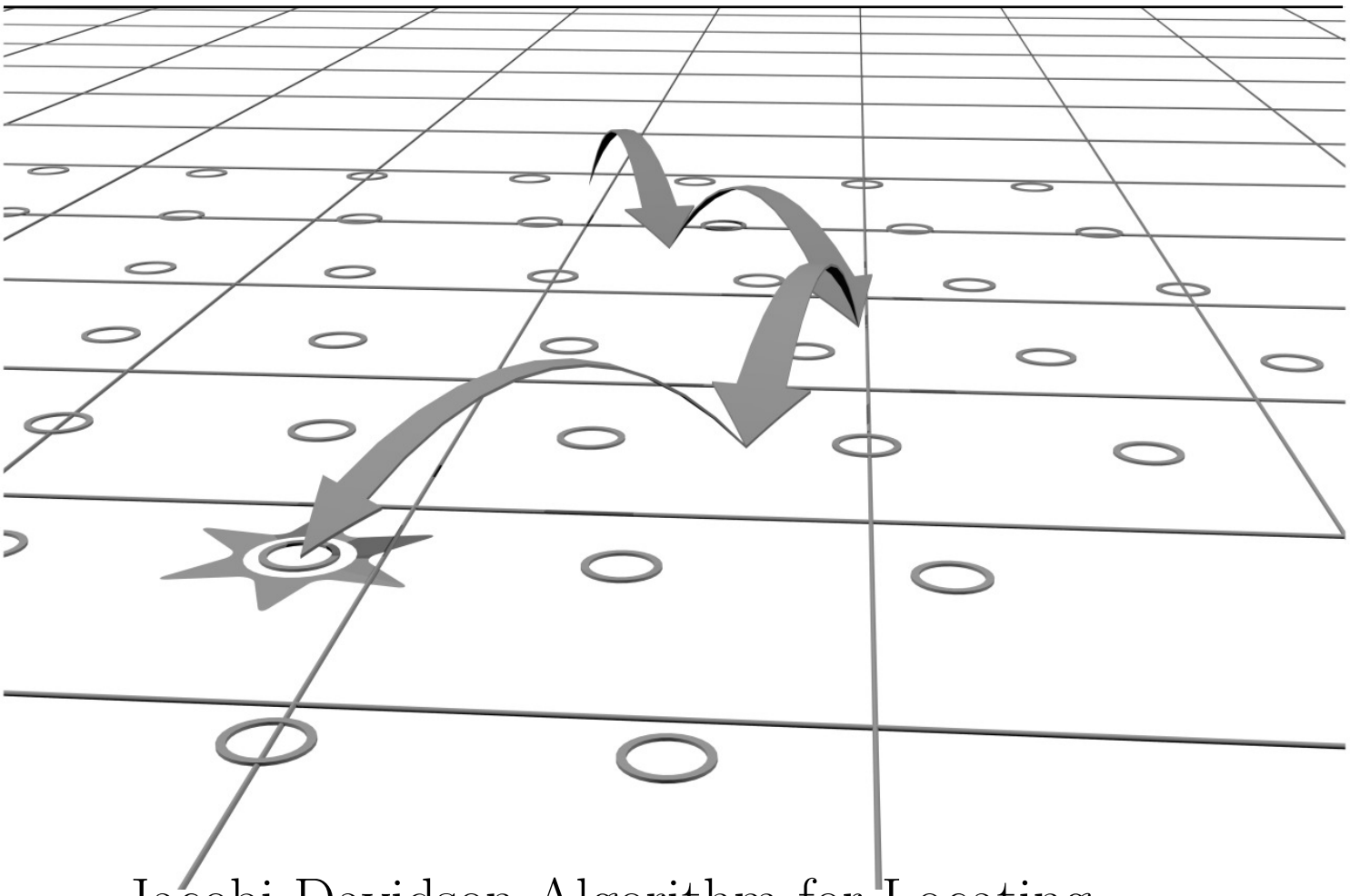

Jacobi-Davidson Algorithm for Locating Resonances in a Few-Body Tunneling System

Bachelor Thesis in Physics and Engineering Physics

Gustav Hjelmare
Jonathan Larsson
David Lidberg
Sebastian Östnell

Department of Fundamental Physics
Division of Subatomic Physics
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2014

Jacobi-Davidson Algorithm for Locating Resonances in a Few-Body Tunneling System

Gustav Hjelmare Jonathan Larsson
David Lidberg Sebastian Östnell

Supervisors:
Christian Forssén Jimmy Rotureau

Examiner:
Daniel Persson

June 9, 2014
Department of Fundamental Physics
Chalmers University of Technology

The Jacobi-Davidson Algorithm for Locating Resonances in a Few-Body Tunneling System.
Authors: Gustav Hjelmare[a], Jonathan Larsson[b], David Lidberg[c], Sebastian Östnell[d]

Email:
hjelmare@student.chalmers.se[a]
jonl@student.chalmers.se[b]
davlid@student.chalmers.se[c]
gusostse@student.gu.se[d]

FUFX02 - Bachelor thesis and Fundamental Physics Bachelor Thesis No. FUFX02-14-02
Supervisors: Christian Forssén, Jimmy Rotureau
Examiner: Daniel Persson

Department of Fundamental Physics
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
+46 (31) 772 1000

Printed by Chalmers reproservice
Göteborg, Sweden 2014

Cover: Artist's view of eigenvalue spectra and path of convergence of the Jacobi-Davidson algorithm.

Abstract

A recent theoretical study of quantum few-body tunneling implemented a model using a Berggren basis expansion. This approach leads to eigenvalue problems, involving large, complex-symmetric Hamiltonian matrices. In addition, the eigenspectrum consists mainly of irrelevant scattering states. The physical resonance is usually hidden somewhere in the continuum of these scattering states, making diagonalization difficult.

This thesis describes the theory of the Jacobi-Davidson algorithm for calculating complex eigenvalues and thus identifying the resonance energies of interest. The underlying Davidson method is described and combined with Jacobi's orthogonal complement method to form the Jacobi-Davidson algorithm. The algorithm is implemented and applied to matrices from the theoretical study. Furthermore, a non-hermitian formulation of quantum mechanics is introduced and the Berggren basis expansion explained.

The results show that the ability of the Jacobi-Davidson algorithm to locate a specific interior eigenvalue greatly reduces the computational times compared to previous diagonalization methods. However, the additional computational cost of implementing the Jacobi correction turns out to be unnecessary in this application; thus, the Davidson algorithm is sufficient for finding the resonance state of these matrices.

Acknowledgements

We wish to give a special thanks to our supervisors Christian Forssén and Jimmy Rotureau for their support, great feedback and insightful discussions throughout the entire project.

Besides our supervisors we wish to thank Rikard Lundmark for his support and feedback. Without his recent master thesis providing the matrices needed, this project would not have been the same. We would also like to express our gratitude to the Division of Subatomic Physics for the access to their computers, making the calculations needed for this project possible.

Lastly we wish to thank Filip Hjort, Rebecka Jacobsson and Erik Silfverswärd for the valuable feedback during draft exchanges throughout the bachelor project.

Gustav Hjelmare,
Jonathan Larsson,
David Lidberg,
Sebastian Östnell,
Göteborg 19/5 – 14

Contents

1

Introduction

Quantum mechanics postulates that observable quantities are represented by linear hermitian operators, such as the Hamiltonian operator representing the energy of a system. The eigenvalues of the operator correspond to the measurable values of the observable. Allowing non-hermitian operators introduces complex eigenvalues, representing states with finite lifetimes. This is a useful approach e.g. when studying quantum mechanical tunneling.

