground state population, the `InterleavedRBFitter()` function had problems fitting the data to an exponential decay. The estimated error per Clifford (EPC) was $-1.32 \cdot 10^{-2}$, but with a very large standard deviation, see Figure 6.7. The large gate errors could well be due to some misinterpretation of the pulse schedules for X and Cx gates. Another potential source of error is the Hamiltonian model.

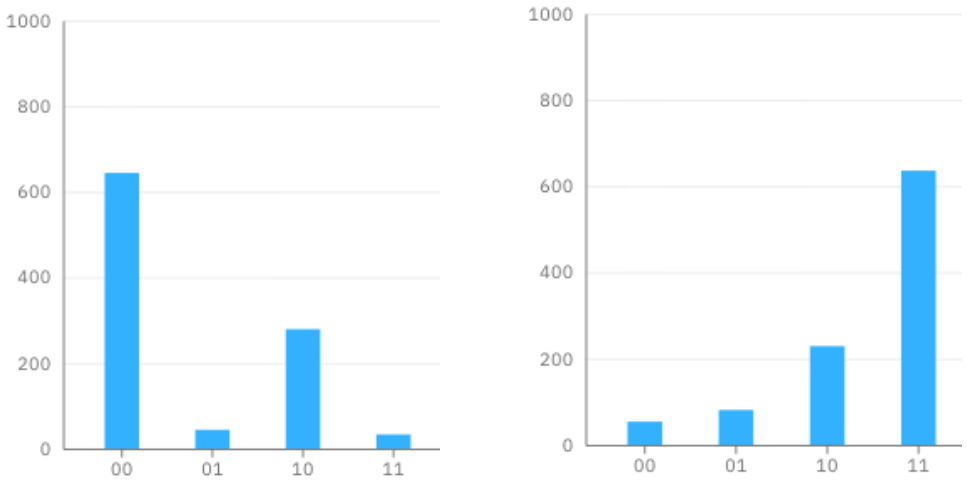


Figure 6.6: Classification results after 1000 shots in two of the randomized circuits with one interleaved SWAP gate. Ideally the $|00\rangle$ state should have 100% of the population.

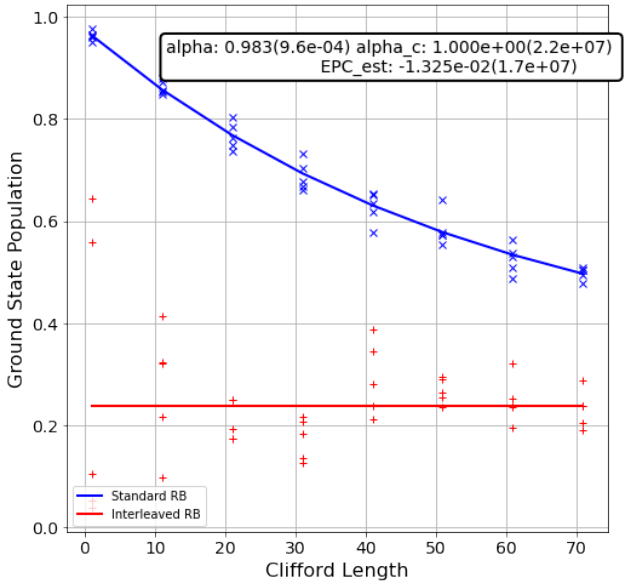


Figure 6.7: Randomized benchmarking results on the Casablanca hardware using the InterleavedRBFitter.

6.9 Conclusion

As mentioned in the introduction of this chapter, no winners were selected for the IBM SWAP gate challenge. It is clear that current NISQ-era machines are indeed noisy and difficult to control in practice. This use case highlights many interesting areas of exploration to make multi-qubit control possible in noisy-systems. Accurate system characterization, both of the bare system Hamiltonian and for noise processes inherent to a specific system, are necessary for the successful design of useful control signals. As seen in Chapter 5, even if armed with an (on average) accurate Hamiltonian model it is possible for noise processes to severely degrade the performance of predetermined optimal controls. It is clear that robust optimization techiniques, or optimization under uncertainty, could be leveraged to create control signals robust to noise.

Acknowledgment

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This report is contribution LLNL-TR-821599. We gratefully acknowledge financial support from Lawrence Livermore National Laboratory through the Laboratory Directed Research and Development (LDRD) program, grant # 20-ERD-028.

Chapter 7

Conclusion

In this thesis, we investigated two distinct but related problems with regards to wave phenomena. We will now summarize the results presented and discuss further avenues of exploration for both topics.

In Chapter 2 we presented and analyzed the WaveHoltz iteration, a new iterative method for solving the Helmholtz equation, for energy-conserving problems. We demonstrated that the iteration results in positive definite and sometimes symmetric matrices that are amenable to solution by iterative methods such as Krylov subspace methods. The numerical experiments indicated that the WaveHoltz iteration is a promising method with more favorable scaling for problems with outflow/impedance boundary conditions which are of much practical interest, e.g. for seismic applications.

In Chapter 3 we extended the analysis of Chapter 2 to problems with damping and/or impedance conditions. We additionally showed that the WaveHoltz iteration converges to the discrete Helmholtz solution to an order matching the order of the timestepper for arbitrary order modified equation centered timestepping schemes. We then demonstrated that knowledge of how the timestepping modifies the discrete WaveHoltz iteration allows one to *completely* remove time discretization errors.

In Chapter 4 we applied the WaveHoltz iteration to the "elastic" Helmholtz equation (also known as the Navier equation) for energy-conserving problems with Dirichlet and/or free surface boundary conditions. We additionally presented a second order implicit timestepping scheme with

a modification to remove time discretization errors as was done for a family of explicit schemes in Chapter 3. Numerical experiments indicate scaling similar to that of the acoustic Helmholtz equation considered in Chapter 2.

The WaveHoltz method has many avenues of exploration left. For the most part we have used unconditioned Krylov solvers to accelerate the WaveHoltz iteration, but the spectral properties of the WaveHoltz operator, $I - \mathcal{S}$, indicate that preconditioning should be possible. Given that the spectral radius of $\mathcal{S}$ is smaller than one for problems that are not in resonance, it may be possible to construct polynomial preconditioners of $I - \mathcal{S}$ via a Neumann series or Padé approximations of $\mathcal{S}$ arising from slightly unstable WaveHoltz iterations with a small number of timesteps per iteration. Further, we have not exploited adaptivity in space or time or any ideas from the sweeping preconditioner class of methods.

For the elastic WaveHoltz method, we have thus far only considered energy-conserving problems with Dirichlet and/or free surface boundary conditions. It is clear that further investigation into problems with impedance/absorbing boundary conditions is needed. For DG discretizations, the CFL condition of explicit timesteppers may be restrictive for high order spatial discretizations on fine meshes. Improved methods for the inversion of the matrices for implicit time-corrected schemes could drastically improve the performance and runtime of high order DG methods for elastic problems.

In this thesis, we presented some time-corrected centered schemes for the wave equation in second order form. However, problems with damping or impedance/absorbing boundary conditions require the use of the general WaveHoltz iteration in which the wave equation is solved as a first order system in time. Another possible area of exploration is in devising modified timestepping schemes for these systems. Related to this, we have not explored using spatial discretizations specifically designed to reduce dispersion/pollution errors of the Helmholtz equation together with the WaveHoltz iteration.

Beginning in Chapter 5, we shifted focus from time-domain Helmholtz solvers to the optimal control of quantum system. In this chapter, we considered closed quantum systems governed by

the time-dependent Schrödinger equation where the controls are microwave pulses used to enact a user-specified quantum logic gate. We defined a pair of objective functions measuring the infidelity of the logic gate generated by a given control and a time-average of leakage into higher energy level states of superconducting qubits to design high-fidelity gates. To decouple the timestep size from the size of the parameter space, we introduced a novel basis of B-spline wavelets with carrier waves designed specifically to drive transitions between energy levels in a quantum system. Using a "discretize-then-optimize" approach, we devised a pair of partitioned Runge Kutta schemes to compute *exact* discrete gradients. We demonstated that this approach allows the construction of high-fidelity gates using a small number of parameters for systems of superconducting qubits. We additionally perform a brief study of risk-neutral controls to design controls that are more robust to noise or uncertainty in the system Hamiltonian.

In Chapter 6, we briefly outlined the approach for a submitted solution to the IBM SWAP Gate Challenge as a practical application of the methods of Chapter 5. The goal of the IBM SWAP Gate Challenge was to ask teams of researchers to attempt to reduce the errors of a standard SWAP gate by 50% or more on IBM's Casablanca system. Despite simulated results (both with IBM's simulator and the open-source quantum control toolbox Juqbox.jl) indicating the design of a high-fidelity gate, the experimental results did not meet the desired target set by IBM.

Despite the unsuccessful submission, Chapter 6 reveals many fruitful directions and research questions to explore. One approach is in more thorough and advanced methods for system characterization. As briefly elucidated in Chapter 5, controls that enact high-fidelity gates in a noise-free optimization may quickly degrade in performance with the presence of unaccounted noise in the Hamiltonian. Thus part of the system characterization procedure could benefit from characterization not only of a base Hamiltonian model, but also quantification and identification of noise processes and their distributions. Armed with more accurate Hamiltonians, it would be advantageous to to explore robust and risk-neutral/averse optimization methods to extend the ideas of Chapter 6. This would ultimately result in controls that need to (1) be calibrated less often, and (2) are more resistant to noise.

Bibliography

[1] Julia documentation. https://docs.julialang.org/en/v1/. Accessed: 2019-04-09.

[2] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), pages 265–283, 2016.

[3] Tuomas Airaksinen, Anssi Pennanen, and Jari Toivanen. A damping preconditioner for time-harmonic wave equations in fluid and elastic material. Journal of computational physics, 228(5):1466–1479, 2009.

[4] R. Anderson, J. Andrej, A. Barker, J. Bramwell, J.-S. Camier, J. Cerveny V. Dobrev, Y. Dudouit, A. Fisher, Tz. Kolev, W. Pazner, M. Stowell, V. Tomov, I. Akkerman, J. Dahm, D. Medina, and S. Zampini. MFEM: A modular finite element library. Computers & Mathematics with Applications, 2020.

[5] Laurent Anné, Patrick Joly, and Quang Huy Tran. Construction and analysis of higher order finite difference schemes for the 1D wave equation. Computational Geosciences, 4(3):207–249, 2000.

[6] D. Appelö. Absorbing Layers and Non-Reflecting Boundary Conditions for Wave Propagation Problems. PhD thesis, Royal Institute of Technology, October 2005.

[7] D. Appelö, J. W. Banks, W. D. Henshaw, and D. W. Schwendeman. Numerical methods for solid mechanics on overlapping grids: Linear elasticity. Journal of Computational Physics, 231(18):6012–6050, 2012.

[8] D. Appelö and T. Colonius. A high-order super-grid-scale absorbing layer and its application to linear hyperbolic systems. Journal of Computational Physics, 228(11):4200–4217, 2009.

[9] D. Appelö and T. Hagstrom. A new discontinuous Galerkin formulation for wave equations in second order form. SIAM Journal On Numerical Analysis, 53(6):2705–2726, 2015.

[10] D. Appelö and T. Hagstrom. An energy-based discontinuous Galerkin discretization of the elastic wave equation in second order form. Comput. Meth. Appl. Mech. Engrg., 338:362–391, 2018.

[11] D. Appelö and G. Kreiss. A new absorbing layer for elastic waves. Journal of Computational Physics, 215(2):642–660, 2006.

[12] D. Appelö and N. A. Petersson. A stable finite difference method for the elastic wave equation on complex geometries with free surfaces. Communications in Computational Physics, 5(1):84–107, 2009.

[13] Daniel Appelö, Fortino Garcia, Allen Alvarez Loya, and Olof Runborg. El waveholtz method. in preparation.

[14] Daniel Appelö, Fortino Garcia, and Olof Runborg. WaveHoltz: Iterative solution of the Helmholtz equation via the wave equation. SIAM Journal on Scientific Computing, 42(4):A1950–A1983, 2020.

[15] Anton Arnold, Sjoerd Geevers, Ilaria Perugia, and Dmitry Ponomarev. An adaptive finite element method for high-frequency scattering problems with variable coefficients, 2021.

[16] I. Babu*v*ska and S. Sauter. Is the pollution effect of the fem avoidable for the Helmholtz equation considering high wave numbers? SIAM Journal on Numerical Analysis, 34(6):2392–2423, 1997.

[17] A. H. Baker, Tz. V. Kolev, and U. M. Yang. Improving algebraic multigrid interpolation operators for linear elasticity problems. Numerical Linear Algebra with Applications, 17(2-3):495–517, 2010.

[18] Claude Bardos and Jeffrey Rauch. Variational algorithms for the Helmholtz equation using time evolution and artificial boundaries. Asymptotic analysis, 9(2):101–117, 1994.

[19] Manuel M Baumann. Fast Iterative Solution of the Time-Harmonic Elastic Wave Equation at Multiple Frequencies. PhD thesis, PhD thesis, Delft University of Technology, 2018.

[20] Alvin Bayliss, Charles I Goldstein, and Eli Turkel. An iterative method for the Helmholtz equation. Journal of Computational Physics, 49(3):443–457, 1983.

[21] E. Bécache, S. Fauqueux, and P. Joly. Stability of perfectly matched layers, group velocities and anisotropic waves. J. Comput. Phys., 188:399–433, 2003.

[22] Mikhail Belonosov, Victor Kostin, Dmitry Neklyudov, and Vladimir Tcheverda. 3D numerical simulation of elastic waves with a frequency-domain iterative solver. Geophysics, 83(6):T333–T344, 10 2018.

[23] Hadrien Bériot, Albert Prinn, and Gwénaël Gabard. Efficient implementation of high-order finite elements for helmholtz problems. International Journal for Numerical Methods in Engineering, 106(3):213–240, 2016.

[24] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: A fresh approach to numerical computing. SIAM Review, 59(1):65–89, 2017.

[25] A. Björck. Numerical methods in matrix computations, volume 59. Springer, 2015.

[26] Alexandre Blais, Arne L. Grimsmo, S. M. Girvin, and Andreas Wallraff. Circuit quantum electrodynamics. Rev. Mod. Phys., 93:025005, May 2021.

[27] A. Borzì, G. Ciarmella, and M. Sprengel. Formulation and Numerical Solution of Quantum Control Problems. Computational science and engineering. SIAM, 2017.

[28] A. Brandt and I. Livshits. Wave-ray multigrid method for standing wave equations. Electron. Trans. Numer. Anal, 6(162-181):91, 1997.

[29] M.O. Bristeau, R. Glowinski, and J. Périaux. Controllability methods for the computation of time-periodic solutions; application to scattering. Journal of Computational Physics, 147(2):265–292, 1998.

[30] Romain Brunet, Victorita Dolean, and Martin J. Gander. Can classical Schwarz methods for time-harmonic elastic waves converge? In Domain Decomposition Methods in Science and Engineering XXV, pages 425–432, Cham, 2020. Springer International Publishing.

[31] Romain Brunet, Victorita Dolean, and Martin J Gander. Natural domain decomposition algorithms for the solution of time-harmonic elastic waves. SIAM Journal on Scientific Computing, 42(5):A3313–A3339, 2020.

[32] T. Caneva, T. Calarco, and S. Montangero. Chopped random-basis quantum optimization. Physical Review A, 84(2), Aug 2011.

[33] T. Chaumont-Frelet, M. J. Grote, S. Lanteri, and J. H. Tang. A controllability method for Maxwell's equations, 2021.

[34] Z. Chen and X. Xiang. A source transfer domain decomposition method for Helmholtz equations in unbounded domain. SIAM Journal on Numerical Analysis, 51(4):2331–2356, 2013.

[35] Jonás D De Basabe, Mrinal K. Sen, and Mary F Wheeler. The interior penalty discontinuous Galerkin method for elastic wave propagation: grid dispersion. Geophysical Journal International, 175(1):83–93, 10 2008.