In recent experiments at the University of Heidelberg [1] [2], systems of ultra-cold ^{6}Li atoms in an optomagnetical trap [1] were prepared, and the behaviour of the atoms as they tunneled out of the trap was studied. Due to its small size and low energy, such a system requires a quantum mechanical description. For his master's thesis, Rikard Lundmark has modeled [3] a system like the one studied in Heidelberg. In order to calculate tunneling rates, for comparison with those calculated from experimental data, one particular eigenvalue of a large, dense, non-hermitian matrix needs to be determined.

Several common eigensolving algorithms find the largest or smallest eigenvalues of the eigenvalue spectrum. By first applying a spectral shift-and-invert operation[2] around an eigenvalue approximation, followed by applying a common eigensolving algorithm, eigenvalues in the interior of the spectrum can be found. This strategy can be successful, especially if the spectrum is not very dense in the region of interest.

For the large matrices encountered by Lundmark when modeling the Heidelberg experiments, the eigenvalue of interest is close to other eigenvalues. The approach outlined above then requires the calculation of many eigenvalues around the initial approximation, in order to find the one eigenvalue of interest. In this case the approach is not very effective, since most of the eigenvalues found are eventually discarded.

The method investigated in this thesis, which is based on a paper by Sleijpen and van der Vorst [4] that combines work done by Davidson [5] and Jacobi [6], is well-suited to non-hermitian matrices. The method has also shown promise for being able to search for eigenvalues in the interior of the spectrum.

[1] A magnetic field enclosure with laser-cooling of particles.

[2] A spectral shift-and-invert transforms the eigenvalues of a matrix in such a way that the eigenvalues close to some value τ become very large, while the corresponding eigenvector remains the same.

1.1 Background

In quantum mechanics, the state of a particle can be expressed as a wave function $\psi(t,\mathbf{r})$ which can be written as a product of a time-dependent and position-dependent part

$$\psi(t,\mathbf{r}) = \psi_t(t)\psi_r(\mathbf{r}). \tag{1.1}$$

These states evolve in time according to the time-dependent Schrödinger equation

$$i\hbar\frac{\partial}{\partial t}|\psi\rangle = \mathrm{H}|\psi\rangle, \tag{1.2}$$

where H is the Hamiltonian operator. If the Hamiltonian operator is time-independent, the equation can be rewritten as

$$\psi(t,\mathbf{r}) = e^{-\frac{iE}{\hbar}t}\psi_r(\mathbf{r}) \tag{1.3}$$

$$\mathrm{H}\psi_r(\mathbf{r}) = E\psi_r(\mathbf{r}), \tag{1.4}$$

where E is the energy associated with the state and equation (1.4) is referred to as the time-independent Schrödinger equation. If the energy E is real, the exponential factor in equation (1.3) constitutes a complex phase shift over time of the state $\psi_r(\mathbf{r})$ and has no effect on the probability distribution $|\psi(t,\mathbf{r})|^2$.

When modeling quantum systems, two distinct types of systems exist: closed systems and open systems. Closed systems are limited by infinite potential barriers on both sides, giving rise to an infinite number of *bound states*. A common example is the harmonic oscillator potential shown in Figure 1.1. The energy levels of a closed system are discrete, and represent states with infinite lifetimes.

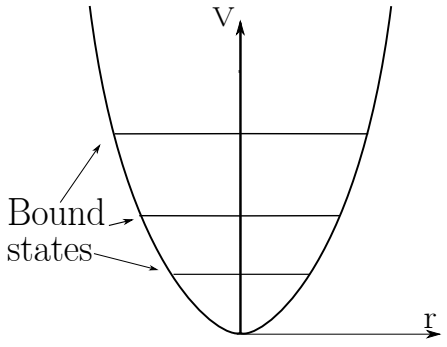

Figure 1.1: A closed quantum system with an infinite number of bound states

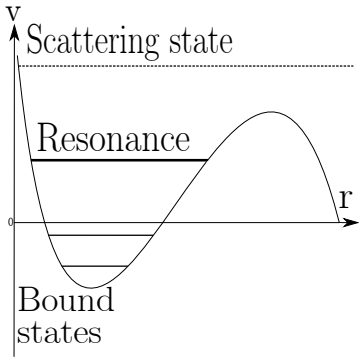

Figure 1.2: An open quantum system allowing particles to tunnel in and out

In an open quantum system as shown in Figure 1.2, the height of the potential barrier is finite, and there are a limited number of bound states with corresponding discrete energies. At higher energies, a continuum of unbound states exist, which are called *scattering states*. In the continuum there is also another class of states called *resonance states*. The resonance states are quasi-bound states representing particles in temporary interaction with the system. These resonance states are useful when describing tunneling processes.

Unlike bound and unbound states, resonance states can be associated with a complex energy

$$E = E_0 - i\frac{\Gamma}{2}. \tag{1.5}$$

Inserting a complex energy in equation (1.3) changes the phase factor to also include an exponential decay of the probability distribution

$$|\psi(t,\mathbf{r})|^2 = |e^{-\frac{iE_0}{\hbar}t}e^{-\frac{\Gamma}{2\hbar}t}\psi_r(\mathbf{r})|^2 = e^{-\frac{\Gamma}{\hbar}t}|\psi_r(\mathbf{r})|^2 \tag{1.6}$$

The parameter Γ, also known as the width of the resonance state, is related to the tunneling rate γ as

$$\gamma = \frac{\Gamma}{\hbar}. \tag{1.7}$$

This means that by using complex energies in the quantum mechanical formulation of the system one can describe decaying states using the time-independent formalism.

To be able to generate these complex energies in the time-independent Schrödninger equation (1.4), the Hamiltonian operator can no longer be hermitian. Thus, we have a *non-hermitian* formulation of a quantum system.

1.2 Purpose and Outline

The purpose of this project is to adapt and implement an eigensolving algorithm for use with large non-hermitian matrices arising from a quantum mechanical tunneling problem. The algorithms studied are the Jacobi-Davidson and the Davidson algorithms, which have previously shown promise for non hermitian matrices [7]. In our thesis we also present some of the basic theory of the origin of these matrices, in order to give our work a clear context.

In Chapter 2, we present some fundamental concepts of basis expansion in the complex-momentum *Berggren basis*. The theory of expressing two-body states is also introduced.

The complex symmetric matrices studied in this thesis originate from a masters thesis by Rikard Lundmark [3]. Lundmark's project used the Berggren basis expansion to model a tunneling system with two particles. The details on this topic is presented in Chapter 3.

Chapter 4 presents some theory and the implementation of the Jacobi-Davidson algorithm. The properties of the algorithm are discussed and some important details of the implementation are explained. The results of applying the Jacobi-Davidson algorithm to matrices calculated by Lundmark is presented in Chapter 5 along with an example of some calculations that were previously too costly, but become possible with this algorithm.

The behaviour of the Jacobi-Davidson algorithm and comparisons with another algorithm is discussed in Chapter 6.

As a main goal we offer an alternative to common methods of extracting single eigenvalues from this type of matrices, and highlight the advantages of using the eigensolving algorithms presented in chapter 4.

2

Theory

Most realistic quantum mechanical problems are far too complex to solve analytically. We instead have to rely on numerical methods. In our case this means we have to express the two-particle Schrödinger equation on a matrix eigenvalue form. This chapter demonstrates how this is accomplished by first deriving the matrix formulation of the Schrödinger equation in a single-particle basis. Then the complex-momentum Berggren basis is presented, and the need for discretization of a continous basis is discussed. Lastly we demonstrate how a single-particle basis is expanded to a two-particle basis.

2.1 Basis Expansion

Consider the time-independent Schrödinger equation

$$\mathrm{H}\,|\psi\rangle = E\,|\psi\rangle\,, \tag{2.1}$$

where the state $|\psi\rangle$ is expressed in an orthonormal basis $\{\psi_i\}$ such that

$$\langle\psi_i\,|\psi_j\rangle = \delta_{ij}. \tag{2.2}$$

Furthermore, in order for the basis to span the full Hilbert space of the system it needs to be a complete basis, meaning we can formulate the *resolution of identity* in this basis as

$$\mathrm{I} = \sum_n |\psi_n\rangle\langle\psi_n|, \tag{2.3}$$

where I is the identity operator.

We use this fundamental relation to derive the matrix formulation of equation (2.1). After insertion of the resolution of identity and multiplication from the left with $\langle\psi_m|$ we arrive at

$$\sum_n \langle\psi_m|\mathrm{H}|\psi_n\rangle\psi_n = E\psi_m. \tag{2.4}$$

With this formulation of the Schrödinger equation we can compute the matrix elements $\mathrm{H}_{mn} = \langle\psi_m|\mathrm{H}|\psi_n\rangle$ of the Hamiltonian expressed in this basis.

2.2 The Berggren basis

The above section used a discrete basis to derive a matrix formulation of the Schrödinger equation, but when studying resonances, scattering states must be included. For this purpose the continuous momentum basis can be used. The continuous basis states are denoted by $|\mathbf{k}\rangle$ in contrast to the previous discrete basis states $|\psi\rangle$.

The same requirements that apply for a discrete basis (see Section 2.1) also apply to a continuous one. The orthonormality condition now becomes

$$\langle \mathbf{k}|\mathbf{k}'\rangle = \delta(\mathbf{k}-\mathbf{k}'), \tag{2.5}$$

and the corresponding resolution of identity for a continuous basis is

$$\mathrm{I} = \int \mathrm{d}\mathbf{k}\, |\mathbf{k}\rangle \langle \mathbf{k}|\,. \tag{2.6}$$

In the momentum basis $|\mathbf{k}\rangle$ are the eigenstates of the momentum operator $\mathbf{p} = -i\hbar\nabla$ representing the momentum $\mathbf{p}$ of a particle. Momentum is related to a wave vector via

$$\mathbf{k} = \frac{\mathbf{p}}{\hbar}. \tag{2.7}$$

The energy solutions of the Schrödinger equation in the momentum basis are related to the wavenumber k by

$$E = \frac{\hbar^2 k^2}{2m} \quad k = \frac{\sqrt{2mE}}{\hbar}. \tag{2.8}$$

In scattering theory the states of the system are interpreted as poles in the complex energy plane [8]. Allowing complex energies as in Chapter 1, equations (2.8) give the placement of k in the complex plane. If one illustrates the solutions of the Schrödinger equation in the momentum basis on the complex momentum plane as in Figure 2.1, the energy of bound states $E < 0$ are represented by discrete points on the imaginary axis, while the energy of unbound states $E > 0$ are a continuum along the real axis.

In the momentum basis the bound states together with the real continuum states form the completeness relation

$$\mathrm{I} = \sum_{n \in \, bound \; state} |n\rangle \langle n| + \int_0^\infty \mathrm{d}\xi\, |\xi\rangle \langle \xi|\,. \tag{2.9}$$

where $|\xi\rangle$ are the scattering eigenstates.

The resonance states, with their complex energies, are located in the fourth quadrant of the complex plane, outside the usual contour used in scattering theory [9]. Altering the contour to include the resonance poles would allow for the calculation of the resonance states.

In 1968 Tore Berggren proved [9] that the real segment of the contour can be deformed to a new basis that includes the resonance poles. The contour proposed by Berggren can be seen in Figure 2.2.

The continuum states along the L_+ segment combined with resonance and bound states encircled by the contour form the *Berggren basis* and allow the formulation of the Berggren completeness relation The resonance states are mirrored in the imaginary axis, and can be

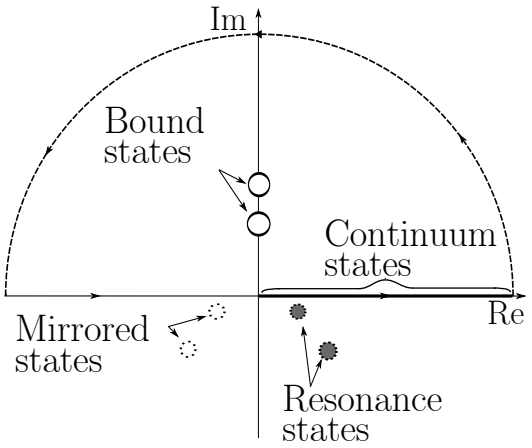

Figure 2.1: Complex momentum plane with standard contour

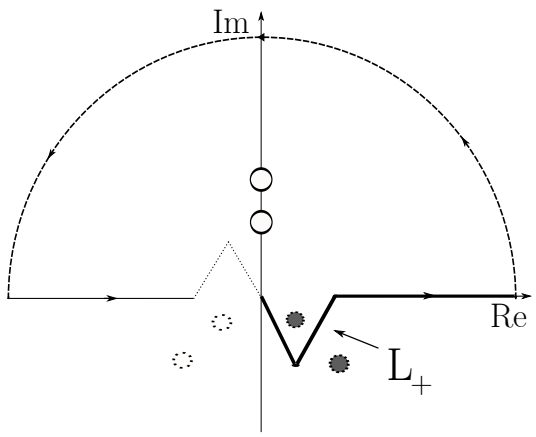

Figure 2.2: Complex momentum plane with Berggren contour

seen as incoming and outgoing resonances. Berggren showed that this symmetry leads to a new inner product without conjugation

$$\langle \mathbf{k}|\mathbf{k}'\rangle = \int \mathrm{d}\mathbf{k}\phi(\mathbf{k})\phi(\mathbf{k}'). \tag{2.10}$$

This is different from the conjugated inner product used in hermitian formulation

$$\langle \mathbf{k}|\mathbf{k}'\rangle = \int \mathrm{d}\mathbf{k}\phi(\mathbf{k})\overline{\phi(\mathbf{k}')}. \tag{2.11}$$

which is not used here.

The eigenstates of a system will in general change along with the contour used to describe the system, but since the resonance pole is a physical phenomenon, it will not change. Using this fact one can confirm whether an eigenstate generated by the discretized Berggren basis corresponds to the resonance pole, by altering the contour of the Berggren basis and verifying that the resonance pole remains unchanged.

2.2.1 Discretization

A continuous basis does not allow for a matrix representation of the Hamiltonian. Thus, continuous bases need to be discretized.

The process of discretization transforms the continuum of basis states into a finite number of basis states by turning the integral into a sum over these states.

$$\int_a^b f(x)\mathrm{dx} \approx \sum_{i=1}^{n} w_i f(x_i), \tag{2.12}$$

where the integrand is evaluated at a set number of discrete points x_i all weighted by a corresponding w_i. The weights are determined by the choice of quadrature [10]. This process is necessary for the numerical calculation, in order to allow for a matrix representation of the Hamiltonian according to equation (2.4).