[36] P. de Fouquieres, S.G. Schirmer, S.J. Glaser, and Ilya Kuprov. Second order gradient ascent pulse engineering. Journal of Magnetic Resonance, 212(2):412–417, Oct 2011.

[37] A El Kacimi and Omar Laghrouche. Wavelet based ILU preconditioners for the numerical solution by PUFEM of high frequency elastic wave scattering. Journal of Computational Physics, 230(8):3119–3134, 2011.

[38] L. Emsley and G. Bodenhausen. Gaussian pulse cascades: New analytical functions for rectangular selective inversion and in-phase excitation in NMR. Chem. Phys., 165(6):469–476, 1989.

[39] B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. Math. Comp., 31:629, 1977.

[40] B. Engquist and L. Ying. Sweeping preconditioner for the Helmholtz equation: hierarchical matrix representation. Communications on pure and applied mathematics, 64(5):697–735, 2011.

[41] B. Engquist and L. Ying. Sweeping preconditioner for the Helmholtz equation: moving perfectly matched layers. Multiscale Modeling & Simulation, 9(2):686–710, 2011.

[42] B. Engquist and H. Zhao. Approximate separability of the Green's function of the Helmholtz equation in the high frequency limit. Communications on Pure and Applied Mathematics, 71(11):2220–2274, 2018.

[43] Y. Erlangga, C. Oosterlee, and C. Vuik. A novel multigrid based preconditioner for heterogeneous Helmholtz problems. SIAM Journal on Scientific Computing, 27(4):1471–1492, 2006.

[44] Y.A. Erlangga. Advances in iterative methods and preconditioners for the Helmholtz equation. Archives of Computational Methods in Engineering, 15(1):37–66, 2008.

[45] Yogi A Erlangga, Cornelis Vuik, and Cornelis W Oosterlee. Comparison of multigrid and incomplete LU shifted-Laplace preconditioners for the inhomogeneous Helmholtz equation. Applied numerical mathematics, 56(5):648–666, 2006.

[46] Yogi A Erlangga, Cornelis Vuik, and Cornelis Willebrordus Oosterlee. On a class of preconditioners for solving the Helmholtz equation. Applied Numerical Mathematics, 50(3-4):409–425, 2004.

[47] O.G. Ernst and M.J. Gander. Why it is difficult to solve Helmholtz problems with classical iterative methods. In Numerical analysis of multiscale problems, pages 325–363. Springer, 2012.

[48] B. Ewing, S. J. Glaser, and G. P. Drobny. Development and optimization of shaped NMR pulses for the study of coupled spin systems. Chem. Phys., 147:121–129, 1990.

[49] M. Gander and F. Nataf. AILU for Helmholtz problems: a new preconditioner based on an analytic factorization. Comptes Rendus de l'Académie des Sciences - Series I - Mathematics, 331(3):261–266, 2000.

[50] M. Gander and H. Zhang. A class of iterative solvers for the Helmholtz equation: Factorizations, sweeping preconditioners, source transfer, single layer potentials, polarized traces, and optimized Schwarz methods. SIAM Review, 61(1):3–76, 2019.

[51] Fortino Garcia, Daniel Appelö, and Olof Runborg. Analysis of an iterative solution of the Helmholtz equation via the wave equation for impedance boundary conditions. in preparation.

[52] Fortino Garcia and N. Anders Petersson. Juqbox. Github, 2021.

[53] Fortino Garcia and N. Anders Petersson. Juqbox.jl. Github, 2021.

[54] Xiaozhen Ge, Haijin Ding, Herschel Rabitz, and Re-Bing Wu. Robust quantum control in games: An adversarial learning approach. Physical Review A, 101(5):052317, 2020.

[55] Xiaozhen Ge and Re-Bing Wu. Risk-sensitive optimization for robust quantum controls. arXiv preprint arXiv:2104.01323, 2021.

[56] J. Charles Gilbert and Patrick Joly. Higher Order Time Stepping for Second Order Hyperbolic Problems and Optimal CFL Conditions, pages 67–93. Springer Netherlands, Dordrecht, 2008.

[57] A. Gillman, A.H. Barnett, and P-G. Martinsson. A spectrally accurate direct solution technique for frequency-domain scattering problems with variable media. BIT Numerical Mathematics, 55(1):141–170, Mar 2015.

[58] R. Glowinski and T. Rossi. A mixed formulation and exact controllability approach for the computation of the periodic solutions of the scalar wave equation. (i): Controllability problem formulation and related iterative solution. Comptes Rendus Math., 343(7):493–498, 2006.

[59] G. H. Golub and C. F. Van Loan. Matrix Computations. Johns Hopkins University Press, 1996.

[60] Dan Gordon and Rachel Gordon. Robust and highly scalable parallel solution of the Helmholtz equation with large wave numbers. Journal of Computational and Applied Mathematics, 237(1):182–196, 2013.

[61] M. J. Grote, F. Nataf, J. H. Tang, and P.-H. Tournier. Parallel Controllability Methods For the Helmholtz Equation. arXiv e-prints, page arXiv:1903.12522, Mar 2019.

[62] M. J. Grote, A. Schneebeli, and D. Schötzau. Discontinuous Galerkin finite element method for the wave equation. SIAM Journal on Numerical Analysis, 44(6):2408–2431, 2006.

[63] M.J. Grote and J.H. Tang. On controllability methods for the Helmholtz equation. Journal of Computational and Applied Mathematics, 358:306–326, 2019.

[64] Stefanie Günther and N. Anders Petersson. Quandary: Optimal control for open quantum systems. `https://github.com/LLNL/quandary`, 2021.

[65] William W. Hager. Runge-Kutta methods in optimal control and the transformed adjoint system. Numerische Mathematik, 87(2):247–282, Dec 2000.

[66] T. Hagstrom. Radiation boundary conditions for the numerical simulation of waves. Acta Numerica, 8:47–106, 1999.

[67] E. Hairer, C. Lubich, and G. Wanner. Geometric Numerical Integration. Number 31 in Springer series in computational mathematics. Springer-Verlag, Heidelberg, 2nd edition, 2006.

[68] E. Heikkola, S. Mönkölä, A. Pennanen, and T. Rossi. Controllability method for acoustic scattering with spectral elements. Journal of Computational and Applied Mathematics, 204(2):344–355, 2007.

[69] E. Heikkola, S. Mönkölä, A. Pennanen, and T. Rossi. Controllability method for the Helmholtz equation with higher-order discretizations. Journal of Computational Physics, 225(2):1553–1576, 2007.

[70] C. V. Hile and G. A Kriegsmann. A hybrid numerical method for loaded highly resonant single mode cavities. Journal of Computational Physics, 142(2):506–520, 1998.

[71] J.R. Johansson, P.D. Nation, and Franco Nori. Qutip 2: A python framework for the dynamics of open quantum systems. Computer Physics Communications, 184(4):1234 – 1240, 2013.

[72] D. Ketcheson, L. Lóczi, and T. Kocsis. On the absolute stability regions corresponding to partial sums of the exponential function. IMA Journal of Numerical Analysis, 35(3):1426–1455, 2015.

[73] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbruggen, and S. Glaser. Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms. J. Magnetic Resonance, 172:296–305, 2005.

[74] Q Kong and A Zettl. Eigenvalues of regular Sturm–Liouville problems. Journal of differential equations, 131(1):1–19, 1996.

[75] H.-O. Kreiss and J. Oliger. Comparison of accurate methods for the integration of hyperbolic equations. Tellus, 24:199–215, 1972.

[76] O. A. Ladyzhenskaya. On the limiting-amplitude principle. Uspekhi Mat. Nauk, 12(4):161–164, 1957.

[77] Alistair L Laird and M Giles. Preconditioned iterative solution of the 2D Helmholtz equation. 2002.

[78] N. Leung, M. Abdelhafez, Jens Koch, and D. Schuster. Speedup for quantum optimal control from automatic differentiation based on graphics processing units. Phys. Rev. A, 95:0432318, 2017.

[79] Randall J. LeVeque. Finite volume methods for hyperbolic problems. Cambridge University Press, Cambridge, 2002.

[80] Yang Li, Ludovic Métivier, Romain Brossier, Bo Han, and Jean Virieux. 2d and 3d frequency-domain elastic wave modeling in complex media with a parallel iterative solver. Geophysics, 80(3):T101–T118, 2015.

[81] I. Livshits and A. Brandt. Accuracy properties of the Wave-ray multigrid algorithm for Helmholtz equations. SIAM Journal on Scientific Computing, 28(4):1228–24, 2006.

[82] Dennis Lucarelli. Quantum optimal control via gradient ascent in function space and the time-bandwidth quantum speed limit. Physical Review A, 97(6), Jun 2018.

[83] S. Machnes, E. Assémat, D. Tannor, and F. K. Wilhelm. Tunable, flexible, and efficient optimization of control pulses for practical qubits. Physical Review Letters, 120(15), Apr 2018.

[84] E. Magesan and J. M. Gambetta. Effective Hamiltonian models of the cross-resonance gate. Phys. Rev. A, 101, 2020.

[85] Marcus J. Grote, Frédéric Nataf, Jet Hoe Tang, and Pierre-Henri Tournier. Parallel controllability methods for the Helmholtz equation. Computer Methods in Applied Mechanics and Engineering, 362:112846, 2020.

[86] K. Mattsson. Summation by parts operators for finite difference approximations of second-derivatives with variable coefficients. Journal of Scientific Computing, 51(3):650–682, Jun 2012.

[87] Sanna Mönkölä, Erkki Heikkola, Anssi Pennanen, and Tuomo Rossi. Time-harmonic elasticity with controllability and higher-order discretization methods. Journal of Computational Physics, 227(11):5513–5534, 2008.

[88] C.S. Morawetz. The limiting amplitude principle. Communications on Pure and Applied Mathematics, 15(3):349–361, 1962.

[89] K. W. Morton and D. F. Mayers. Numerical Solution of Partial Differential Equations: An Introduction. Cambridge University Press, 2 edition, 2005.

[90] M. Nielsen and I. Chuang. Quantum computation and quantum information. Cambridge University Press, 2000.

[91] J. Nocecdal and S. J. Wright. Numerical Optimization. Springer, 2nd edition, 2006.

[92] Sina Ober-Blöbaum. Discrete mechanics and optimal control. PhD thesis, University of Paderborn, 2008.

[93] Zhichao Peng and Daniel Appelö. EM-Waveholtz: A flexible frequency-domain method built from time-domain solvers. arXiv preprint arXiv:2103.14789, 2021.

[94] N. Anders Petersson and Fortino Garcia. Optimal control of closed quantum systems via B-splines with carrier waves. Submitted, 2021.

[95] R. E. Plessix and W. A. Mulder. Separation-of-variables as a preconditioner for an iterative Helmholtz solver. Applied numerical mathematics, 44(3):385–400, 2003.

[96] G. Rizzuti and W.A. Mulder. Multigrid-based 'shifted-Laplacian' preconditioning for the time-harmonic elastic wave equation. Journal of Computational Physics, 317:47–65, 2016.

[97] J. M. Sanz-Serna. Symplectic Runge–Kutta schemes for adjoint equations, automatic differentiation, optimal control, and more. SIAM Review, 58(1):3–33, 2016.

[98] Yunong Shi, Nelson Leung, Pranav Gokhale, Zane Rossi, David I. Schuster, Henry Hoffmann, and Frederic T. Chong. Optimized compilation of aggregated instructions for realistic quantum computers. Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems - ASPLOS '19, 2019.

[99] Gregory R Shubin and John B Bell. A modified equation approach to constructing fourth order methods for acoustic wave propagation. SIAM Journal on Scientific and Statistical Computing, 8(2):135–151, 1987.

[100] C.C. Stolk. A rapidly converging domain decomposition method for the Helmholtz equation. Journal of Computational Physics, 241:240–252, 2013.

[101] Christiaan C. Stolk. A time-domain preconditioner for the Helmholtz equation, 2020.

[102] Eran Treister. Shifted Laplacian multigrid for the elastic Helmholtz equation, 2018.

[103] Paul Tsuji, Jack Poulson, Björn Engquist, and Lexing Ying. Sweeping preconditioners for elastic wave propagation with spectral element methods. ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique, 48(2):433–447, 2014.

[104] B. R. Vainberg. On short-wave asymptotic behaviour of solutions to steady-state problems and the asymptotic behaviour as $t \to \infty$ of solutions of time-dependent problems. Uspekhi Mat. Nauk, 30(2):1–58, 1975.

[105] A. Vion and C. Geuzaine. Double sweep preconditioner for optimized Schwarz methods applied to the Helmholtz problem. Journal of Computational Physics, 266:171–190, 2014.

[106] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical Programming, 106(1):25–57, Mar 2006.

[107] S. Wang, M.V. de Hoop, and J. Xia. On 3d modeling of seismic wave propagation via a structured parallel multifrontal direct Helmholtz solver. Geophysical Prospecting, 59(5):857–873, 2011.

[108] Shen Wang, Maarten V de Hoop, Jianlin Xia, and Xiaoye S Li. Massively parallel structured multifrontal solver for time-harmonic elastic waves in 3-D anisotropic media. Geophysical Journal International, 191(1):346–366, 2012.

[109] H. Weyl. Über die asymptotische verteilung der eigenwerte. Nachr. Konigl. Ges. Wiss., pages 110–117, 1911.

[110] D. B. Zax, G. Goelman, and S. Vega. Amplitude-modulated composite pulses. J. Magn. Reson., 80(2):375–382, 1988.

[111] L. Zepeda-Núñez, A. Scheuer, R. J. Hewett, and L. Demanet. The method of polarized traces for the 3D Helmholtz equation. ArXiv e-prints, January 2018.

[112] L. Zepeda-Núñez and L. Demanet. The method of polarized traces for the 2d Helmholtz equation. Journal of Computational Physics, 308:347–388, 2016.

[113] L. Zepeda-Núñez and L. Demanet. Nested domain decomposition with polarized traces for the 2d Helmholtz equation. SIAM Journal on Scientific Computing, 40(3):B942–B981, 2018.

.1 Proof of Lemma 2.1.1

Proof. We show the results for the rescaled transfer function

$$\bar{\beta}(r) := \beta(r\omega) = \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right)\cos(r\omega t)dt = \frac{1}{\pi}\int_0^{2\pi}\left(\cos(t) - \frac{1}{4}\right)\cos(rt)dt.$$

By direct integration we get

$$\bar{\beta}(r) = \frac{1}{\pi}\int_0^{2\pi}\frac{1}{2}(\cos((r+1)t) + \cos((r-1)t)) - \frac{1}{4}\cos(rt)dt = \tag{1}$$

$$\frac{1}{2\pi}\left(\frac{\sin(2\pi(r+1))}{r+1} + \frac{\sin(2\pi(r-1))}{r-1} - \frac{1}{2}\frac{\sin(2\pi r)}{r}\right) = \mathrm{sinc}(r+1) + \mathrm{sinc}(r-1) - \frac{1}{2}\mathrm{sinc}(r), \tag{2}$$

where

$$\mathrm{sinc}(r) = \frac{\sin(2\pi r)}{2\pi r}.$$

We use the fact that $\sin(x) \leq x - \tilde{\alpha}x^3$ in the interval $x \in [0, \pi]$ for any $\tilde{\alpha} \in [0, \pi^{-2}]$. This leads to the following estimate for the sinc function

$$0 \leq \mathrm{sinc}(r) \leq 1 - \alpha r^2, \qquad r \in [-0.5,\ 0.5], \qquad \alpha \in [0,\ 4]. \tag{3}$$

We also note that $\operatorname{sinc}(r+n) = \operatorname{sinc}(r)r/(r+n)$ for all integer n.

We now first consider $0 \le r \le 0.5$ and use (3) with $\alpha = 4$ and $\alpha = 0$,

$$|\bar{\beta}(r)| = \operatorname{sinc}(r)\left|\frac{r}{r+1} + \frac{r}{r-1} - \frac{1}{2}\right| = \frac{1}{2}\operatorname{sinc}(r)\frac{1+3r^2}{1-r^2} \le \frac{1}{2}\frac{1-4r^2+3r^2}{1-r^2} = \frac{1}{2}.$$

For $0.5 \le r \le 1.5$ we instead center around $r = 1$ and get for $|\delta| \le 0.5$,

$$\bar{\beta}(1+\delta) = \operatorname{sinc}(\delta)\frac{3(\delta+1)^2+1}{2(2+\delta)(1+\delta)} \ge 0,$$

since $\operatorname{sinc}(\delta) \ge 0$. Moreover, using again (3) with $\alpha = 1$,

$$\bar{\beta}(1+\delta) \le (1-\delta^2)\frac{3(\delta+1)^2+1}{2(2+\delta)(1+\delta)} = \frac{4+2\delta-3\delta^2-3\delta^3}{2(2+\delta)} \le \frac{4+2\delta-2\delta^2-\delta^3}{2(2+\delta)} = 1 - \frac{\delta^2}{2}.$$

Finally, for $r > 1$ we have $1/(r+1) - 1/2r \ge 0$ and therefore

$$|\bar{\beta}(r)| = \frac{|\sin(2\pi r)|}{2\pi r}\left(\frac{r}{r+1} + \frac{r}{r-1} - \frac{1}{2}\right) \le \frac{1}{2\pi}\left(\frac{1}{r+1} + \frac{1}{r-1} - \frac{1}{2r}\right) \le \frac{3}{4\pi}\frac{1}{(r-1)}.$$

We note also that for $r \ge 1.5$ this gives $|\bar{\beta}(r)| \le 3/2\pi \le 1/2$.

To prove (2.13) we use the Taylor expansion of $\bar{\beta}$ around $r = 1$,

$$\bar{\beta}(1+\delta) = 1 + \frac{\delta^2}{2}\bar{\beta}''(1) + \frac{\delta^3}{6}\bar{R}(\delta),$$

where $\bar{R}(\delta)$ is the remainder term, which can be bounded as

$$|\bar{R}(\delta)| \le \sup_{r\ge 0}\left|\bar{\beta}^{(3)}(r)\right| \le \frac{1}{\pi}\int_0^{2\pi} t^3\left(1+\frac{1}{4}\right)dt = 5\pi^3.$$

Hence, $|R(\delta)| \le |\bar{R}(\delta)|/6 \le 5\pi^3/6$. Finally,

$$\beta''(1) = \operatorname{sinc}''(2) + \operatorname{sinc}''(0) - \frac{1}{2}\operatorname{sinc}''(1) = \frac{-1}{2} - \frac{(2\pi)^2}{3} + 1 = -2b_1.$$

This shows (2.13) and concludes the proof of the lemma. □

.2 Verification of Discrete Solution

Here we verify that (2.19) is indeed a solution to the difference equation (2.18). Direct substitution yields

$$\begin{aligned}
\hat{w}_j^{n+1} - 2\hat{w}_j^n + \hat{w}_j^{n-1} + \Delta t^2 \lambda_j^2 \hat{w}_j^n &= (\hat{v}_j - \hat{v}_j^\infty)\cos(\tilde{\lambda}_j t_n)\left(\cos(\tilde{\lambda}_j \Delta t) - 2 + \Delta t^2 \lambda_j^2 + \cos(\tilde{\lambda}_j \Delta t)\right) \\
&\quad + \hat{v}_j^\infty \cos(\omega t_n)\left(\cos(\omega \Delta t) - 2 + \Delta t^2 \lambda_j^2 + \cos(\omega \Delta t)\right) \\
&= (\hat{v}_j - \hat{v}_j^\infty)\cos(\tilde{\lambda}_j t_n)\left(-4\sin^2(\tilde{\lambda}_j \Delta t/2) + \Delta t^2 \lambda_j^2)\right) \\
&\quad + \hat{v}_j^\infty \cos(\omega t_n)\left(-4\sin^2(\omega \Delta t/2) + \Delta t^2 \lambda_j^2\right) \\
&= \hat{v}_j^\infty \cos(\omega t_n)\left(-\tilde{\omega}^2 + \lambda_j^2\right)\Delta t^2 = -\Delta t^2 f_j \cos(\omega t_n).
\end{aligned}$$

Second, the initial conditions are satisfied since

$$\begin{aligned}
\hat{w}_j^0 &= (\hat{v}_j - \hat{v}_j^\infty) + \hat{v}_j^\infty = \hat{v}_j, \\
\hat{w}_j^{-1} &= (\hat{v}_j - \hat{v}_j^\infty)\cos(\tilde{\lambda}_j \Delta t) + \hat{v}_j^\infty \cos(\omega \Delta t) = (\hat{v}_j - \hat{v}_j^\infty)\left(1 - \frac{1}{2}\Delta t^2 \lambda_j^2\right) + \hat{v}_j^\infty \left(1 - \frac{1}{2}\Delta t^2 \tilde{\omega}^2\right) \\
&= \hat{v}_j \left(1 - \frac{1}{2}\Delta t^2 \lambda_j^2\right) + \frac{1}{2}\Delta t^2 \hat{v}_j^\infty \left(\lambda_j^2 - \tilde{\omega}^2\right) = \hat{v}_j \left(1 - \frac{1}{2}\Delta t^2 \lambda_j^2\right) - \frac{1}{2}\Delta t^2 \hat{f}_j.
\end{aligned}$$

This shows that (2.19) solves (2.18).

.3 Proof of Lemma 2.1.5

Proof. In general, we introduce the trapezoidal rule applied to $\cos(\alpha t)$ in $[0, 1]$,

$$\mathcal{T}_h(\alpha) := h \sum_{n=0}^{M} \eta_n \cos(\alpha t_n) \approx \int_0^1 \cos(\alpha t) dt = \frac{\sin(\alpha)}{\alpha}, \qquad h = 1/M,$$

from which we attain the following lemma:

Lemma .3.1. *The error in $\mathcal{T}_h(\alpha)$ satisfies*[1]

$$\left| \int_0^1 \cos(\alpha t) dt - \mathcal{T}_h(\alpha) \right| \leq \frac{h^2 |\alpha|}{\pi^2}, \qquad \textit{when } |h\alpha| \leq \pi.$$

[1] Note that this estimate is sharper than the standard error estimate for the trapezoidal rule, which would have the factor α^2 from the second derivative of the integrand, not just α.

Proof. A direct calculation shows that

$$\mathcal{T}_h(\alpha) = g(h\alpha)\int_0^1 \cos(\alpha t)dt, \qquad g(x) = \frac{x}{2\tan(x/2)}.$$

The function $g(x)$ can be bounded as $1 - x^2/\pi^2 \le g(x) \le 1$ for $|x| \le \pi$. This gives

$$\left|\int_0^1 \cos(\alpha t)dt - \mathcal{T}_h(\alpha)\right| = |1 - g(h\alpha)| \left|\frac{\sin\alpha}{\alpha}\right| \le \frac{(h\alpha)^2}{\pi^2}\left|\frac{\sin\alpha}{\alpha}\right| \le \frac{h^2|\alpha|}{\pi^2}.$$

3049 □

Since

$$\left(\cos(\omega t) - \frac{1}{4}\right)\cos(\lambda t) = \frac{1}{2}\left(\cos((\omega+\lambda)t) + \cos((\omega-\lambda)t) - \frac{1}{2}\cos(\lambda t)\right),$$

and $h = \Delta t/T$, we can write

$$\begin{aligned}\beta_h(\lambda) &= \frac{\Delta t}{T}\sum_{n=0}^{M}\eta_n\left(\cos((\omega+\lambda)t_n) + \cos((\omega-\lambda)t_n) - \frac{1}{2}\cos(\lambda t_n)\right)\\ &= \left[\mathcal{T}_h(T(\omega+\lambda)) + \mathcal{T}_h(T(\omega-\lambda)) - \frac{1}{2}\mathcal{T}_h(T\lambda)\right].\end{aligned}$$

From Lemma .3.1 we then get that

$$|\beta(\tilde\lambda_j) - \beta_h(\tilde\lambda_j)| \le \frac{h^2}{\pi^2}\left(|T(\omega+\tilde\lambda_j)| + |T(\omega-\tilde\lambda_j)| + \frac{1}{2}|T\tilde\lambda_j|\right) \le \frac{5h^2T}{2\pi^2}(\omega+\tilde\lambda_j) = \frac{5\Delta t^2}{2\pi^2 T}(\omega+\tilde\lambda_j),$$

when

$$\pi \ge hT(\omega+\tilde\lambda_j) = \Delta t(\omega+\tilde\lambda_j),$$

3050 which is true by (3.25) and the fact that $\arcsin(x) \le \pi x/2$ for $x \in [0,1]$:

$$\tilde\lambda_j = \frac{2}{\Delta t}\arcsin\left(\frac{\Delta t\lambda_j}{2}\right) \le \frac{\pi}{2}\lambda_j. \tag{4}$$

3051 Next, we use the inequality $|x - \sin(x)| \le x^3/6$ for $|x| \le \pi/2$ to show that

$$\left|\frac{\sin(xh)}{h} - x\right| = \frac{1}{h}\left|\sin(xh) - xh\right| \le \frac{(xh)^3}{6h} = \frac{h^2x^3}{6}, \qquad |hx| \le \frac{\pi}{2}. \tag{5}$$

It gives us an estimate for $\tilde\lambda_j - \lambda_j$,

$$|\lambda_j - \tilde\lambda_j| = \left|\frac{\sin\left(\Delta t\tilde\lambda_j/2\right)}{\Delta t/2} - \tilde\lambda_j\right| \le \frac{\Delta t^2}{24}\tilde\lambda_j^3,$$

which is valid for all j since $\Delta t\tilde{\lambda}_j/2 \le \Delta t\pi\lambda_j/4 \le \Delta t\pi\lambda_N/4 \le \pi/2$, by (3.25) and (4).

By Lemma 2.1.1

$$|\beta(\omega+r)| \le \begin{cases} 1-\frac{r^2}{2\omega^2}, & |r/\omega| \le \frac{1}{2}, \\ \\ \frac{1}{2}, & |r/\omega| \ge \frac{1}{2}. \end{cases}$$

We now claim that the statement (2.20) in the lemma holds for all j if $\Delta t\omega \le \min(\delta_h, 1)$. On the one hand, if $|\omega - \tilde{\lambda}_j| \ge \omega/2$, by (3.25) and (4)

$$|\beta_h(\tilde{\lambda}_j)| \le |\beta(\tilde{\lambda}_j)| + \frac{5\Delta t^2}{2\pi^2 T}(\omega+\tilde{\lambda}_j) \le \frac{1}{2} + \frac{5\Delta t^2}{2\pi^2 T}(\omega+\tilde{\lambda}_j) = \frac{1}{2} + \frac{5\Delta t\omega}{4\pi^3}\Delta t(\omega+\tilde{\lambda}_j) \le \frac{1}{2} + \frac{5\Delta t\omega}{4\pi^2} \le 0.63.$$

On the other hand, if $|\omega - \tilde{\lambda}_j| < \omega/2$,

$$\frac{|\omega-\tilde{\lambda}_j|}{\omega} \ge \frac{|\omega-\lambda_j|}{\omega} - \frac{|\tilde{\lambda}_j-\lambda_j|}{\omega} \ge \delta_h - \frac{\Delta t^2}{24\omega}\tilde{\lambda}_j^3 \ge \delta_h - \frac{\Delta t^2}{24\omega}(\omega+|\tilde{\lambda}_j-\omega|)^3 \ge \delta_h - \frac{(3/2)^3}{24}\Delta t^2\omega^2 \ge \frac{55}{64}\delta_h,$$

since $\min(\delta_h, 1)^2 \le \delta_h$. Then

$$\begin{aligned} |\beta_h(\tilde{\lambda}_j)| &\le |\beta(\tilde{\lambda}_j)| + \frac{5\Delta t^2}{48T}(\omega+\tilde{\lambda}_j) \le 1 - \frac{1}{2}\left(\frac{|\omega-\tilde{\lambda}_j|}{\omega}\right)^2 + \frac{5\Delta t^2\omega}{96\pi}(\omega+\omega+\omega/2) \\ &\le 1 - \frac{55^2}{2\cdot 64^2}\delta_h^2 + \Delta t^2\omega^2\frac{25}{192\pi} \le 1 - \left(\frac{55^2}{2\cdot 64^2} - \frac{25}{192\pi}\right)\delta_h^2 \le 1 - 0.3\delta_h^2. \end{aligned}$$

This proves the lemma.

□

.4 Proof of Lemma 3.1.1

We show the results for the rescaled function

$$\bar{\gamma}(r) := \gamma(r\omega) = \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right)\sin(r\omega t)\,dt = \frac{1}{\pi}\int_0^{2\pi}\left(\cos(t) - \frac{1}{4}\right)\sin(rt)\,dt.$$

By direct integration we get

$$\bar{\gamma}(r) = \frac{1}{\pi}\int_0^{2\pi}\frac{1}{2}\left(\sin((r+1)t) + \sin((r-1)t)\right) - \frac{1}{4}\sin(rt)\,dt = \frac{(1+3r^2)\sin^2(\pi r)}{2\pi r(r^2-1)} = \frac{\pi r(1+3r^2)\mathrm{sinc}^2(r/2)}{2(r^2-1)},$$

where

$$\mathrm{sinc}(r) = \frac{\sin(2\pi r)}{2\pi r}.$$

From [14] we have the following expression for β:

$$\bar{\beta}(r) = \frac{1}{\pi}\int_0^{2\pi} \frac{1}{2}\left(\cos((r+1)t) + \cos((r-1)t)\right) - \frac{1}{4}\cos(rt)\,dt = \frac{(1+3r^2)\sin(2\pi r)}{4\pi r(r^2-1)} = \frac{(1+3r^2)\text{sinc}(r)}{2(r^2-1)}.$$

Then the eigenvalues of the WaveHoltz operator applied to the first order system are

$$|\bar{\mu}(r)|^2 = \bar{\beta}^2(r) + \bar{\gamma}^2(r) = \frac{(1+3r^2)^2\sin^2(\pi r)}{4\pi^2 r^2(r^2-1)^2} = \frac{(1+3r^2)^2\text{sinc}^2(r/2)}{4(r^2-1)^2}.$$

We now first consider $0 \le r \le 0.5$ and note that $|\mu(r)|^2$ is a positive, increasing function on this interval so that

$$|\bar{\mu}(r)|^2 \le |\bar{\mu}(1/2)| = \frac{49}{9\pi^2} \le 0.56.$$

For $1/2 \le r \le 3/2$ we instead center around $r = 1$ and get for $|\delta| \le 1/2$,

$$|\bar{\mu}(1+\delta)|^2 = \frac{(3(\delta+1)^2+1)^2\sin^2(\pi\delta)}{4\pi^2\delta^2(1+\delta)^2(2+\delta)^2} = \frac{(3(\delta+1)^2+1)^2\text{sinc}^2(\delta/2)}{4(1+\delta)^2(2+\delta)^2}.$$

We use the fact that $\sin(x) \le x - \tilde{\alpha}x^3$ in the interval $x \in [0,\pi]$ for any $\tilde{\alpha} \in [0, \pi^{-2}]$. This leads to the following estimate for the sinc function

$$0 \le \text{sinc}(r/2) \le 1 - \alpha r^2, \quad r \in [-0.5, 0.5], \quad \alpha \in [0,1]. \tag{6}$$

Using (6) with $\alpha = 1$, gives

$$\begin{aligned}|\bar{\mu}(1+\delta)|^2 \le \frac{(3(\delta+1)^2+1)^2(1-\delta^2)^2}{4(1+\delta)^2(2+\delta)^2} = \frac{(4+2\delta-3\delta^2-3\delta^3)^2}{4(2+\delta)^2} \le \frac{(4+2\delta-2\delta^2-\delta^3)^2}{4(2+\delta)^2} &= \left(1-\frac{\delta^2}{2}\right)^2 \\ &= 1-\delta^2+\frac{\delta^4}{4} \\ &\le 1-\frac{15}{16}\delta^2,\end{aligned}$$

since $|\delta| < 1/2$. A Taylor expansion around $\delta = 0$ for $|\delta| \le 1/2$ immediately gives the bound

$$\sqrt{1-\delta^2} \le 1 - \frac{\delta^2}{2} \implies |\mu(1+\delta)| \le \sqrt{1-\frac{15}{16}\delta^2} \le 1 - \frac{15\delta^2}{32}.$$