2.3 Expansion to Two-Particle Basis

Above we demonstrated how the Schrödinger equation can be expanded into a matrix eigenvalue problem, and introduced the Berggren basis. So far, though, we have considered only the case of one particle. Now we consider a system of two particles. This is done by constructing a two-particle basis from two suitable one-particle bases as described in [8] and [11].

In the case for two particles in the single-particle basis states $|i\rangle$ and $|j\rangle$, we construct the two-particle basis state

$$|i,\, j\rangle = |i\rangle \otimes |j\rangle \tag{2.13}$$

as a tensor product of the single-particle basis states. Orthonormality for these two-particle states demands that

$$\langle i,\, j|k,\, l\rangle = \delta_{ik}\delta_{jl}. \tag{2.14}$$

Often when dealing with many-body states careful consideration has to be taken to account for symmetry, due to the particles being indistinguishable. However, the experimental background for this thesis, as will be described in Section 3, consists of two *distinguishable* atoms, meaning that in this specific case we do not have to consider any special symmetry conditions for our two-particle states.

The two-particle expansion leads to a basis where each state is represented by two indices (i,j), which in turn means that a matrix representation of the Hamiltonian has four indices. To avoid this inconvenience, we can list the two-particle states in some arbitrary order, and use their position in the list as the new index. The order used here will be

$$(1,1),(1,2),...,(2,1),(2,2)....,(M,N)$$

where M and N are the number of basis states in the respective single-particle bases. For the remainder of this thesis, both single-particle bases will have the same number of basis states, thus $M = N$.

With this simplification the elements of a two-particle Hamiltonian matrix can be calculated as

$$\mathrm{H}_{(i,\, j)(k,\, l)} = \langle i,\, j|\mathrm{H}|k,\, l\rangle = \langle m|\mathrm{H}|n\rangle = \mathrm{H}_{mn}. \tag{2.15}$$

From this equation we note that the dimension of H will be the product of the number of basis states in the respective single-particle bases. The choice of single-particle basis will depend on the system in question, and if it is continuous it needs to be discretized to some finite dimension N. A larger single-particle basis dimension results in more accurate approximations but also increases the dimension of the two-particle basis significantly, thus increasing computational cost.

3

Model Background

The matrices investigated in this thesis were provided by Rikard Lundmark from his recent study of tunneling theory describing optomagnetical traps [3]. In this section we give a brief summary of the experimental and theoretical background from which these matrices were calculated. For more detailed information on this topic, see Lundmark's thesis [3].

3.1 The Heidelberg Experiments

In experiments by G. Zürn et al. at the University of Heidelberg [1][2], ultracold ^{6}Li atoms were confined in an optomagnetical trap. Measurements were made of the number of particles in the trap as a function of time, and from these measurements the tunneling rates were calculated. The trap was elongated and asymmetric, so that the system was effectively one-dimensional and with tunneling occuring only in one direction.

Due to their 1/2 spin, the ^{6}Li atoms behave as fermions, with the $|\uparrow\rangle$ and $|\downarrow\rangle$ states experiencing slightly different trapping potentials. Additionally, there was a short-ranged interaction between particles in different spin states, tunable by varying the magnetic field strength in the trap. This interaction alters the total binding energy of the states. This grants a possibility to study the tunneling rate for different interaction strengths. The model used by Lundmark describes systems of two atoms in the trap, and models the interaction as a contact interaction.

3.2 Optomagnetical Potential

The potential trapping the ^{6}Li atoms can be described as the sum of three components

$$V(x,y,z) = V_x(x) + V_y(y) + V_z(z) \tag{3.1}$$

where the x- and y-components are given by

$$V_x(x) = pV_{0r}\left(1 - e^{-\frac{2x^2}{w_{0x}^2}}\right) \tag{3.2}$$

$$V_y(y) = pV_{0r}\left(1 - e^{-\frac{2y^2}{w_{0y}^2}}\right) \tag{3.3}$$

and the z-component by

$$V_z(z) = pV_0\left(1 - \frac{1}{1 + \left(\frac{z}{z_r}\right)^2}\right) - c_{B|state\rangle}\mu_m B' z, \tag{3.4}$$

where B' is a magnetic field gradient and the factor $c_{B|state\rangle}$ describes the spin dependence of the potential[1].

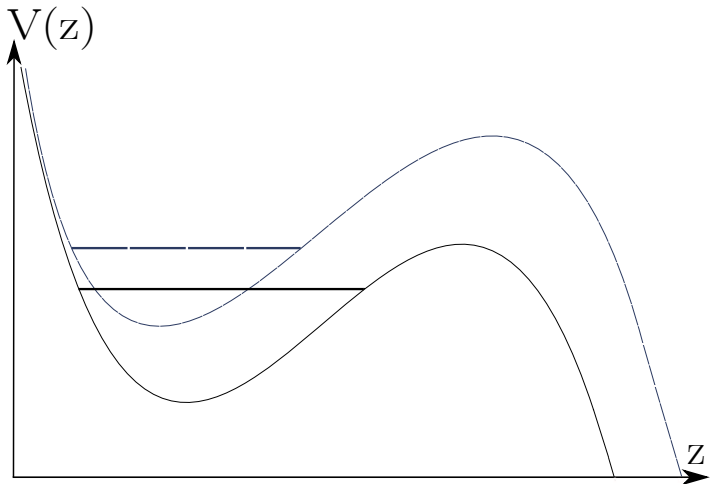

Figure 3.1: The potential $V_z(z)$ differs for particles in different spin states, due to the factor $c_{B|state\rangle}$. For both spin states, the potential increases significantly in the negative z-direction, but drops off in the positive z-direction. Thus, particles can only tunnel out of the trap in the positive z-direction.

Thus, the trap is symmetric in the x- and y-directions, but not in the z-direction. The second term of (3.4) gives rise to both a spatial asymmetry along the z-axis, and a difference in the potential seen by atoms in different spin states, as seen in Figure 3.1. The asymmetry in the z-direction limits tunneling out of the trap to only happen in one direction. Experimental parameters were chosen to make the trap shallow, so that only one single-particle resonance state exists, and effectively one-dimensional[2], thus enabling the use of a one-dimensional model.

[1] For details concerning the remaining parameters in equations (3.2), (3.3) and (3.4), see tables 2.1 and 2.2 in Lundmark's thesis [3].

[2] In a harmonic oscillator approximation of the trapping potential, $\omega_z : \omega_\perp \approx 1 : 10$.

3.3 The Two Particle Hamiltonian

The total energy of the system is the sum of the energies of the individual particles and the inter-particle interaction. This is represented by the Hamiltonian

$$\mathrm{H} = \mathrm{H}_{|\uparrow\rangle} + \mathrm{H}_{|\downarrow\rangle} + \mathrm{V}_{\uparrow\downarrow}. \tag{3.5}$$

Here $\mathrm{H}_{|\uparrow\rangle}$ and $\mathrm{H}_{|\downarrow\rangle}$ are the Hamiltonians for the $|\uparrow\rangle$ and $|\downarrow\rangle$ state particles respectively, representing their energies due to the potential (3.4). $\mathrm{V}_{\uparrow\downarrow}$ is a term representing the short-ranged inter-particle interaction. In Lundmark's thesis, this is modeled as a contact interaction $\mathrm{V}_{\uparrow\downarrow} = g\delta(z_\uparrow - z_\downarrow)$. This interaction is directly proportional to the *coupling coefficient* g, which corresponds to controlling the interaction strength in the experiments by varying the magnetic field strength in the trap [3].