If we consider $r \ge 3/2$,

$$|\bar{\mu}(r)|^2 = \frac{(1+3r^2)^2\text{sinc}^2(r/2)}{4(r^2-1)^2} \le \frac{(1+3r^2)^2}{4(r^2-1)^2},$$

which is a positive and decreasing function. It follows that

$$|\bar{\mu}(r)|^2 \leq \frac{(1+3(3/2)^2)^2}{4((3/2)^2-1)^2} \leq 0.44,$$

for $r \geq 3/2$. Finally, for a more general bound for $r > 1$ we have $1/(r+1) - 1/2r \geq 0$ so that

$$|\bar{\mu}(r)|^2 = \frac{(1+3r^2)^2 \sin^2(\pi r)}{4\pi^2 r^2 (r^2-1)^2} \leq \frac{(1+3r^2)^2}{4\pi^2 r^2 (r^2-1)^2} = \frac{1}{\pi^2}\left(\frac{1}{r+1} + \frac{1}{r-1} - \frac{1}{2r}\right)^2 \leq \left(\frac{3}{2\pi(r-1)}\right)^2,$$

which gives

$$|\bar{\mu}(r)| \leq \frac{3}{2\pi(r-1)}.$$

To prove (3.13), we use a Taylor expansion of $\bar{\mu}(r)$ about $r = 1$ in the interval $r \in (1/2, 3/2)$,

$$|\bar{\mu}(1+\delta)| = 1 + \frac{\delta^2}{2}\frac{d^2}{dr^2}\left[|\bar{\mu}(r)|\right]_{r=1} + \frac{\delta^3}{6}\bar{R}(\delta),$$

where $\bar{R}(\delta)$ is the remainder term. We note that by product rule we have

$$\frac{d}{dr}|\bar{\mu}(r)| = \frac{1}{|\bar{\mu}|}(\bar{\beta}\bar{\beta}' + \bar{\gamma}\bar{\gamma}').$$

Since

$$\frac{d}{dr}|\bar{\mu}(r)|^{-s} = -s|\bar{\mu}(r)|^{-s-1}\frac{d}{dr}|\bar{\mu}(r)| = \frac{-s}{|\bar{\mu}(r)|^{s+2}}(\bar{\beta}\bar{\beta}' + \bar{\gamma}\bar{\gamma}'),$$

by repeated product rule we can then show that

$$\begin{aligned}\frac{d^3}{dr^3}|\bar{\mu}(r)| = {} & \frac{3}{|\bar{\mu}|^5}(\bar{\beta}\bar{\beta}' + \bar{\gamma}\bar{\gamma}')^2 - \frac{1}{|\bar{\mu}|^3}(\bar{\beta}\bar{\beta}'' + (\bar{\beta}')^2 + \bar{\gamma}\bar{\gamma}'' + (\bar{\gamma}')^2)(1 + \bar{\beta}\bar{\beta}' + \bar{\gamma}\bar{\gamma}') \\ & + \frac{1}{|\bar{\mu}|}(\bar{\beta}\bar{\beta}''' + 3\bar{\beta}'\bar{\beta}'' + \bar{\gamma}\bar{\gamma}''' + 3\bar{\gamma}'\bar{\gamma}'').\end{aligned}$$

We note that $|\bar{\mu}(r)| \geq |\bar{\mu}(3/2)| \geq 1/\pi$ in the interval $1/2 \leq r \leq 3/2$, and that we have the following bound

$$\sup_{r \geq 0}\left|\bar{\beta}^{(s)}(r)\right| \leq \frac{1}{\pi}\int_0^{2\pi} t^s \left(1 + \frac{1}{4}\right) dt = 5\frac{2^{s-1}\,\pi^s}{s+1},$$

which similarly holds for $\sup_{r\geq 0}\left|\bar{\gamma}^{(s)}(r)\right|$ for $s = 0, 1, 2, \ldots$. Thus by Taylor's theorem we have

$$\begin{aligned}|\bar{R}(\delta)| \leq \sup_{1/2\leq r\leq 3/2}\left|\frac{d^3}{dr^3}|\bar{\mu}(r)|\right| &\leq \frac{3}{|\bar{\mu}(3/2)|^5}\frac{25^2\pi^2}{4} + \frac{3}{|\bar{\mu}(3/2)|^3}\left(\frac{50\pi^2}{3}+\frac{25\pi^2}{2}\right)\left(1+\frac{25\pi}{2}\right) + \frac{3\cdot 75\pi^3}{|\bar{\mu}(3/2)|} \\ &\leq \frac{3}{4}25^2\pi^7 + 3\pi^3\left(\frac{50\pi^2}{3}+\frac{25\pi^2}{2}\right)\left(1+\frac{25\pi}{2}\right) + 3\cdot 75\pi^4 \\ &= \frac{25\pi^4}{4}\left(36+20\pi+250\pi^2+75\pi^3\right).\end{aligned}$$

Then, $|R(\delta)| \leq |\bar{R}(\delta)|/6$. Finally,

$$\frac{d^2}{dr^2}\left[|\bar{\mu}(r)|\right]_{r=1} = \frac{1}{6}(3-2\pi^2) = -2b_1.$$

.5 Wave Equation Extension

Let $\Omega = (-\infty, 0)$ and let $f \in L^2(\Omega)$ be compactly supported in Ω away from $x = 0$. Additionally, assume $1/c^2 \in L^1_{\text{loc}}(\Omega)$ with $c(0) = c_0$ on the interval $[-\delta, 0]$ for some $\delta > 0$. We consider the semi-infinite problem

$$\begin{aligned}&w_{tt} = \frac{\partial}{\partial x}\left[c^2(x)\frac{\partial}{\partial x}w\right] - \text{Re}\{f(x)e^{i\omega t}\}, \quad x \leq 0, \quad 0 \leq t \leq T, \\ &w(0,x) = v_0(x), \quad w_t(0,x) = v_1(x), \\ &\alpha w_t(t,0) + \beta c_0 w_x(t,0) = 0,\end{aligned}$$

Let $\tilde{w}$ solve the extended wave equation

$$\begin{aligned}&\tilde{w}_{tt} = \frac{\partial}{\partial x}\left[\tilde{c}^2(x)\frac{\partial}{\partial x}\tilde{w}\right] - \text{Re}\{\tilde{f}(x)e^{i\omega t}\}, \quad x \in \mathbb{R}, \quad 0 \leq t \leq T, \\ &\tilde{w}(0,x) = \tilde{v}_0(x), \quad \tilde{w}_t(0,x) = \tilde{v}_1(x), \\ &\alpha\tilde{w}_t(t,0) + \beta c_0 \tilde{w}_x(t,0) = 0.\end{aligned}$$

where $\tilde{f}$ is a zero extension, $\tilde{c}$ is the extended wavespeed

$$\tilde{c}(x) = \begin{cases} c_0, & -\delta < x \leq 0, \\ \tilde{c}_0, & x > 0, \end{cases}$$

and (**I**) $\tilde{v}_0 \in H^1$ and $\tilde{v}_1 \in L^2$. We then choose the extension of v_0 and v_1 such that

$$\tilde{v}_1(x) + \tilde{v}_0'(x) = 0, \quad x > 0, \quad (\mathbf{II})$$

which will have purely right-going waves in the extended region $x > 0$. Moreover, since c is constant in $[-\delta, 0]$ the solution will then be of the form

$$\tilde{w}(t,x) = \begin{cases} w_L(x + c_0 t) + w_R(x - c_0 t), & -\delta \le x \le 0, \\ w_T(x - \tilde{c}_0 t), & x > 0, \end{cases}$$

for some functions w_L, w_R, and w_T. At $x = 0$ where $\tilde{c}$ is (potentially) discontinuous, the weak solution satisfies the interface conditions that $\tilde{w}$ and $\tilde{c}^2 \tilde{w}_x$ are both continuous. These requirements lead to the relations

$$w_L(c_0 t) + w_R(-c_0 t) = w_T(-\tilde{c}_0 t),$$
$$c_0^2 (w_L'(c_0 t) + w_R'(-c_0 t)) = \tilde{c}_0^2 w_T'(-c_0 t).$$

It follows that

$$\tilde{w}_t(t, 0^-) = c_0(w_L'(c_0 t) - w_R(-c_0 t)) = -\tilde{c}_0 w_T'(-\tilde{c}_0 t), \quad c_0 \tilde{w}_x(t, 0^-) = \frac{\tilde{c}_0^2}{c_0} w_T'(-c_0 t),$$

so that the impedance condition

$$\alpha \tilde{w}_t(t, 0^-) + \beta c_0 \tilde{w}_x(t, 0^-) = \left(-\alpha \tilde{c}_0 + \beta \frac{\tilde{c}_0^2}{c_0} \right) w_T'(-c_0 t) = 0,$$

is satisfied if

$$\tilde{c}_0 = \frac{\alpha}{\beta} c_0. \quad (\mathbf{III})$$

With this choice of the extended wavespeed $\tilde{c}_0$, both $\tilde{w}$ and w satisfy the same PDE and condition at $x = 0$ so that they must be equal for $x < 0$. In summary, if we have that conditions (**I**-**III**) are satisfied, we have that $\tilde{w}(t,x) = w(t,x)$ for $x < 0$. We note that a similar argument can be made for an interior impedance problem on a bounded domain, $a \le x \le b$, to a problem on $\mathbb{R}$. In

this case, assuming $c(a) = c_a$, $c(b) = c_b$ where c is constant near the endpoints, then the following problem has $\tilde{w}(t,x) = w(t,x)$ for $a \le x \le b$:

$$\begin{aligned}
&\tilde{w}_{tt} = \frac{\partial}{\partial x}\left[\tilde{c}^2(x)\frac{\partial}{\partial x}\tilde{w}\right] - \mathrm{Re}\{\tilde{f}(x)e^{-i\omega t}\}, \quad x \in \mathbb{R}, \quad 0 \le t \le T,\\
&\tilde{w}(0,x) = \tilde{v}_0(x), \quad \tilde{w}_t(0,x) = \tilde{v}_1(x),
\end{aligned}$$

where $\tilde{v}_0$ and $\tilde{c}$ are the constant extensions (with $\gamma = \alpha/\beta$)

$$\tilde{v}_0(x) = \begin{cases} v_0(a_0), & x < a, \\ v_0(x), & a \le x \le b, \\ v_0(b_0), & b < x, \end{cases} \qquad \tilde{c}(x) = \begin{cases} \gamma c_a, & x < a, \\ c(x), & a \le x \le b, \\ \gamma c_b, & b < x, \end{cases}$$

and $\tilde{v}_1$, $\tilde{f}$ are zero extensions of v_1 and f, respectively.

Since the solutions to the wave equation have finite speed of propagation, we may replace the domain $\mathbb{R}$ for $\tilde{w}$ by a large enough domain with any boundary condition given that any reflections at the new boundary do not re-enter the region $a \le x \le b$. Let $\tilde{a} < a - c_a T/2$ and $\tilde{b} > b + c_b T/2$. We define the extension operator E such that $[v_0, v_1]^T \to [\tilde{v}_0, \tilde{v}_1]^T$ where $\tilde{v}_0$ and $\tilde{c}$ are the extensions as above and $\tilde{v}_1$, $\tilde{f}$ are zero extensions of v_1 and f, respectively. We now consider the (finite interval) extended problem with homogeneous Neumann conditions

$$\begin{aligned}
&\tilde{w}_{tt} = \frac{\partial}{\partial x}\left[\tilde{c}^2(x)\frac{\partial}{\partial x}\tilde{w}\right] - \mathrm{Re}\{\tilde{f}(x)e^{-i\omega t}\}, \quad \tilde{a} \le x \le \tilde{b}, \quad 0 \le t \le T,\\
&\tilde{w}(0,x) = \tilde{v}_0(x), \quad \tilde{w}_t(0,x) = \tilde{v}_1(x),\\
&\tilde{w}_x(t,\tilde{a}) = 0, \quad \tilde{w}_x(t,\tilde{b}) = 0.
\end{aligned}$$

Defining the projection operator P as the restriction of $\tilde{w}$ to $a \le x \le b$ then it follows that $P\tilde{w} = w$ where w is the original wave solution to the interior impedance problem.

.6 Verification of Discrete Solution

Here we verify that (3.27) is a solution to the difference equation (3.26). Direct substitution yields

$$\begin{aligned}
\hat{w}_j^{n+1} - 2\hat{w}_j^n + \hat{w}_j^{n-1} + 2\left[\sum_{k=1}^{m} \frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!}\right]\hat{w}_j^n = \\
(\hat{v}_j - \hat{v}_j^\infty)\cos(\tilde{\lambda}_j t_n)\left(\cos(\tilde{\lambda}_j\Delta t) - 2 + 2\sum_{k=1}^{m}\frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!} + \cos(\tilde{\lambda}_j\Delta t)\right) \\
+ \hat{v}_j^\infty\cos(\omega t_n)\left(\cos(\omega\Delta t) - 2 + 2\sum_{k=1}^{m}\frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!} + \cos(\omega\Delta t)\right) \\
= (\hat{v}_j - \hat{v}_j^\infty)\cos(\tilde{\lambda}_j t_n)\left(-4\sin^2(\tilde{\lambda}_j\Delta t/2) + 2\sum_{k=1}^{m}\frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!}\right) \\
+ \hat{v}_j^\infty\cos(\omega t_n)\left(-4\sin^2(\omega\Delta t/2) + 2\sum_{k=1}^{m}\frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!}\right) \\
= \hat{v}_j^\infty\cos(\omega t_n)\left(-4\sin^2(\omega\Delta t/2) + 2\sum_{k=1}^{m}\frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!}\right).
\end{aligned}$$

Finally, if

$$\sin^2(\omega\Delta t/2) = \sum_{k=1}^{m}\frac{(-1)^{k+1}\left(\Delta t\tilde{\omega}\right)^{2k}}{2(2k)!},$$

then

$$\begin{aligned}
\hat{w}_j^{n+1} - 2\hat{w}_j^n + \hat{w}_j^{n-1} + 2\left[\sum_{k=1}^{m} \frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!}\right]\hat{w}_j^n \\
= \hat{v}_j^\infty\cos(\omega t_n)\left(-4\sin^2(\omega\Delta t/2) + 2\sum_{k=1}^{m}\frac{(-1)^{k+1}\Delta t^{2k}\lambda_j^{2k}}{(2k)!}\right) \\
= \hat{v}_j^\infty\cos(\omega t_n)\left(2\sum_{k=1}^{m}\frac{(-1)^{k}\Delta t^{2k}\left(\tilde{\omega}^{2k} - \lambda_j^{2k}\right)}{(2k)!}\right) \\
= \hat{v}_j^\infty\cos(\omega t_n)\left(2\sum_{k=1}^{m}\frac{(-1)^{k}\Delta t^{2k}}{(2k)!}(\tilde{\omega}^2 - \lambda_j^2)\sum_{\ell=0}^{k-1}\tilde{\omega}^{2(k-\ell-1)}\lambda^{2\ell}\right) \\
= \hat{f}_j\cos(\omega t_n)\left(2\sum_{k=1}^{m}\frac{(-1)^{k}\Delta t^{2k}}{(2k)!}\sum_{\ell=0}^{k-1}\tilde{\omega}^{2(k-\ell-1)}\lambda^{2\ell}\right),
\end{aligned}$$

since

$$\tilde{\omega}^{2k} - \lambda^{2k} = \left(\tilde{\omega}^2 - \lambda^2\right)\sum_{\ell=0}^{k-1}\tilde{\omega}^{2(k-\ell-1)}\lambda^{2\ell}, \quad k = 1, 2, \ldots,$$

as desired. Moreover, the initial conditions are satisfied as

$$\begin{aligned}
\hat{w}_j^0 &= (\hat{v}_j - \hat{v}_j^\infty) + \hat{v}_j^\infty = \hat{v}_j, \\
\hat{w}_j^{-1} &= (\hat{v}_j - \hat{v}_j^\infty)\cos(\tilde{\lambda}_j \Delta t) + \hat{v}_j^\infty \cos(\omega \Delta t) \\
&= (\hat{v}_j - \hat{v}_j^\infty)\left(1 - \sum_{k=1}^{m} \frac{(-1)^{k+1}\left(\Delta t \lambda_j\right)^{2k}}{2k!}\right) + \hat{v}_j^\infty \left(1 - \sum_{k=1}^{m} \frac{(-1)^{k+1}\left(\Delta t \tilde{\omega}\right)^{2k}}{2k!}\right) \\
&= \hat{v}_j \left(1 - \sum_{k=1}^{m} \frac{(-1)^{k+1}\left(\Delta t \lambda_j\right)^{2k}}{2k!}\right) - \hat{v}_j^\infty \left(\sum_{k=1}^{m} \frac{(-1)^{k+1}\Delta t^{2k}\left(\tilde{\omega}^{2k} - \lambda^{2k}\right)}{2k!}\right) \\
&= \hat{v}_j \left(1 + \sum_{k=1}^{m} \frac{(-1)^{k}\left(\Delta t \lambda_j\right)^{2k}}{2k!}\right) + \hat{f}_j \left(\sum_{k=1}^{m} \frac{(-1)^{k}\Delta t^{2k}}{2k!} \sum_{\ell=0}^{k-1} \tilde{\omega}^{2(k-\ell-1)} \lambda^{2\ell}\right).
\end{aligned}$$

This shows that (3.27) solves (3.26).