The single-particle bases used consist of the eigenstates of the single-particle Hamiltonians, i.e. the solutions $|i\rangle_{|\uparrow\rangle}$ and $|j\rangle_{|\downarrow\rangle}$ to $\mathrm{H}_{|\uparrow\rangle}\,|i\rangle_{|\uparrow\rangle} = E\,|i\rangle_{|\uparrow\rangle}$ and $\mathrm{H}_{|\downarrow\rangle}\,|j\rangle_{|\downarrow\rangle} = E\,|j\rangle_{|\downarrow\rangle}$, which have been calculated using a discretized Berggren basis.

These two single-particle bases are combined according to Section 2.3, and the Hamiltonian in equation (3.5) is expressed in this two-particle basis. Thus the matrix representations of $\mathrm{H}_{|\uparrow\rangle}$ and $\mathrm{H}_{|\downarrow\rangle}$ have non-zero elements only on the diagonal, but the interaction term gives a dense distribution of non-zero matrix elements. However, the interaction is typically much smaller than the particle energies, and so H is dense but with diagonal elements much larger than off-diagonal elements.

Due to the way the two-particle basis is generated, the number of basis states is the product of the number of basis states in the respective single-particle bases. Thus, for single-particle bases with 160 to 200 basis states, the two-particle basis has between 25600 and 40000 states. The Hamiltonian has the same size as the two-particle basis, so in the extreme case the matrix under consideration is of dimension 40000×40000.

This is the matrix representation of the Hamiltonian in equation (3.5) that appears in the eigenvalue problem

$$\mathrm{H}\psi = E\psi \tag{3.6}$$

where ψ is a two-particle state expressed in the same basis. The eigenvalue corresponding to the resonance state determines the tunneling rates according to equation (1.7).

3.4 Pole Approximation

Due to the difference in the potential felt by atoms in different spin states, the single-particle bases consisting of eigenstates of the single-particle Hamiltonians differ depending on which spin state is considered. Thus, the basis expansion to two-particle states is done using two slightly different Berggren bases for the two particles. This means that careful consideration of the particle order is needed.

The resonance state for each particle in their specific single-particle basis can be identified, as discussed in Section 2.2. The two-particle state corresponding to the two particles in their respective resonance state, is denoted the *pole approximation*. If the contribution of the interaction is small compared to the individual energies of the particles, the resonance state of the system is expected to have a large overlap with the pole approximation. For this reason the pole approximation is used as the starting vector in the algorithms of the next chapter.

4

Jacobi-Davidson Algorithm

As we have seen in the previous chapter, the system of two interacting particles leads to a dense Hamiltonian matrix of large dimension. In order to study the energy of resonances in the system we look for the eigenvalue of this matrix corresponding to the resonance state. While there are several algorithms available for finding extreme eigenvalues of hermitian matrices, we are looking for a particular eigenvalue close to a pole approximation in the interior part of the eigenvalue spectrum of a non-hermitian matrix. This problem is less well-understood.

Most modern eigenvalue algorithms are based on the idea of projecting the matrix onto a subspace and finding the eigenvalues of the projection, thus approximating the eigenvalues of the original matrix [10]. Algorithms previously used for finding the resonance eigenvalues, such as the Lanczos and Arnoldi methods, employ a Krylov subspace for this purose. A Krylov subspace is spanned by the image of a vector under powers of the given matrix[1]. These methods search for extreme eigenvalues and therefore require several spectral shifts and calculation of many unwanted eigenvalues in order to find the resonance. The methods are not very effective in our application and computational times are several hours.

The algorithm described in this section is the Jacobi-Davidson algorithm [4] and uses a more general subspace. It is based on an eigensolving algorithm by Davidson combined with a rarely used eigenvalue iteration by Jacobi. In this chapter we first describe the Davidson algorithm and the Jacobi iteration, and then move on to the full implementation of the Jacobi-Davidson algorithm.

The algorithms described in this section were implemented in the interpreted programming language MATLAB [12].

4.1 The Davidson Method

The Davidson method was introduced by Ernest R. Davidson in 1975 [5] as an iterative method for finding eigenvalues of diagonally dominant matrices. Since then it has been modified and adapted to various matrices with different properties [13] [14]. Here we will present the basic Davidson algorithm needed for our purpose.

[1] For details on Krylov subspaces, see Appendix A

Let ($\mathbf{u}$, λ) be a solution to the standard eigenvalue problem of a $n \times n$ matrix A

$$\mathrm{A}\mathbf{u} = \lambda\mathbf{u}. \tag{4.1}$$

In the search for an approximation of $\mathbf{u}$, the Davidson method uses a subspace $\mathcal{K}_k$ of dimension $k < n$, $\mathcal{K}_k = \mathrm{span}\{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_k\}$ called a search space. Any approximation $\hat{\mathbf{u}}_k$ is then expressed as a linear combination of the orthogonal search space base vectors $\{\mathbf{v}_i\}_1^k$ as

$$\hat{\mathbf{u}}_k = s_1\mathbf{v}_1 + s_2\mathbf{v}_2 + ... + s_k\mathbf{v}_k \equiv \mathrm{V}_k\mathbf{s}_k. \tag{4.2}$$

The best approximation ($\hat{\mathbf{u}}_k$,θ_k) to the eigenpair ($\mathbf{u}$, λ) is such that the *Galerkin condition* holds

$$\mathrm{A}\hat{\mathbf{u}}_k - \theta_k\hat{\mathbf{u}}_k = \mathrm{A}\mathrm{V}_k\mathbf{s}_k - \theta_k\mathrm{V}_k\mathbf{s} \perp \mathcal{K}_k. \tag{4.3}$$

By multiplying the expression above with V_k^T from the left, the matrix is projected onto the search space, and the eigenvalues of the projected matrix $\mathrm{M}_k = \mathrm{V}_k^T\mathrm{A}\mathrm{V}_k$ are obtained from

$$\mathrm{V}_k^T\mathrm{A}\mathrm{V}_k\mathbf{s}_k - \theta_k\mathbf{s}_k = \mathrm{M}_k\mathbf{s} - \theta_k\mathbf{s}_k = 0. \tag{4.4}$$

The approximations $\hat{\mathbf{u}}_k = \mathrm{V}_k\mathbf{s}_k$ and θ_k are called *Ritz pairs* of A with respect to the subspace $\mathcal{K}_k$. Since the matrix M has dimension k there are k such Ritz pairs. Each Ritz pair is an approximation for a true eigenpair of the matrix A. This method is the foundation of many iterative eigensolving algoritms and likewise for the Davidson method.