.7 Well-definedness of modified frequencies

For an order $2m$ ME scheme we have the relation

$$\sin^2(\omega \Delta t/2) = \sum_{j=1}^{m} \frac{(-1)^{j+1}\left(\Delta t \tilde{\omega}\right)^{2j}}{2(2j)!} = \sin^2(\tilde{\omega}\Delta t/2) + \mathcal{O}(\Delta t^{2m+2}), \tag{7}$$

for which we note that, under the assumptions of Theorem 3.3.1, $\Delta t, \omega > 0$ so that $\Delta t \tilde{\omega} = 0$ does not satisfy the above relation. We seek to show that if $\Delta t \omega \leq 1$ then we have a well-defined (and unique) real-valued $\tilde{\omega}$ such that $0 \leq \Delta t \tilde{\omega} \leq 2$. To that end, we now define the polynomial

$$p(x) = \sum_{j=1}^{m} \frac{(-1)^{j+1} x^j}{2(2j)!} - \sin^2(\omega \Delta t/2), \tag{8}$$

and note that $(\Delta t \tilde{\omega})^2$ is a root of $p(x)$. On the interval $[0, 1/2]$ we have that $\sin^2(x)$ is increasing so that

$$0 \leq \sin^2(\omega \Delta t/2) \leq \sin^2(1/2) \leq 0.23 < 1,$$

immediately giving $p(0) < 0$. Moreover,

$$\begin{aligned} p(4) = -\frac{1}{2}\sum_{j=1}^{m}\frac{(-1)^j 2^{2j}}{(2j)!} - \sin^2(\omega\Delta t/2) &= -\frac{1}{2}\sum_{j=1}^{\infty}\frac{(-1)^j 2^{2j}}{(2j)!} + \frac{1}{2}\sum_{j=m+1}^{\infty}\frac{(-1)^j 2^{2j}}{(2j)!} - \sin^2(\omega\Delta t/2) \\ &= -\frac{1}{2}(\cos(2)-1) + \frac{1}{2}\sum_{j=m+1}^{\infty}\frac{(-1)^j 2^{2j}}{(2j)!} - \sin^2(\omega\Delta t/2) \\ &= \sin^2(1) + \frac{1}{2}\sum_{j=m+1}^{\infty}\frac{(-1)^j 2^{2j}}{(2j)!} - \sin^2(\omega\Delta t/2). \end{aligned}$$

We note that

$$\sum_{j=2}^{\infty}\frac{2^{2j}}{(2j)!} = \sum_{j=0}^{\infty}\frac{2^{2j}}{(2j)!} - 3 = \cosh(2) - 3,$$

so that

$$\frac{1}{2}\sum_{j=m+1}^{\infty}\frac{(-1)^j 2^{2j}}{(2j)!} > -\frac{1}{2}\sum_{j=m+1}^{\infty}\frac{2^{2j}}{(2j)!} \geq -\frac{1}{2}\sum_{j=2}^{\infty}\frac{2^{2j}}{(2j)!} = -\frac{1}{2}(\cosh(2)-3).$$

This gives that

$$\begin{aligned} p(4) = \sin^2(1) + \frac{1}{2}\sum_{j=m+1}^{\infty}\frac{(-1)^j 2^{2j}}{(2j)!} - \sin^2(\omega\Delta t/2) &\geq \sin^2(1) - \frac{1}{2}(\cosh(2)-3) - \sin^2(\omega\Delta t/2) \\ &\geq \sin^2(1) - \frac{1}{2}(\cosh(2)-3) - \sin^2(1) > 0. \end{aligned}$$

By the intermediate value theorem, it follows that $p(x)$ has a root in the interval $[0, 4]$. We next need to show that $p'(x) \neq 0$ on this interval to guarantee the root is unique. Taking a derivative,

$$\frac{d}{dx}p(x) = \sum_{j=1}^{m}\frac{(-1)^{j+1} j x^{j-1}}{(2j)!} = \frac{1}{2}\left[1 + \sum_{j=2}^{m}\frac{(-1)^{j+1}x^{j-1}}{(2j-1)!}\right].$$

We then have

$$\sum_{j=2}^{m}\frac{(-1)^{j+1}x^{j-1}}{(2j-1)!} \geq -\sum_{j=2}^{m}\frac{2^{j-1}}{(2j-1)!} = -\frac{1}{\sqrt{2}}\sum_{j=2}^{m}\frac{\sqrt{2}^{2j-1}}{(2j-1)!} \geq -\frac{1}{\sqrt{2}}\left[\sum_{j=1}^{\infty}\frac{\sqrt{2}^{2j-1}}{(2j-1)!} - \sqrt{2}\right] = -\frac{\sinh(\sqrt{2})}{\sqrt{2}} + 1,$$

so that

$$\frac{d}{dx}p(x) = \frac{1}{2}\left[1 + \sum_{j=2}^{m}\frac{(-1)^{j+1}x^{j-1}}{(2j-1)!}\right] \geq \frac{1}{2} - \frac{\sinh(\sqrt{2})}{\sqrt{2}} + 1 > \frac{1}{10} > 0. \tag{9}$$

This gives that there is a unique, real-valued $\tilde{\omega}$ with $\Delta t\tilde{\omega} \leq 2$ that satisfies the relation (7), which we choose as our modified frequency.

We may similarly prove that the relation (3.28) is well-defined by showing that the right hand side of (3.28) is smaller than (or equal) to one in magnitude. Under the assumptions of Theorem 3.3.1, we have $\Delta t\lambda_j \leq 1$ for each j so that we may bound the right hand side of (3.28) via

$$\left|\sum_{k=1}^{m} \frac{(-1)^{k+1}\left(\Delta t\lambda_j\right)^{2k}}{2(2k)!}\right| \leq \sum_{k=1}^{m} \frac{\left(\Delta t\lambda_j\right)^{2k}}{2(2k)!} \leq \frac{1}{2}\sum_{k=1}^{m} \frac{1}{(2k!} = \frac{1}{2}(\cosh(1)-1) \approx 0.2715 < 1. \tag{10}$$

.8 Error in discrete Helmholtz frequency

We let $x = \Delta t\omega$, $\tilde{x} = \Delta t\tilde{\omega}$, and $R_m = p(x^2) - p(\tilde{x}^2)$ where $p(x)$ is defined as in (8). By the mean value theorem we have

$$|R_m| = |p(x^2) - p(\tilde{x}^2)| = |(x^2 - \tilde{x}^2)p'(\xi)| = |x - \tilde{x}||x + \tilde{x}||p'(\xi)|,$$

for some $\xi \in [0, 2]$, so that

$$|x - \tilde{x}| \leq \frac{|R_m|}{|x + \tilde{x}||p'(\xi)|}.$$

A straightforward bound gives

$$|R_m| = |p(x^2) - \sin^2(\omega\Delta t/2) + \sin^2(\omega\Delta t/2) - p(\tilde{x}^2)| = |p(x^2) - \sin^2(\omega\Delta t/2)| = \left|\sum_{j=m+1}^{\infty} \frac{(-1)^{j+2}\left(\Delta t\omega\right)^{2j}}{2(2j)!}\right| \leq \frac{\left(\Delta t\omega\right)^{2m+2}}{2(2m+2)!},$$

which gives

$$|x - \tilde{x}| \leq \frac{|R_m|}{|x + \tilde{x}||p'(\xi)|} \leq \frac{\Delta t^{2m+1}\omega^{2m+2}}{2(2m+2)!(\omega + \tilde{\omega})|p'(\xi)|} \implies |\omega - \tilde{\omega}| \leq \frac{\Delta t^{2m}\omega^{2m+2}}{2(2m+2)!(\omega + \tilde{\omega})|p'(\xi)|}.$$

By (9) we have that $|p'(x)| > 1/10$ in $[0, 4]$, which finally gives

$$|\omega - \tilde{\omega}| \leq \frac{10\Delta t^{2m}\omega^{2m+2}}{2(2m+2)!(\omega + \tilde{\omega})} \leq \frac{5\Delta t^{2m}\omega^{2m+1}}{(2m+2)!(1 + \tilde{\omega}/\omega)} \leq \frac{5\Delta t^{2m}\omega^{2m+1}}{(2m+2)!} \approx \mathcal{O}(\Delta t^{2m}). \tag{11}$$

.9 Verification of Discrete Solution

Here we verify that (4.30) is a solution to the difference equation (4.29). Direct substitution yields

$$\begin{aligned}
\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\hat{u}_j^{n+1} - \alpha\hat{u}_j^n + \left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\hat{u}_j^{n-1} = \\
&(\hat{u}_{0,j}-\hat{v}_j)\cos(\tilde{\lambda}_j t_n)\left(\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\cos(\tilde{\lambda}_j\Delta t) - \alpha + \left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\cos(\tilde{\lambda}_j\Delta t)\right) \\
&+ \hat{v}_j\cos(\omega t_n)\left(\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\cos(\omega\Delta t) - \alpha + \left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\cos(\omega\Delta t)\right) \\
&= (\hat{u}_{0,j}-\hat{v}_j)\cos(\tilde{\lambda}_j t_n)\left(\frac{\alpha}{2} - \alpha + \frac{\alpha}{2}\right) \\
&+ \hat{v}_j\cos(\omega t_n)\cos(\omega\Delta t)\left(\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right) - \left(2+\Delta t^2\omega^2\right) + \left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)\right) \\
&= \hat{v}_j\cos(\omega t_n)\cos(\omega\Delta t)\Delta t^2\left(\lambda_j^2-\omega^2\right) \\
&= -\Delta t^2\hat{f}_j\cos(\omega t_n)\cos(\omega\Delta t),
\end{aligned}$$

as desired. Moreover, we can check that the initial condition is satisfied since

$$\hat{u}_j^0 = (\hat{u}_{0,j}-\hat{v}_j) + \hat{v}_j = \hat{u}_{0,j},$$

and

$$\begin{aligned}
\hat{u}_j^{-1} = (\hat{u}_{0,j}-\hat{v}_j)\cos(\tilde{\lambda}_j\Delta t) + \hat{v}_j\cos(\omega\Delta t) &= \hat{u}_{0,j}\cos(\tilde{\lambda}_j\Delta t) + \hat{v}_j\left(\cos(\omega\Delta t) - \cos(\tilde{\lambda}_j\Delta t)\right) \\
&= \hat{u}_{0,j}\cos(\tilde{\lambda}_j\Delta t) + \hat{v}_j\cos(\omega\Delta t)\left(1 - \left(1+\frac{\omega^2\Delta t^2}{2}\right)\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)^{-1}\right) \\
&= \hat{u}_{0,j}\cos(\tilde{\lambda}_j\Delta t) + \hat{v}_j\cos(\omega\Delta t)\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)^{-1}\frac{\Delta t^2}{2}\left(\lambda_j^2-\omega^2\right) \\
&= \left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)^{-1}\left(\frac{\alpha}{2}\hat{u}_{0,j} - \frac{\Delta t^2}{2}\hat{f}_j\cos(\omega\Delta t)\right).
\end{aligned}$$

This shows that (4.30) is indeed a solution to the implicit scheme (4.25).

.10 Time-step Restriction

To understand how restrictive the requirement (4.26) is, we plot α for various values of $\omega\Delta t$ below in Figure 1. From Figure 1 we see that $|\alpha| < 2$ for $\Delta t < r/\omega$ where $r \approx 1.93$. This choice of

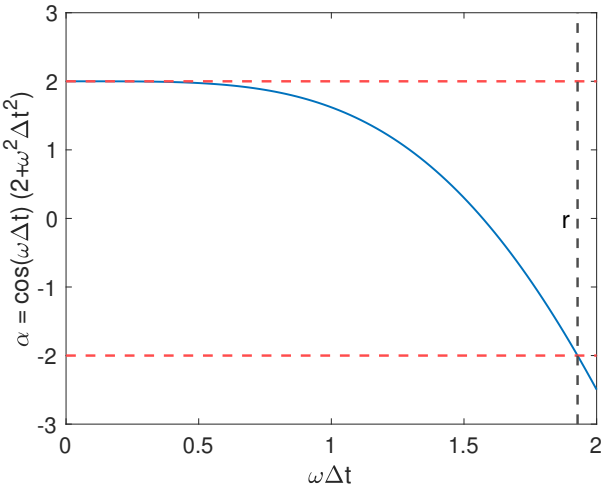


Figure 1: Values of $\alpha = \cos(\omega\Delta t)(2+\omega^2\Delta t^2)$ for values of $\omega\Delta t$ in the interval $[0, 2]$. The red lines indicate the desired bound on α, and the black line indicates the maximum allowable value of $\omega\Delta t$ at $r \approx 1.93$.

the time-step corresponds to a requirement of at least four time-steps per iteration. However, the WaveHoltz kernel is a constant with four time-steps so that at least five time-steps are needed for stability.

.11 Motivation of Conjecture 1

For the stability requirement (4.28), which requires at least five time-steps per iteration, we note that from (4.31) we have that the right hand side is a decreasing function of λ so that

$$\cos(\tilde{\lambda}_j\Delta t) \le \lim_{\lambda_j\to\infty} \cos(\omega\Delta t)\left(1+\frac{\omega^2\Delta t^2}{2}\right)\left(1+\frac{\Delta t^2}{2}\lambda_j^2\right)^{-1} = 0,$$

implying that

$$0 \le \tilde{\lambda}_j \le \frac{\pi}{2\Delta t} = \frac{k}{4}\omega,$$

where $k \ge 5$ is the number of quadrature points used in the trapezoidal rule. We now consider the (continous) rescaled filter transfer function,

$$\bar{\beta}(r) := \beta(r\omega) = \frac{2}{T}\int_0^T \left(\cos(\omega t) - \frac{1}{4}\right)\cos(r\omega t)dt = \frac{1}{\pi}\int_0^{2\pi}\left(\cos(t)-\frac{1}{4}\right)\cos(rt)dt,$$

with discrete analogue

$$\bar{\beta}_h(r) = \frac{\Delta t}{\pi} \sum_{n=0}^{M} \eta_n \cos(r t_n) \left(\cos(t_n) - \frac{1}{4} \right), \qquad \eta_n = \begin{cases} \frac{1}{2}, & n = 0 \text{ or } n = M, \\ 1, & 0 < n < M. \end{cases}$$

Let us now take a look at the rescaled discrete filter function, $\bar{\beta}_h(r)$. It is sufficient to consider only the range $r \in [0, k/4]$, which we plot in Figure 2.