The next step in obtaining a better approximation for $\mathbf{u}$ and λ is to expand the search space with $\mathbf{v}_{k+1}$. Here the Davidson method for solving eigenproblems differs from others since the method does not rely on a Krylov subspace. Instead, Davidson suggests expanding the search space to target a certain eigenpair. By setting $\mathbf{u} = \hat{\mathbf{u}}_k + \mathbf{t}$, equation (4.1) can now be written

$$\mathrm{A}(\hat{\mathbf{u}}_k + \mathbf{t}) = \lambda(\hat{\mathbf{u}}_k + \mathbf{t}). \tag{4.5}$$

Since λ is the true eigenvalue it can be expressed as $\lambda = \theta_k + \epsilon$ and equation (4.5) can be rewritten

$$\begin{aligned}\big[\mathrm{A} - (\theta_k + \epsilon)\mathrm{I}\big]\mathbf{t} &= -\big[\mathrm{A} - (\theta_k + \epsilon)\mathrm{I}\big]\hat{\mathbf{u}}_k \\ &= -\mathbf{r}_k + \epsilon\hat{\mathbf{u}}_k.\end{aligned} \tag{4.6}$$

Here $\mathbf{r}_k = \mathrm{A}\hat{\mathbf{u}}_k - \theta_k\hat{\mathbf{u}}_k$ is the residual vector at the current iteration k. At this step Davidson [5] assumes that ϵ is negligible, and also replaces A with its diagonal elements $\mathrm{D_A}$. This results in the diagonal equation

$$(\mathrm{D_A} - \theta\mathrm{I})\mathbf{t} = -\mathbf{r}. \tag{4.7}$$

$\mathbf{t}$ is then made orthogonal to $\mathcal{K}_k$ via a *modified Gram-Schmidt* process[2], normalized and then used to expand $\mathcal{K}_k$ to $\mathcal{K}_{k+1}$. The advantage of choosing $\mathrm{D_A}$ is that solving equation (4.7) becomes trivial. This explains why the Davidson method is such an efficient eigenvalue solver for diagonally dominant matrices. For a non-diagonally dominant matrix other approximations of A than the diagonal might be necessary [15]. Note also that neglecting ϵ but not replacing A would lead to stagnation, since (A - θ_k)$\mathbf{t}$ = -$\mathbf{r}_k$ has the solution $\mathbf{t} = -\hat{\mathbf{u}}_k$.

Since the Davidson algorithm is similar to the Jacobi-Davidson algorithm, the Davidson algorithm is not presented in explicit pseudo-code in this section. Instead, the differences between the two methods will be discussed in Section 4.3 when the Jacobi-Davidson algorithm is presented.

[2]A modified Gram-Schmidt process is described in Appendix B.

4.2 Jacobi Correction

In 1846 Jacobi proposed a procedure to iteratively approximate an eigenvalue of a matrix and it's corresponding eigenvector [6]. Starting with a normalized initial approximation $\hat{\mathbf{u}}$ of the eigenvector $\mathbf{u}$, the idea of the method is to separate the eigenvalue problem into a part parallel with $\hat{\mathbf{u}}$ and a part orthogonal to $\hat{\mathbf{u}}$. The approximation is then improved in each iteration by looking for a correction $\mathbf{t}$ in the orthogonal complement to $\hat{\mathbf{u}}$. The eigenvalue equation is then written as

$$\mathrm{A}(\hat{\mathbf{u}} + \mathbf{t}) = \lambda(\hat{\mathbf{u}} + \mathbf{t}), \tag{4.8}$$

along with the orthogonality condition $\hat{\mathbf{u}} \perp \mathbf{t}$. Here Jacobi seperates the problem. The part of equation (4.8) parallel to $\hat{\mathbf{u}}$ becomes

$$\hat{\mathbf{u}}^T\mathrm{A}(\hat{\mathbf{u}} + \mathbf{t}) = \lambda\hat{\mathbf{u}}^T(\hat{\mathbf{u}} + \mathbf{t}), \tag{4.9}$$

which is a scalar equation and with the orthogonality condition $\mathbf{t} \perp \hat{\mathbf{u}}$ it can be rewritten as

$$\hat{\mathbf{u}}^T\mathrm{A}\hat{\mathbf{u}} + \hat{\mathbf{u}}^T\mathrm{A}\mathbf{t} = \lambda. \tag{4.10}$$

We identify the first term in equation (4.10) as the Rayleigh quotient[3] of $\hat{\mathbf{u}}$ with regard to A and we use this as the approximation θ of the true eigenvalue λ. Next, consider the orthogonal part of equation (4.8)

$$(\mathrm{I} - \hat{\mathbf{u}}\hat{\mathbf{u}}^T)\mathrm{A}(\hat{\mathbf{u}} + \mathbf{t}) = \lambda(\mathrm{I} - \hat{\mathbf{u}}\hat{\mathbf{u}}^T)(\hat{\mathbf{u}} + \mathbf{t}). \tag{4.11}$$

Once again we use the orthogonality of $\hat{\mathbf{u}}$ and $\mathbf{t}$ and arrive at

$$(\mathrm{I} - \hat{\mathbf{u}}\hat{\mathbf{u}}^T)(\mathrm{A} - \lambda\mathrm{I})(\mathrm{I} - \hat{\mathbf{u}}\hat{\mathbf{u}}^T)\mathbf{t} = -\mathbf{r}, \tag{4.12}$$

where the right-hand side of equation (4.12) is the residual $\mathbf{r} = \mathrm{A}\hat{\mathbf{u}} - \theta\hat{\mathbf{u}}$ of the eigenvalue problem. The orthogonality condition is preserved as $\mathbf{r} \perp \hat{\mathbf{u}}$. Since λ is the unknown eigenvalue we replace it by its approximation θ.

By solving equation (4.12) for $\mathbf{t}$, we can update our eigenpair approximation by setting

$$\hat{\mathbf{u}}_{k+1} = \hat{\mathbf{u}}_k + \mathbf{t} \tag{4.13}$$

$$\theta_{k+1} = \theta_k + \hat{\mathbf{u}}_{k+1}^T\mathrm{A}\mathbf{t}. \tag{4.14}$$

This is the Jacobi Orthogonal Component Correction (JOCC) method. In short, a correction in the orthogonal complement of an approximate eigenvector is calculated and then used to update the approximation for the next iteration.

4.3 The Jacobi-Davidson Algorithm

The Jacobi-Davidson algorithm (JD) is a combination of the JOCC and the Davidson methods introduced by Sleijpen and van der Vorst [4] in 1996. In a sense, both these methods searches for a correction to some approximation $\hat{\mathbf{u}}_k$ in the subspace orthogonal to $\hat{\mathbf{u}}_k$. In the JOCC

[3] The Rayleigh quotient $\dfrac{\mathbf{v}^T\mathrm{A}\mathbf{v}}{\mathbf{v}^T\mathbf{v}}$ is an approximation of an eigenvalue from an approximative eigenvector. If $\mathbf{v}$ is an eigenvector of A, the Rayleigh quotient of $\mathbf{v}$ with regard to A is equal to the corresponding eigenvalue.

method this correction is calculated from the orthogonal projection of A onto that subspace, and in the Davidson method an expansion to a search space is calculated from the latest residual and then ortogonalized.

The idea of the JD algorithm is to implement the JOCC correction equation (4.12) in the Davidson algorithm. That is, while in the Davidson algorithm the correction equation (4.7) gives an expanding vector from

$$(\mathrm{D_A} - \theta_k \mathrm{I})\mathbf{t} = -\mathbf{r}_k, \tag{4.15}$$

the JD instead expands the search space from equation (4.12)

$$(\mathrm{I} - \hat{\mathbf{u}}_k \hat{\mathbf{u}}_k^T)(\mathrm{A} - \theta_k \mathrm{I})(\mathrm{I} - \hat{\mathbf{u}}_k \hat{\mathbf{u}}_k^T)\mathbf{t} = -\mathbf{r}_k, \;\; \hat{\mathbf{u}}_k \perp \mathbf{t}. \tag{4.16}$$

The expansion of the search space is now guided in directions orthogonal to the previous Ritz vector. Conceptually, by carefully choosing Ritz vectors from the projected matrix M, the search space is funneled in directions where the sought state potentially has large components.

4.4 Implementation

The Jacobi-Davidson algorithm follows the outline of the Davidson method. The matrix is projected onto the search space spanned by the columns of V and a Ritz pair is chosen as approximation of the sought eigenpair. Since we are working in the Berggren basis, the scalar product is defined as in equation (2.10). The notation $\mathbf{x}^T$ therefore only means transposition, and never conjugation. This of course also effects the norm of the vectors, now defined as

$$||x|| = \sqrt{\mathbf{x}^T \mathbf{x}}. \tag{4.17}$$

The key element of the Jacobi-Davidson is the expansion of the search space. The expansion is based upon a Ritz pair from the projected matrix M and the choice of Ritz pair will determine which eigenpair the algorithm will converge to. A natural approach in the JD algorithm is often to target an eigenvalue close to some value τ. However, for our purpose this is a dangerous approach. Since we are starting from a pole approximation which is an approximation of the eigenvector rather than the eigenvalue, τ would be set to the Rayleigh quotient of the initial vector. This could lead to convergence to any eigenvalue in that region since the spectrum of eigenvalues is dense around the resonance state and the Rayleigh quotient could very well be closer to some other eigenvalue. Instead, we select the Ritz pair with the largest overlap to the pole approximation. Since the pole approximation will be the first vector in our search space, the overlap of a Ritz vector is decided by the magnitude of its first element.

With this selection technique, we are now ready to state our implemented Jacobi-Davidson algorithm in pseudo-code:

Algorithm 1 Jacobi-Davidson algorithm for computing the eigenvalue of A with largest overlap with the initial guess $\hat{\mathbf{u}}_0$ and residual $< tol$.

1: $\mathbf{v}_1 = \hat{\mathbf{u}}_0/||\hat{\mathbf{u}}_0||$, $\mathrm{w}_1 = \mathrm{A}\mathbf{v}_1$, $\mathrm{V} = [\mathbf{v}_1]$, $\mathrm{W} = [\mathbf{w}_1]$
2: $\theta_1 = \mathbf{v}_1^T\mathbf{w}_1$, $\mathrm{M} = [\theta_1]$, $k = 1$
3: **loop**
4: Compute the eigenvalues of M and let $(\mathbf{s}_k, \theta_k)$ be the eigenpar with largest component in direction [1, 0, 0, ...]^T.
5: $\hat{\mathbf{u}}_k = \mathrm{V}\mathbf{s}_k$, $\mathbf{r}_k = \mathrm{W}\mathbf{s}_k$ - $\theta_k\hat{\mathbf{u}}_k$
6: **if** $||\mathbf{r}_k|| < tol$ **then**
7: **return** $(\hat{\mathbf{u}}_k, \theta_k)$
8: **end if**
9: Solve the CEQ approximately using Algorithm 2:
$(\mathrm{I} - \hat{\mathbf{u}}_k\hat{\mathbf{u}}_k^T)(\mathrm{A} - \theta_k\mathrm{I})(\mathrm{I} - \hat{\mathbf{u}}_k\hat{\mathbf{u}}_k^T)\mathbf{t} = -\mathbf{r}_k,\ \mathbf{t} \perp \hat{\mathbf{u}}_k$
10: **for** i = 1, ..., m **do**
11: $\mathbf{t} = \mathbf{t}$ - $\mathbf{v}_i^T\mathbf{t}\mathbf{v}_i$
12: **end for**
13: $\mathbf{t} = \mathbf{t}/||\mathbf{t}||$, $k = k + 1$
14: Expand V as V = [V $\mathbf{t}$]
15: Expand W as W = [W A$\mathbf{t}$]
16: Expand M to M = $\mathrm{V}^T\mathrm{W}$
17: **end loop**

As can be seen in Algorithm 1 the columns in W are simply AV. Saving W as a seperate matrix saves computational effort, since otherwise AV has to be computed on line 5 and also on line 16.

The success of the JD algorithm is highly dependent on how efficiently the correction equation (CEQ) is solved. The equation along with the orthogonality condition is stated here again for convenience:

$$(\mathrm{I} - \mathbf{u}\mathbf{u}^T)(\mathrm{A} - \theta\mathrm{I})(\mathrm{I} - \mathbf{u}\mathbf{u}^T)\mathbf{t} = -\mathbf{r},\ \ \mathbf{u} \perp \mathbf{t}. \tag{4.12}$$

This is a linear equation system of the same dimension as the original problem. An analytical solution of the CEQ is therefore computationally impossible, since the inverse of $\mathrm{A} - \theta\mathrm{I}$ is not available. Instead, the CEQ is solved approximately using the iterative algorithm Generalized Minimal Residual method (GMRES). The GMRES is an iterative Krylov algorithm described in Appendix C. However, as the eigenvalue approximation θ becomes a better approximation of the real eigenvalue, the matrix $\mathrm{A} - \theta\mathrm{I}$ becomes increasingly ill-conditioned, resulting in poor performance of the GMRES method. This issue is resolved by applying a *preconditioner* to the CEQ before attempting to solve it. After rewriting equation (4.12) as

$$\tilde{\mathrm{A}}\mathbf{t} = -\mathbf{r}, \tag{4.18}$$

where $\tilde{\mathrm{A}} = (\mathrm{I} - \mathbf{u}\mathbf{u}^T)(\mathrm{A} - \theta\mathrm{I})(\mathrm{I} - \mathbf{u}\mathbf{u}^T)$, a preconditioner $\tilde{\mathrm{K}}$ is applied from the left resulting in

$$\tilde{\mathrm{K}}^{-1}\tilde{\mathrm{A}}\mathbf{t} = -\tilde{\mathrm{K}}^{-1}\mathbf{r}. \tag{4.19}$$

The goal of the preconditioning is to make the matrix less ill-conditioned, thus improving the convergence of the GMRES. For this purpose, the preconditioner $\tilde{\mathrm{K}}$ should approximate $\tilde{\mathrm{A}}$.

The perfect preconditioner to this system is $\tilde{\mathrm{A}}$ itself, resulting in the trivial equation system $\mathbf{It} = -\tilde{\mathrm{K}}^{-1}\mathbf{r}$. However, calculating such a preconditioner is equal to solving the original problem. Hence, a good preconditioner should also be easily invertible.

In the GMRES method, the Krylov subspace basis $\{\mathbf{v}_i\}$ is generated as vectors $\mathbf{w}_k = \tilde{\mathrm{K}}^{-1}\tilde{\mathrm{A}}\mathbf{v}_k$ which are then orthogonalized through an Arnoldi iteration[4]. This approach introduces yet another issue as neither $\tilde{\mathrm{A}}$ nor $\tilde{\mathrm{K}}$ is explicitly available.