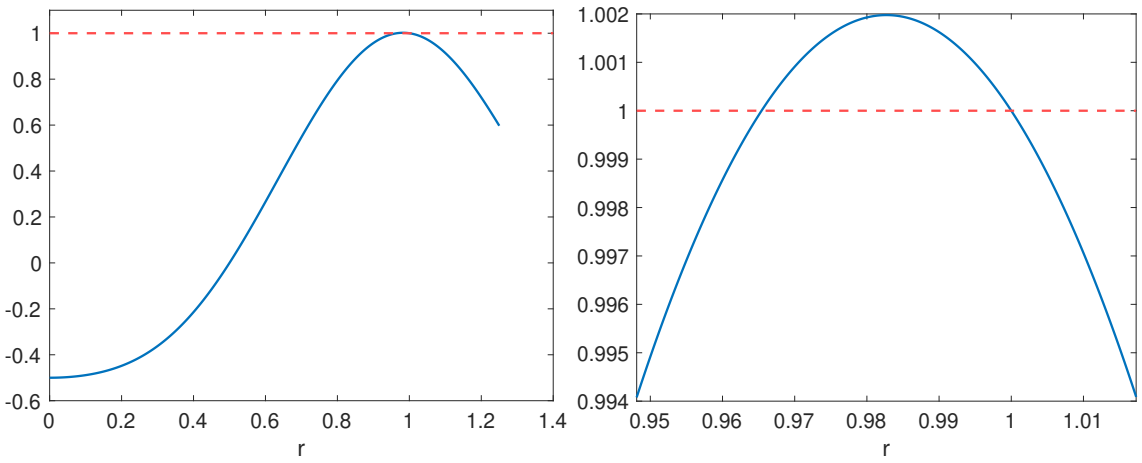


Figure 2: A plot of the discrete filter function using five time-steps $0 \leq r \leq 5/4$. On the left we plot the full range of values of r, and on the right we zoom in close to resonance, i.e. $r = 1$.

From Figure 2 we see that it is possible to integrate and get eigenvalues of the WHI operator to be larger than one for a small range near resonance, $r = 1$. To get a sense of the size of this gap, we perform a simple bisection where we find the leftmost point $r^* < 1$ such that $\beta_h(r^*) = 1$ for increasing number of quadrature points $k = 5, 6, \ldots, 100$. We plot the result below in Figure 3. From Figure 3 we see that, perhaps unsurprisingly, the gap shrinks with increasing number of quadrature points. The curve in red in Figure 3 indicates the bound

$$1 - r^* \leq 0.022 \cdot \Delta t^2,$$

so that we see that this gap shrinks as Δt^2. Moreover, if $|1 - r| \geq 0.022 \cdot \Delta t^2$ then $|\beta_h(r)| < 1$. For $r = \tilde{\lambda}_j / \omega$, we may thus obtain convergence of the iteration if it can be guaranteed the time-step is chosen such that $\tilde{\lambda}_j \notin [\omega(1 - 0.022 \cdot \Delta t^2), \omega]$.

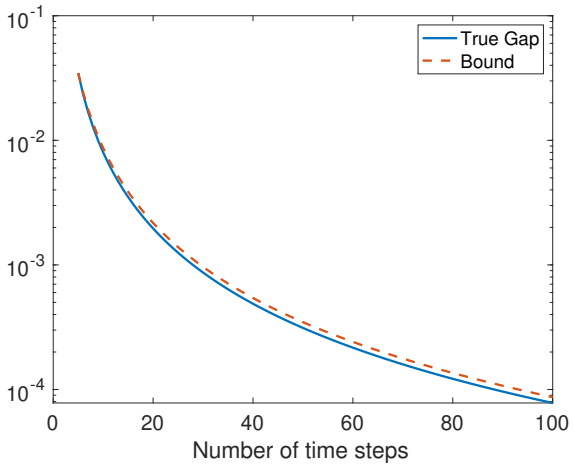


Figure 3: A bound on the gap from resonance that creates problematic modes. The blue curve is the true gap, $1 - r^*$, and the dotted red curve is a proposed bound.

Assuming we have a bound similar to the above, let us now try to get a minimum on the distance $|\tilde{\lambda}_j - \omega|$. By the mean-value theorem we have the bound $|\cos(x) - \cos(y)| \leq |x - y|$, so that

$$\frac{|\cos(\tilde{\lambda}_j \Delta t) - \cos(\omega \Delta t)|}{\Delta t} \leq |\tilde{\lambda}_j - \omega|.$$

Some algebra shows that

$$\begin{aligned}
\cos(\omega\Delta t) - \cos(\tilde{\lambda}_j \Delta t) &= \cos(\omega\Delta t) - \cos(\omega\Delta t)\left(1 + \frac{\omega^2 \Delta t^2}{2}\right)\left(1 + \frac{\Delta t^2}{2}\lambda_j^2\right)^{-1} \\
&= \cos(\omega\Delta t)\frac{\Delta t^2}{2}\left(\lambda_j^2 - \omega^2\right)\left(1 + \frac{\Delta t^2}{2}\lambda_j^2\right)^{-1} \\
&= \cos(\omega\Delta t)\frac{\Delta t^2}{2}\left(\lambda_j - \omega\right)\left(\lambda_j + \omega\right)\left(1 + \frac{\Delta t^2}{2}\lambda_j^2\right)^{-1} \\
&= \cos(\omega\Delta t)\frac{\Delta t^2}{2}\omega\delta_h\left(\lambda_j + \omega\right)\left(1 + \frac{\Delta t^2}{2}\lambda_j^2\right)^{-1},
\end{aligned}$$

where $\delta_h = \min_j |\lambda_j - \omega|/\omega > 0$ is the minimum gap to resonance. We then have

$$\begin{aligned}
|\tilde{\lambda}_j - \omega| = \cos(\omega\Delta t)\frac{\Delta t}{2}\omega\delta_h\left(\lambda_j + \omega\right)\left(1 + \frac{\Delta t^2}{2}\lambda_j^2\right)^{-1} &\geq \cos(\omega\Delta t)\frac{\Delta t}{2}\omega\delta_h\left(\lambda_1 + \omega\right)\left(1 + \frac{\Delta t^2}{2}\lambda_N^2\right)^{-1} \\
&\geq \cos(2\pi/5)\frac{\Delta t}{2}\omega\delta_h\left(\lambda_1 + \omega\right)\left(1 + \frac{\Delta t^2}{2}\lambda_N^2\right)^{-1} \\
&\geq \cos(2\pi/5)\frac{\Delta t}{2}\omega\delta_h\left(\lambda_1 + \omega\right)\left(1 + 2(\pi\lambda_N/5)^2\right)^{-1},
\end{aligned}$$

so that we need to choose Δt such that

$$\cos(\omega\Delta t)\frac{\Delta t}{2}\omega\delta_h\left(\lambda_1+\omega\right)\left(1+\frac{\Delta t^2}{2}\lambda_N^2\right)^{-1}\geq 0.022\cdot\Delta t^2,$$

which gives the condition

$$\Delta t\leq\frac{\cos(2\pi/5)\omega^2\delta_h}{0.044\cdot(1+2(\pi\lambda_N/5)^2)}\leq\frac{\cos(2\pi/5)\omega\delta_h\left(\lambda_1+\omega\right)}{0.044\cdot(1+2(\pi\lambda_N/5)^2)}\approx 7.02\cdot\frac{\delta_h\omega\left(\lambda_1+\omega\right)}{1+2(\pi\lambda_N/5)^2}.$$

3097

3098 .12 Composite quantum systems and essential states

To simplify the notation we assume a bipartite quantum system ($Q=2$); the case $Q=1$ is trivial and $Q>2$ follows by straightforward generalizations. Let the number of energy levels in the subsystems be n_1 and n_2, respectively, for a total of $N=n_1\cdot n_2$ states in the coupled system. We use the canonical unit vectors $\boldsymbol{e}_j^{(n_q)}\in\mathbb{R}^{n_q}$, for $j=0,\ldots,n_q-1$, as a basis for subsystem q, where the superscript indicates its size. These basis vectors can be used to describe the total state of the coupled system,

$$\boldsymbol{\psi}=\sum_{j_2=0}^{n_2-1}\sum_{j_1=0}^{n_1-1}\psi_{j_2,j_1}\left(\boldsymbol{e}_{j_2}^{(n_2)}\otimes\boldsymbol{e}_{j_1}^{(n_1)}\right)=\sum_{k=0}^{N-1}\vec{\psi}_k\boldsymbol{e}_k^{(N)}. \tag{12}$$

3099 Here, $\vec{\psi}\in\mathbb{C}^N$ denotes the one-dimensional representation of the two-dimensional state vector $\boldsymbol{\psi}$,
3100 using a natural ordering of the elements, i.e., $\vec{\psi}_k=\psi_{j_2,j_1}$ for $k=j_1+n_1j_2=:k_{ind}(j_2,j_1)$. The
3101 mapping $k=k_{ind}(j_2,j_1)$ is invertible for $k\in[0,N-1]$.

3102 We classify the energy levels in the total state vector as either essential or guarded levels. The
3103 unitary gate transformation is only specified for the essential levels. The guard levels are retained
3104 to justify the truncation of the modal expansion of Schrödinger's equation, and to avoid leakage of
3105 probability to even higher energy levels.

Let the number of essential energy levels in the subsystems be m_1 and m_2, respectively, where $0<m_q\leq n_q$. Similar to the total state vector, we use the canonical unit vectors as a basis for the essential subspace of each subsystem. The total number of essential levels equals $E=m_1\cdot m_2$. Let

the essential energy levels in the total state vector be represented by the essential state vector $\boldsymbol{\phi}$. Similar to the full state vector, we flatten its two-dimensional indexing using a natural ordering,

$$\boldsymbol{\phi} = \sum_{i_2=0}^{m_2-1} \sum_{i_1=0}^{m_1-1} \phi_{i_2,i_1} \left(\boldsymbol{e}_{i_2}^{(m_2)} \otimes \boldsymbol{e}_{i_1}^{(m_1)} \right) = \sum_{\ell=0}^{E-1} \vec{\phi}_\ell \boldsymbol{e}_\ell^{(E)} \in \mathbb{C}^E, \tag{13}$$

where $\ell = i_1 + m_1 i_2 =: \ell_{ind}(i_2, i_1)$. The elements in the essential state vector are defined from the total state vector by $\phi_{i_2,i_1} = \psi_{i_2,i_1}$, for $i_1 \in [0, m_1 - 1]$ and $i_2 \in [0, m_2 - 1]$.

The initial condition for the solution operator matrix $U(t)$ in Schrödinger's equation (5.3) needs to span a basis for the E-dimensional essential state space. Here we use the canonical basis consisting of the unit vectors $\boldsymbol{e}_\ell^{(E)}$. Let the columns of the initial condition matrix be $U_0 = [\boldsymbol{g}_0, \boldsymbol{g}_1, \ldots, \boldsymbol{g}_{E-1}] \in \mathbb{R}^{N \times E}$. Because the total probabilities in each column vector $\boldsymbol{g}_k$ must sum to one, the basis vectors in the total state space become

$$\boldsymbol{g}_\ell = U_0 \boldsymbol{e}_\ell^{(E)}, \quad g_{k,\ell} = \begin{cases} 1, & k = k_{ind}(i_2(\ell), i_1(\ell)), \\ 0, & \text{otherwise}, \end{cases} \quad \text{for } \ell = 0, 1, \ldots, E-1. \tag{14}$$

Here, $i_2(\ell) = \lfloor \ell / m_1 \rfloor$ and $i_1(\ell) = \ell - m_1 \cdot i_2(\ell)$.

The target gate matrix $V_E \in \mathbb{C}^{E \times E}$ defines the unitary transformation between the essential levels in the initial and final states, $\boldsymbol{\phi}_T = V_E \boldsymbol{\phi}_0$, for all $\boldsymbol{\phi}_0 \in \mathbb{C}^E$. Because V_E is unitary, each of its columns has norm one. To preserve total probabilities, we define the target gate transformation according to

$$V_{tg} = U_0 V_E \in \mathbb{C}^{N \times E}. \tag{15}$$

This implies that each column of V_{tg} also has norm one.

Example: A bipartite quantum system As a small example, consider a composite system considering of two subsystems, each with three energy levels, $n_1 = n_2 = 3$. In this case, the

dimension of the full state vector is $N = 9$. It can be written as:

$$\boldsymbol{\psi} = \sum_{j_2=0}^{n_2-1}\sum_{j_1=0}^{n_1-1} \psi_{j_2,j_1}\left(\boldsymbol{e}_{j_2}^{(3)} \otimes \boldsymbol{e}_{j_1}^{(3)}\right) = \begin{bmatrix} \psi_{0,0} \\ \psi_{0,1} \\ \psi_{0,2} \\ \psi_{1,0} \\ \psi_{1,1} \\ \psi_{1,2} \\ \psi_{2,0} \\ \psi_{2,1} \\ \psi_{2,2} \end{bmatrix}, \quad \boldsymbol{\psi} = \sum_{k=0}^{8} \vec{\psi}_k \boldsymbol{e}_k^{(9)} = \begin{bmatrix} \vec{\psi}_0 \\ \vec{\psi}_1 \\ \vec{\psi}_2 \\ \vec{\psi}_3 \\ \vec{\psi}_4 \\ \vec{\psi}_5 \\ \vec{\psi}_6 \\ \vec{\psi}_7 \\ \vec{\psi}_8 \end{bmatrix}. \tag{16}$$

If both systems have two essential levels, i.e. $m_1 = m_2 = 2$, there are $E = 4$ essential levels in the composite system. In this case the total and essential state vectors are related by

$$\vec{\phi} = \begin{bmatrix} \psi_{0,0} \\ \psi_{0,1} \\ \psi_{1,0} \\ \psi_{1,1} \end{bmatrix} = U_0^\dagger \vec{\psi}, \quad U_0^\dagger = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \in \mathbb{R}^{4\times 9}. \tag{17}$$

3110

3111 .13 The Hamiltonian in a rotating frame of reference

3112 The time-dependent and unitary change of variables $\widetilde{\boldsymbol{\psi}}(t) = R(t)\boldsymbol{\psi}(t)$ where $R^\dagger R = I$, results
3113 in the transformed Schrödinger equation

$$\frac{d\widetilde{\boldsymbol{\psi}}}{dt} = -i\widetilde{H}(t)\widetilde{\boldsymbol{\psi}}, \quad \widetilde{H}(t) = R(t)H(t)R^\dagger(t) + i\dot{R}(t)R^\dagger(t). \tag{18}$$

The rotating frame of reference is introduced by taking the unitary transformation to be the matrix (5.14). Because both $R(t)$ and the system Hamiltonian (5.10) are diagonal, $RH_sR^\dagger = H_s$. The time derivative of the transformation can be written

$$\dot{R}(t) = \left(\bigoplus_{q=Q}^{1} i\omega_{r,q}A_q^\dagger A_q\right)\left(\bigotimes_{q=Q}^{1} \exp\left(i\omega_{r,q}tA_q^\dagger A_q\right)\right), \tag{19}$$

where $\oplus$ denotes the Kronecker sum, $C \oplus D = C \otimes I_D + I_C \otimes D$. Therefore,

$$\dot{R}(t)R^\dagger(t) = \bigoplus_{q=Q}^{1} i\omega_{r,q} A_q^\dagger A_q = \sum_{q=1}^{Q} i\omega_{r,q} a_q^\dagger a_q. \tag{20}$$

As a result, the first term in the Hamiltonian (5.10) is modified by the term $i\dot{R}(t)R^\dagger(t)$. After noting that $Ra_qR^\dagger = e^{-i\omega_{r,q}t}a_q$, the transformed Hamiltonian can be written as

$$H_s^{rw} = \sum_{q=1}^{Q} \left(\Delta_q a_q^\dagger a_q - \frac{\xi_q}{2} a_q^\dagger a_q^\dagger a_q a_q - \sum_{p>q} \xi_{qp} a_q^\dagger a_q a_p^\dagger a_p \right), \tag{21}$$

$$\widetilde{H}_c(t) = \sum_{q=1}^{Q} f_q(t;\boldsymbol{\alpha}) \left(e^{-i\omega_{r,q}t} a_q + e^{i\omega_{r,q}t} a_q^\dagger \right), \tag{22}$$

where $\Delta_q = \omega_q - \omega_{r,q}$ is the detuning frequency. The above system Hamiltonian corresponds to (5.15).

To slow down the time scales in the control Hamiltonian, we want to absorb the highly oscillatory factors $\exp(\pm i\omega_{r,q}t)$ into $f_q(t)$. Because the control function $f_q(t)$ is real-valued, this can only be done in an approximate fashion. We make the ansatz,

$$f_q(t) := 2\,\mathrm{Re}\left(d_q(t)e^{i\omega_{r,q}t}\right) = d_q(t)e^{i\omega_{r,q}t} + \bar{d}_q(t)e^{-i\omega_{r,q}t}, \tag{23}$$

where $\bar{d}_q$ denotes the complex conjugate of d_q. By substituting this expression into the transformed control Hamiltonian (22), we get

$$\widetilde{H}_c(t) = \sum_{q=1}^{Q} \left(d_q(t)a_q + \bar{d}_q(t)a_q^\dagger + \bar{d}_q(t)e^{-2i\omega_{r,q}t}a_q + d_q(t)e^{2i\omega_{r,q}t}a_q^\dagger \right).$$

The rotating frame approximation follows by ignoring terms that oscillate with frequency, $\pm 2i\omega_{r,q}$, resulting in the approximate control Hamiltonian (5.16).

.14 Conditions for resonance

Consider the scalar function $y(t) := \psi_{\boldsymbol{j}}^{(1)}(t)$. It satisfies an ordinary differential equation of the form

$$\frac{dy(t)}{dt} + \kappa_{\boldsymbol{j}} y(t) = \sum_{\ell} c_\ell e^{i\nu_\ell t}, \quad \nu_k \in \mathbb{R}. \tag{24}$$

We are interested in cases when $y(t)$ grows in time, corresponding to resonance. Conditions for resonance are provided in the following lemma.

Lemma .14.1. *Let $\kappa \in \mathbb{R}$ and $\nu \in \mathbb{R}$ be constants. The solution of the scalar ordinary differential equation*

$$\frac{dy(t)}{dt} + i\kappa y(t) = ce^{i\nu t}, \quad y(0) = y_0, \tag{25}$$

is given by

$$y(t) = \begin{cases} y_0 e^{-i\kappa t} + cte^{-i\kappa t}, & \nu + \kappa = 0, \\ y_0 e^{-i\kappa t} - \dfrac{ic}{\nu + \kappa}\left(e^{i\nu t} - e^{-i\kappa t}\right), & \textit{otherwise.} \end{cases} \tag{26}$$

Corresponding to resonance, the function $y(t)$ grows linearly in time when $\nu + \kappa = 0$ and $c \neq 0$.

Proof. Follows by direct evaluation. □

We proceed by analyzing the right hand side of (5.20). It can be shown that the forcing function $\boldsymbol{f}^{(k)}(t)$ is of the form The forcing function $\boldsymbol{f}^{(k)}(t)$ contains the terms

$$\{a_k \boldsymbol{\psi}^{(0)}\}_{\boldsymbol{j}} = \begin{cases} g_{\boldsymbol{j}+\boldsymbol{e}_k}\sqrt{j_k+1}\, e^{-(i\kappa_{\boldsymbol{j}+\boldsymbol{e}_k}t)}, & j_k \in [0, n_1 - 2], \\ 0, & j_k = n_k - 1, \end{cases} \tag{27}$$

and

$$\{a_k^\dagger \boldsymbol{\psi}^{(0)}\}_{\boldsymbol{j}} = \begin{cases} 0, & j_k = 0, \\ g_{\boldsymbol{j}-\boldsymbol{e}_k}\sqrt{j_k}\, e^{-(i\kappa_{\boldsymbol{j}-\boldsymbol{e}_k}t)}, & j_k \in [1, n_k - 1]. \end{cases} \tag{28}$$

Therefore,

$$\boldsymbol{f}_{\boldsymbol{j}}^{(k)}(t) = \begin{cases} -ig_{\boldsymbol{j}+\boldsymbol{e}_k}\sqrt{j_k+1}\, e^{i(\Omega_k - \kappa_{\boldsymbol{j}+\boldsymbol{e}_k})t}, & j_k = 0, \\ \Theta_k(t), & j_k \in [1, n_k - 2], \\ -ig_{\boldsymbol{j}-\boldsymbol{e}_k}\sqrt{j_k}\, e^{-i(\Omega_k + \kappa_{\boldsymbol{j}-\boldsymbol{e}_k})t}, & j_k = n_k - 1, \end{cases} \tag{29}$$

where $\Theta_k(t) = -ig_{\boldsymbol{j}+\boldsymbol{e}_k}\sqrt{j_k+1}\, e^{i(\Omega_k - \kappa_{\boldsymbol{j}+\boldsymbol{e}_k})t} - ig_{\boldsymbol{j}-\boldsymbol{e}_k}\sqrt{j_k}\, e^{-i(\Omega_k + \kappa_{\boldsymbol{j}-\boldsymbol{e}_k})t}$.

The right hand side satisfies $\boldsymbol{f}(t) = \boldsymbol{f}^{(1)}(t) + \boldsymbol{f}^{(2)}(t)$. The first set of frequencies and coefficients on the right hand side of (24) satisfy

$$\nu_1 = \Omega_k - \kappa_{\boldsymbol{j}+\boldsymbol{e}_k}, \quad c_1 = -ig_{\boldsymbol{j}+\boldsymbol{e}_k}\sqrt{j_k+1},$$

for $k = \{1, 2\}$ and $j_k \in [0, n_k - 2]$. The second set of frequencies and coefficients are

$$\nu_2 = -(\Omega_k + \kappa_{\boldsymbol{j}-\boldsymbol{e}_k}), \quad c_2 = -ig_{\boldsymbol{j}-\boldsymbol{e}_k}\sqrt{j_k}.$$

for $k = \{1, 2\}$ and $j_k \in [1, n_k - 1]$. From Lemma .14.1, component $\psi^{(1)}_{\boldsymbol{j}}(t)$ is in resonance if $(\kappa_{\boldsymbol{j}} + \nu_1 = 0, c_1 \neq 0)$ or $(\kappa_{\boldsymbol{j}} + \nu_2 = 0, c_2 \neq 0)$. These conditions are equivalent to (5.22) and (5.23), which

.15 Derivation of the discrete adjoint scheme

We seek to determine a scheme for evolving the Lagrange multiplier (adjoint) variables to satisfy the first order optimality conditions (5.57). In the following, let $\delta_{r,s}$ denote the usual Kronecker delta function.

The terms T^3_j to T^6_j in (5.53) enforce the relations between the stage variables (5.45)-(5.48) according to

$$T^3_j = \sum_{n=0}^{M-1} \left\langle \boldsymbol{U}^{n,1}_j - \boldsymbol{u}^n_j, \boldsymbol{M}^{n,1}_j \right\rangle_2, \tag{30}$$

$$T^4_j = \sum_{n=0}^{M-1} \left\langle \boldsymbol{U}^{n,2}_j - \boldsymbol{u}^n_j - \frac{h}{2}\left(S_n\boldsymbol{U}^{n,1}_j + S_{n+1}\boldsymbol{U}^{n,2}_j - K_n\boldsymbol{V}^{n,1}_j - K_{n+1}\boldsymbol{V}^{n,2}_j\right), \boldsymbol{M}^{n,2}_j \right\rangle_2, \tag{31}$$

$$T^5_j = \sum_{n=0}^{M-1} \left\langle \boldsymbol{V}^{n,1}_j - \boldsymbol{v}^n_j - \frac{h}{2}\left(K_{n+1/2}\boldsymbol{U}^{n,1}_j + S_{n+1/2}\boldsymbol{V}^{n,1}_j\right), \boldsymbol{N}^{n,1}_j \right\rangle_2, \tag{32}$$

$$T^6_j = \sum_{n=0}^{M-1} \left\langle \boldsymbol{V}^{n,2}_j - \boldsymbol{v}^n_j - \frac{h}{2}\left(K_{n+1/2}\boldsymbol{U}^{n,1}_j + S_{n+1/2}\boldsymbol{V}^{n,1}_j\right), \boldsymbol{N}^{n,2}_j \right\rangle_2. \tag{33}$$

Taking the derivative of (5.53) with respect to $\boldsymbol{u}^r_j$

$$0 = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{u}^r_j} = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{u}^r_j} - \left[(\boldsymbol{\mu}^n_j - \boldsymbol{\mu}^{n+1}_j)\delta_{r,n} + \boldsymbol{\mu}^M_j\delta_{r,M} - (\boldsymbol{M}^{n,1}_j + \boldsymbol{M}^{n,2}_j)\delta_{r,n}\right],$$

which gives the conditions

$$\boldsymbol{\mu}_j^M = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{u}_j^M}, \quad \boldsymbol{\mu}_j^n - \boldsymbol{\mu}_j^{n+1} = \boldsymbol{M}_j^{n,1} + \boldsymbol{M}_j^{n,2}, \quad n = 0, 1, \ldots, M-1.$$

Similarly, differentiating (5.53) with respect to $\boldsymbol{v}_j^r$ gives

$$0 = \frac{\partial \mathcal{L}^h}{\partial \boldsymbol{v}_j^r} = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{v}_j^r} - \left[(\boldsymbol{\nu}_j^n - \boldsymbol{\nu}_j^{n+1})\delta_{r,n} + \boldsymbol{\nu}_j^M \delta_{r,M} - (\boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2})\delta_{r,n} \right],$$

which leads to the conditions

$$\boldsymbol{\nu}_j^n - \boldsymbol{\nu}_j^{n+1} = \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2}, \quad \boldsymbol{\nu}_j^M = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{v}_j^M}.$$

Next we take the derivative of (5.53) with respect to $\boldsymbol{U}_j^{n,1}$,

$$\begin{aligned}
\frac{\partial \mathcal{L}^h}{\partial \boldsymbol{U}_j^{n,1}} &= \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}} - \sum_{i=1}^{6} \frac{\partial T_j^i}{\partial \boldsymbol{U}_j^{n,1}} = 0, \\
\frac{\partial T_j^1}{\partial \boldsymbol{U}_j^{n,1}} &= -\frac{h}{2} S_n^T \boldsymbol{\mu}_j^{n+1}, \\
\frac{\partial T_j^2}{\partial \boldsymbol{U}_j^{n,1}} &= -\frac{h}{2} K_{n+1/2}^T \boldsymbol{\nu}_j^{n+1}, \\
\frac{\partial T_j^3}{\partial \boldsymbol{U}_j^{n,1}} &= \boldsymbol{M}_j^{n,1}, \\
\frac{\partial T_j^4}{\partial \boldsymbol{U}_j^{n,1}} &= -\frac{h}{2} S_n^T \boldsymbol{M}_j^{n,2}, \\
\frac{\partial T_j^5}{\partial \boldsymbol{U}_j^{n,1}} &= -\frac{h}{2} K_{n+1/2}^T \boldsymbol{N}_j^{n,1}, \\
\frac{\partial T_j^6}{\partial \boldsymbol{U}_j^{n,1}} &= -\frac{h}{2} K_{n+1/2}^T \boldsymbol{N}_j^{n,2},
\end{aligned}$$

which, using the fact that $S_n^T = -S_n$ and $K_n^T = K_n$, we may write as

$$\boldsymbol{M}_j^{n,1} + \frac{h}{2} S_n \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) - \frac{h}{2} K_{n+1/2} \left(\boldsymbol{\nu}_j^{n+1} + \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2} \right) = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}}.$$

Repeating this procedure for the derivative with respect to $\boldsymbol{U}_j^{n,2}$ gives

$$\begin{aligned}
\frac{\partial \mathcal{L}^h}{\partial \boldsymbol{U}_j^{n,2}} &= \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}} - \sum_{i=1}^{6} \frac{\partial T_j^i}{\partial \boldsymbol{U}_j^{n,2}} = 0, \\
\frac{\partial T_j^1}{\partial \boldsymbol{U}_j^{n,2}} &= -\frac{h}{2} S_{n+1}^T \boldsymbol{\mu}_j^{n+1}, \\
\frac{\partial T_j^2}{\partial \boldsymbol{U}_j^{n,2}} &= -\frac{h}{2} K_{n+1/2}^T \boldsymbol{\nu}_j^{n+1}, \\
\frac{\partial T_j^4}{\partial \boldsymbol{U}_j^{n,2}} &= \boldsymbol{M}_j^{n,2} - \frac{h}{2} S_{n+1}^T \boldsymbol{M}_j^{n,2}, \\
\frac{\partial T_j^3}{\partial \boldsymbol{U}_j^{n,2}} &= \frac{\partial T_j^5}{\partial \boldsymbol{U}_j^{n,2}} = \frac{\partial T_j^6}{\partial \boldsymbol{U}_j^{n,2}} = 0,
\end{aligned}$$

which we may write compactly as

$$\boldsymbol{M}_j^{n,2} + \frac{h}{2} S_{n+1} \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) - \frac{h}{2} K_{n+1/2} \boldsymbol{\nu}_j^{n+1} = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}}.$$

Taking the derivative of (5.53) with respect to $\boldsymbol{V}_j^{n,1}$ gives the set of equations

$$\begin{aligned}
\frac{\partial \mathcal{L}^h}{\partial \boldsymbol{V}_j^{n,1}} &= \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}} - \sum_{i=1}^{6} \frac{\partial T_j^i}{\partial \boldsymbol{V}_j^{n,1}} = 0, \\
\frac{\partial T_j^1}{\partial \boldsymbol{V}_j^{n,1}} &= \frac{h}{2} K_n^T \boldsymbol{\mu}_j^{n+1}, \\
\frac{\partial T_j^2}{\partial \boldsymbol{V}_j^{n,1}} &= -\frac{h}{2} S_{n+1/2}^T \boldsymbol{\nu}_j^{n+1}, \\
\frac{\partial T_j^3}{\partial \boldsymbol{V}_j^{n,1}} &= 0, \\
\frac{\partial T_j^4}{\partial \boldsymbol{V}_j^{n,1}} &= \frac{h}{2} K_n^T \boldsymbol{M}_j^{n,2}, \\
\frac{\partial T_j^5}{\partial \boldsymbol{V}_j^{n,1}} &= \boldsymbol{N}_j^{n,1} - \frac{h}{2} S_{n+1/2}^T \boldsymbol{N}_j^{n,1}, \\
\frac{\partial T_j^6}{\partial \boldsymbol{V}_j^{n,1}} &= -\frac{h}{2} S_{n+1/2}^T \boldsymbol{N}_j^{n,2},
\end{aligned}$$

which gives the condition

$$\boldsymbol{N}_j^{n,1} + \frac{h}{2} S_{n+1/2} \left(\boldsymbol{\nu}_j^{n+1} + \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2} \right) + \frac{h}{2} K_n \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}}.$$

Similarly, taking the derivative with respect to $\boldsymbol{V}_j^{n,2}$ gives

$$\begin{aligned}
\frac{\partial \mathcal{L}^h}{\partial \boldsymbol{V}_j^{n,2}} &= \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}} - \sum_{i=1}^{6} \frac{\partial T_j^i}{\partial \boldsymbol{V}_j^{n,2}} = 0, \\
\frac{\partial T_j^1}{\partial \boldsymbol{V}_j^{n,2}} &= \frac{h}{2} K_{n+1}^T \boldsymbol{\mu}_j^{n+1}, \\
\frac{\partial T_j^2}{\partial \boldsymbol{V}_j^{n,2}} &= -\frac{h}{2} S_{n+1/2}^T \boldsymbol{\nu}_j^{n+1}, \\
\frac{\partial T_j^4}{\partial \boldsymbol{V}_j^{n,2}} &= \frac{h}{2} K_{n+1}^T \boldsymbol{M}_j^{n,2}, \\
\frac{\partial T_j^6}{\partial \boldsymbol{V}_j^{n,2}} &= \boldsymbol{N}_j^{n,2}, \\
\frac{\partial T_j^3}{\partial \boldsymbol{V}_j^{n,2}} &= \frac{\partial T_j^5}{\partial \boldsymbol{V}_j^{n,2}} = 0,
\end{aligned}$$

giving

$$\boldsymbol{N}_j^{n,2} + \frac{h}{2} S_{n+1/2} \boldsymbol{\nu}_j^{n+1} + \frac{h}{2} K_{n+1} \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}}.$$

In summary, the first order optimality conditions (5.57) are satisfied if the following equations hold:

$$\boldsymbol{\mu}_j^n - \boldsymbol{\mu}_j^{n+1} = \boldsymbol{M}_j^{n,1} + \boldsymbol{M}_j^{n,2}, \quad \boldsymbol{\mu}_j^M = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{u}_j^M}, \tag{34}$$

$$\boldsymbol{\nu}_j^n - \boldsymbol{\nu}_j^{n+1} = \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2}, \quad \boldsymbol{\nu}_j^M = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{v}_j^M}, \tag{35}$$

$$\boldsymbol{M}_j^{n,1} + \frac{h}{2} S_n \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) - \frac{h}{2} K_{n+1/2} \left(\boldsymbol{\nu}_j^{n+1} + \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2} \right) = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}}, \tag{36}$$

$$\boldsymbol{M}_j^{n,2} + \frac{h}{2} S_{n+1} \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) - \frac{h}{2} K_{n+1/2} \boldsymbol{\nu}_j^{n+1} = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}}, \tag{37}$$

$$\boldsymbol{N}_j^{n,1} + \frac{h}{2} S_{n+1/2} \left(\boldsymbol{\nu}_j^{n+1} + \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2} \right) + \frac{h}{2} K_n \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}}, \tag{38}$$

$$\boldsymbol{N}_j^{n,2} + \frac{h}{2} S_{n+1/2} \boldsymbol{\nu}_j^{n+1} + \frac{h}{2} K_{n+1} \left(\boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2} \right) = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}}. \tag{39}$$

We now consider the following change of variables

$$\boldsymbol{X}_j^n = \boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2}, \tag{40}$$

$$\boldsymbol{Y}_j^{n,1} = \boldsymbol{\nu}_j^{n+1} + \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2}, \tag{41}$$

$$\boldsymbol{Y}_j^{n,2} = \boldsymbol{\nu}_j^{n+1}, \tag{42}$$

which, upon substitution into (36)-(39), gives the set of equations

$$\boldsymbol{M}_j^{n,1} + \frac{h}{2} S_n \boldsymbol{X}_j^n - \frac{h}{2} K_{n+1/2} \boldsymbol{Y}_j^{n,1} = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}}, \tag{43}$$

$$\boldsymbol{M}_j^{n,2} + \frac{h}{2} S_{n+1} \boldsymbol{X}_j^n - \frac{h}{2} K_{n+1/2} \boldsymbol{Y}_j^{n,2} = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}}, \tag{44}$$

$$\boldsymbol{N}_j^{n,1} + \frac{h}{2} S_{n+1/2} \boldsymbol{Y}_j^{n,1} + \frac{h}{2} K_n \boldsymbol{X}_j^n = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}}, \tag{45}$$

$$\boldsymbol{N}_j^{n,2} + \frac{h}{2} S_{n+1/2} \boldsymbol{Y}_j^{n,2} + \frac{h}{2} K_{n+1} \boldsymbol{X}_j^n = \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}}. \tag{46}$$

By adding (43)-(44),

$$\boldsymbol{M}_j^{n,1} + \boldsymbol{M}_j^{n,2} = -\frac{h}{2} \left[\left(S_n + S_{n+1} \right) \boldsymbol{X}_j^n - K_{n+1/2} \left(\boldsymbol{Y}_j^{n,1} + \boldsymbol{Y}_j^{n,2} \right) \right] + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}} + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}}. \tag{47}$$

Similarly, by adding (45)-(46),

$$\boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2} = -\frac{h}{2} \left[S_{n+1/2} \left(\boldsymbol{Y}_j^{n,1} + \boldsymbol{Y}_j^{n,2} \right) + \left(K_n + K_{n+1} \right) \boldsymbol{X}_j^n \right] + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}} + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}}. \tag{48}$$

Thus, (34)-(35) can be rewritten as

$$\boldsymbol{\mu}_j^n - \boldsymbol{\mu}_j^{n+1} = -\frac{h}{2} \left[\left(S_n + S_{n+1} \right) \boldsymbol{X}_j^n - K_{n+1/2} \left(\boldsymbol{Y}_j^{n,1} + \boldsymbol{Y}_j^{n,2} \right) \right] + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,1}} + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}} \tag{49}$$

$$\boldsymbol{\nu}_j^n - \boldsymbol{\nu}_j^{n+1} = -\frac{h}{2} \left[S_{n+1/2} \left(\boldsymbol{Y}_j^{n,1} + \boldsymbol{Y}_j^{n,2} \right) + \left(K_n + K_{n+1} \right) \boldsymbol{X}_j^n \right] + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}} + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}} \tag{50}$$

By combining $\boldsymbol{X}_j^n = \boldsymbol{\mu}_j^{n+1} + \boldsymbol{M}_j^{n,2}$ and (44),

$$\boldsymbol{X}_j^n = \boldsymbol{\mu}_j^{n+1} - \frac{h}{2} S_{n+1} \boldsymbol{X}_j^n + \frac{h}{2} K_{n+1/2} \boldsymbol{Y}_j^{n,2} + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{U}_j^{n,2}}. \tag{51}$$

Similarly, by combining $\boldsymbol{Y}_j^{n,1} = \boldsymbol{\nu}_j^{n+1} + \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2}$ and (48),

$$\boldsymbol{Y}_j^{n,1} = \boldsymbol{\nu}_j^{n+1} - \frac{h}{2} \left[S_{n+1/2} \left(\boldsymbol{Y}_j^{n,1} + \boldsymbol{Y}_j^{n,2} \right) + \left(K_n + K_{n+1} \right) \boldsymbol{X}_j^n \right] + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,1}} + \frac{\partial \mathcal{J}^h}{\partial \boldsymbol{V}_j^{n,2}}. \tag{52}$$

The time-stepping scheme is completed by the relation

$$\boldsymbol{Y}^{n,2} = \boldsymbol{\nu}_j^{n+1}. \tag{53}$$

The scheme (49)-(53) may be written in the form of Lemma 5.4.1 by defining the slopes according to (5.61)-(5.64). This completes the proof of the lemma.

.16 Proof of Corollary 1

By rearranging (5.59) and (5.60),

$$\boldsymbol{\mu}_j^{n+1} = \boldsymbol{\mu}_j^n + \frac{h}{2}\left(\boldsymbol{\kappa}_j^{n,1} + \boldsymbol{\kappa}_j^{n,2}\right), \tag{54}$$

$$\boldsymbol{\nu}_j^{n+1} = \boldsymbol{\nu}_j^n + \frac{h}{2}\left(\boldsymbol{\ell}_j^{n,1} + \boldsymbol{\ell}_j^{n,2}\right). \tag{55}$$

Hence, $b_1^\mu = b_2^\mu = 1/2$ and $b_1^\nu = b_2^\nu = 1/2$.

To express the stage variables in standard form we substitute (54) into (5.65) and define $\boldsymbol{X}_j^{n,1} = \boldsymbol{X}_j^{n,2} = \boldsymbol{X}_j^n$. Similarly, we substitute (55) into (5.66) and (5.67), resulting in

$$\begin{aligned}
\boldsymbol{X}_j^{n,1} &= \boldsymbol{\mu}_j^n + \frac{h}{2}\boldsymbol{\kappa}_j^{n,1},\\
\boldsymbol{X}_j^{n,2} &= \boldsymbol{\mu}_j^n + \frac{h}{2}\boldsymbol{\kappa}_j^{n,1},\\
\boldsymbol{Y}_j^{n,1} &= \boldsymbol{\nu}_j^n,\\
\boldsymbol{Y}_j^{n,2} &= \boldsymbol{\nu}_j^n + \frac{h}{2}\left(\boldsymbol{\ell}_j^{n,1} + \boldsymbol{\ell}_j^{n,2}\right).
\end{aligned}$$

From these relations we can identify $a_{11}^\mu = a_{21}^\mu = 1/2$ and $a_{12}^\mu = a_{22}^\mu = 0$. Furthermore, $a_{11}^\nu = a_{12}^\nu = 0$ and $a_{21}^\nu = a_{22}^\nu = 1/2$.

For the case without forcing, the formulae for the slopes, (5.65)-(5.67), become

$$\boldsymbol{\kappa}_j^{n,1} = S_n \boldsymbol{X}_j^{n,1} - K_{n+1/2}\boldsymbol{Y}_j^{n,1}, \tag{56}$$

$$\boldsymbol{\kappa}_j^{n,2} = S_{n+1} \boldsymbol{X}_j^{n,2} - K_{n+1/2}\boldsymbol{Y}_j^{n,2}, \tag{57}$$

$$\boldsymbol{\ell}_j^{n,1} = K_n \boldsymbol{X}_j^{n,1} + S_{n+1/2}\boldsymbol{Y}_j^{n,1}, \tag{58}$$

$$\boldsymbol{\ell}_j^{n,2} = K_{n+1} \boldsymbol{X}_j^{n,2} + S_{n+1/2}\boldsymbol{Y}_j^{n,2}. \tag{59}$$

They are consistent approximations of the time derivatives $\dot{\boldsymbol{\mu}}(t_n)$ and $\dot{\boldsymbol{\nu}}(t_n)$, respectively. The scheme is therefore a consistent approximation of the continuous adjoint system.

.17 Computing the gradient of the discrete objective function

Given a solution that satisfies the saddle point conditions of (5.56) and (5.57), the gradient of $\mathcal{L}_h(\boldsymbol{\alpha})$ satisfies

$$\frac{d\mathcal{L}_h}{d\alpha_r} = \frac{\partial \mathcal{J}_{1h}}{\partial \alpha_r}(\boldsymbol{u}, \boldsymbol{v}) + \frac{\partial \mathcal{J}_{2h}}{\partial \alpha_r}(\boldsymbol{U}, \boldsymbol{V}), \quad r = 1, 2, \ldots, D.$$

The gradient of $\mathcal{L}_h$ with respect to $\boldsymbol{\alpha}$ only gets a contribution from the terms in T_j^q that involve the matrices K and S. Let $S'_n = \partial S/\partial \alpha_r(t_n)$ and $K'_n = \partial K/\partial \alpha_r(t_n)$. We have,

$$\begin{aligned}
\frac{\partial T_j^1}{\partial \alpha_r} &= -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle S'_n \boldsymbol{U}_j^{n,1} - K'_n \boldsymbol{V}_j^{n,1} + S'_{n+1}\boldsymbol{U}_j^{n,2} - K'_{n+1}\boldsymbol{V}_j^{n,2}, \boldsymbol{\mu}_j^{n+1}\right\rangle_2, \\
\frac{\partial T_j^2}{\partial \alpha_r} &= -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle K'_{n+1/2}\left(\boldsymbol{U}_j^{n,1} + \boldsymbol{U}_j^{n,2}\right) + S'_{n+1/2}\left(\boldsymbol{V}_j^{n,1} + \boldsymbol{V}_j^{n,2}\right), \boldsymbol{\nu}_j^{n+1}\right\rangle_2, \\
\frac{\partial T_j^3}{\partial \alpha_r} &= 0, \\
\frac{\partial T_j^4}{\partial \alpha_r} &= -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle S'_n \boldsymbol{U}_j^{n,1} - K'_n \boldsymbol{V}_j^{n,1} + S'_{n+1}\boldsymbol{U}_j^{n,2} - K'_{n+1}\boldsymbol{V}_j^{n,2}, \boldsymbol{M}_j^{n,2}\right\rangle_2, \\
\frac{\partial T_j^5}{\partial \alpha_r} &= -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,1} + S'_{n+1/2}\boldsymbol{V}_j^{n,1}, \boldsymbol{N}_j^{n,1}\right\rangle_2, \\
\frac{\partial T_j^6}{\partial \alpha_r} &= -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,1} + S'_{n+1/2}\boldsymbol{V}_j^{n,1}, \boldsymbol{N}_j^{n,2}\right\rangle_2.
\end{aligned}$$

We note that

$$\frac{\partial (T_j^5 + T_j^6)}{\partial \alpha_r} = -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,1} + S'_{n+1/2}\boldsymbol{V}_j^{n,1}, \boldsymbol{N}_j^{n,1} + \boldsymbol{N}_j^{n,2}\right\rangle_2.$$

Let $\boldsymbol{X}_j^n$ and $\boldsymbol{Y}_j^{n,i}$ be defined by (40)-(42). We have,

$$\begin{aligned}
\frac{\partial T_j^4}{\partial \alpha_r} &= -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle S'_n \boldsymbol{U}_j^{n,1} - K'_n \boldsymbol{V}_j^{n,1} + S'_{n+1}\boldsymbol{U}_j^{n,2} - K'_{n+1}\boldsymbol{V}_j^{n,2}, \boldsymbol{X}_j^n - \boldsymbol{\mu}_j^{n+1}\right\rangle_2, \\
\frac{\partial (T_j^5 + T_j^6)}{\partial \alpha_r} &= -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,1} + S'_{n+1/2}\boldsymbol{V}_j^{n,1}, \boldsymbol{Y}_j^{n,1} - \boldsymbol{\nu}_j^{n+1}\right\rangle_2.
\end{aligned}$$

Thus,

$$\frac{\partial (T_j^1 + T_j^4)}{\partial \alpha_r} = -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle S'_n \boldsymbol{U}_j^{n,1} - K'_n \boldsymbol{V}_j^{n,1} + S'_{n+1}\boldsymbol{U}_j^{n,2} - K'_{n+1}\boldsymbol{V}_j^{n,2}, \boldsymbol{X}_j^n\right\rangle_2.$$

Furthermore, from the relation (42),

$$\begin{aligned}\frac{\partial(T_j^2+T_j^5+T_j^6)}{\partial\alpha_r} =& -\frac{h}{2}\sum_{n=0}^{M-1}\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,1}+S'_{n+1/2}\boldsymbol{V}_j^{n,1},\boldsymbol{Y}_j^{n,1}\right\rangle_2\\ &-\frac{h}{2}\sum_{n=0}^{M-1}\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,2}+S'_{n+1/2}\boldsymbol{V}_j^{n,2},\boldsymbol{Y}_j^{n,2}\right\rangle_2,\end{aligned}$$

We can further simplify the expressions by recognizing that $\boldsymbol{V}^{n,1}=\boldsymbol{V}^{n,2}$. By collecting the terms,

$$\begin{aligned}\frac{\partial\mathcal{L}_h}{\partial\alpha_r} = \frac{h}{2}\sum_{j=0}^{E-1}\sum_{n=0}^{M-1}\Big(&\left\langle S'_n\boldsymbol{U}_j^{n,1}+S'_{n+1}\boldsymbol{U}_j^{n,2}-(K'_n+K'_{n+1})\boldsymbol{V}_j^{n,1},\boldsymbol{X}_j^n\right\rangle_2\\ &+\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,1}+S'_{n+1/2}\boldsymbol{V}_j^{n,1},\boldsymbol{Y}_j^{n,1}\right\rangle_2\\ &+\left\langle K'_{n+1/2}\boldsymbol{U}_j^{n,2}+S'_{n+1/2}\boldsymbol{V}_j^{n,1},\boldsymbol{Y}_j^{n,2}\right\rangle_2\Big).\end{aligned}$$

This completes the proof of the lemma.