It turns out that this can circumvented due to the orthogonality condition and by choosing a preconditioner as

$$\tilde{\mathrm{K}} = (\mathrm{I} - \mathbf{u}\mathbf{u}^T)\mathrm{K}(\mathrm{I} - \mathbf{u}\mathbf{u}^T), \tag{4.20}$$

with K being an easily invertible approximation of A - θI. The next Krylov vector $\mathbf{w}$ then no longer has to be calculated from the computationally infeasible $\mathbf{w} = \tilde{\mathrm{K}}^{-1}\tilde{\mathrm{A}}\mathbf{v}_k$. Instead we write

$$\tilde{\mathrm{A}}\mathbf{v} = (\mathrm{I} - \mathbf{u}\mathbf{u}^T)(\mathrm{A} - \theta\mathrm{I})\mathbf{v} = \mathbf{y}, \tag{4.21}$$

where the right projector $(\mathrm{I} - \mathbf{u}\mathbf{u}^T)$ of equation (4.20) have been left out due orthogonality of $\mathbf{v}$ and $\mathbf{u}$. This leads us to

$$\mathbf{y} = \tilde{\mathrm{K}}\mathbf{w} = (\mathrm{I} - \mathbf{u}\mathbf{u}^T)\mathrm{K}\mathbf{w}. \tag{4.22}$$

The right projector have once again been left out, this time due to $\mathbf{w}$ being orthogonal to $\mathbf{u}$. Since the remaining projector removes all components in the direction of $\mathbf{u}$ we can rewrite equation (4.22) as

$$\mathbf{w} = \mathrm{K}^{-1}\mathbf{y} - \alpha\mathrm{K}^{-1}\mathbf{u}, \tag{4.23}$$

where the scalar $\alpha = \frac{\mathbf{u}^T\mathrm{K}^{-1}\mathbf{y}}{\mathbf{u}^T\mathrm{K}^{-1}\mathbf{u}}$. All parts of equation (4.23) are readily available, and we can now calculate new Krylov vectors without having to deal with either $\tilde{\mathrm{A}}$ or $\tilde{\mathrm{K}}$. The implementation of this specialized GMRES method is shown in Algorithm 2.

Algorithm 2 Solving the JD correction equation with left-preconditioned GMRES.

$\hat{\mathbf{r}} = \mathrm{K}^{-1}\mathbf{r}$, $\hat{\mathbf{u}} = \mathrm{K}^{-1}\mathbf{u}$, $\mu = \mathbf{u}^T\hat{\mathbf{u}}$
Compute preconditioned residual as: $\tilde{\mathbf{r}} = \hat{\mathbf{r}} - \frac{\mathbf{u}^T\hat{\mathbf{r}}}{\mu}\hat{\mathbf{u}}$
$\beta = ||\tilde{\mathbf{r}}||$, $\mathbf{v}_0 = \tilde{\mathbf{r}}/||\beta||$
for i=1, ...,m **do**
  $\mathbf{z} = (\mathrm{A} - \theta\mathrm{I})\mathbf{v}_k$
  $\hat{\mathbf{z}} = \mathrm{K}^{-1}\mathbf{z}$
  Compute new Krylov vector as: $\mathbf{w}_k = \hat{\mathbf{z}} - \frac{\mathbf{u}_k^T\hat{\mathbf{z}}}{\mu}\hat{\mathbf{u}}$
  for j=1, ...,i **do**
    $\mathrm{H}_{j,i} = \mathbf{w}_i^T\mathbf{v}_j$
    $\mathbf{w}_i = \mathbf{w}_i - \mathrm{H}_{j,i}\mathbf{v}_j$
  end for
  $\mathrm{H}_{i+1,i} = ||\mathbf{w}_i||$
  $\mathbf{v}_{i+1} = \mathbf{w}_i/\mathrm{H}_{i+1,i}$
  Expand V: $\mathrm{V} = [V\ \mathbf{v}_{i+1}]$
  Solve $\min||\beta\mathbf{e}_i - \mathrm{H}_i\mathbf{y}||$ for $\mathbf{y}$ in a least square sense
end for
return $\mathbf{t} = \mathrm{V}\mathbf{y}$

[4]This process is described in Appendix B.

The choice of K will vary for different applications. As the single-particle energies in the optomagnetical trap are significantly larger than the interaction terms, the matrix representations of the Hamiltonians studied in this thesis will always be diagonally dominant. For this reason, K is chosen to be the diagonal of A, i.e. $K \equiv D_A$.

Because of the projectional operators in $\tilde{K}$ and $\tilde{A}$, using a starting vector $\mathbf{t}_0 = 0$ in the GMRES method ensures the orthogonality of $\mathbf{u}$ and $\mathbf{t}$; the projectors eliminates all components in the direction of $\mathbf{u}$ and with a zero starting vector no components will be introduced in that direction. The specialization of the GMRES method appears in line 2, 5,6 and 7 of Algorithm 2. The minimization problem on line 15 is a least-squares problem in the Berggren basis, meaning we must take into consideration the Berggren scalar product (2.10) when solving for $\mathbf{y}$.

5

Results

In this section we present the results from various computations for finding the resonance eigenvalue using the algorithms discussed in the previous section. The main result of this thesis is the performance of the Jacobi-Davidson algorithm and the Davidson algorithm, for both of which convergence behaviours are presented in Section 5.1. The convergence behaviour is represented by the residual history of the method, i.e. the norm of the residual vector $\mathbf{r}_k$ at each iteration k until the eigenvalue is declared found. In Section 5.2 we present some calculated tunneling rates and interaction energies of the two particles, after sweeping the value of the coupling coefficient g over a large range. Generating this kind of results involves solving the eigenvalue problem many times. This was previously too computationally costly to be feasible. The goal is to illustrate some of the opportunities that arise due to the ability of the algorithms presented here to target one particular eigenvalue directly.

Lastly in Section 5.3 we also present some results from our algorithms applied to an artificial matrix. This is done in order to discuss various advantages and disadvantages in our methods.

5.1 Convergence

The residual history of the JD algorithm applied to a matrix with weak interaction strength is presented in Figure 5.1. The algorithm was applied with 2, 5 and 10 inner GMRES iterations resulting in convergence on the 16th, 10th and 6th outer JD iteration respectively. While the convergence is more consistent and faster with 10 inner iterations, one should keep in mind that each outer JD iteration becomes far more computationally costly when using more inner iterations.

Increasing the interaction strength, by increasing the magnitude of the coupling coefficient g described in Section 3.3, causes the magnitude of all off-diagonal elements to increase. This results in a less diagonally dominant matrix. The JD residual history when applied to this kind of matrix is shown in Figure 5.2. Here, a larger number of inner iterations seems even more important to the stability of the convergence, since we can see some unstable behaviour even with 10 inner iterations. It should be noted that applying such a large coupling coefficient as $g = -1.57$ does not generate any physical results since the imaginary part of the eigenvalue gets smaller than the margin of error [3].

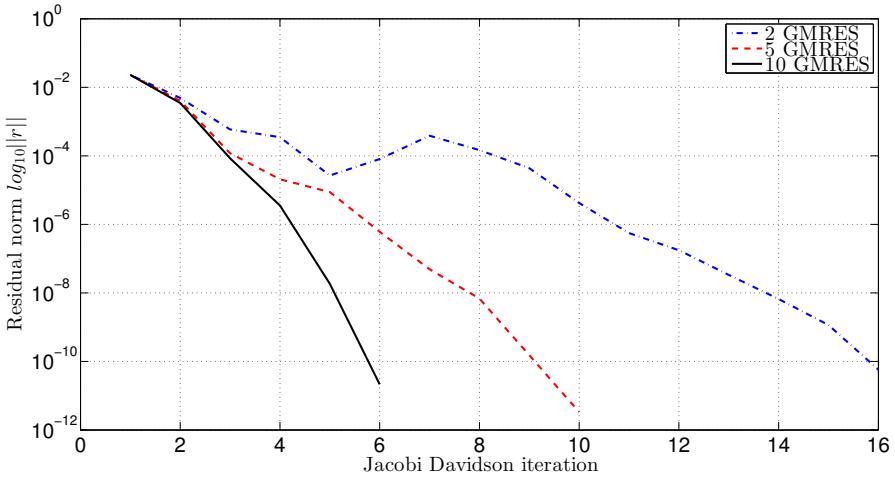

Figure 5.1: Residual history from the JD algorithm applied with 2, 5 and 10 inner iterations. The matrix has dimension 40000 and coupling coefficient $g = -0.64$.

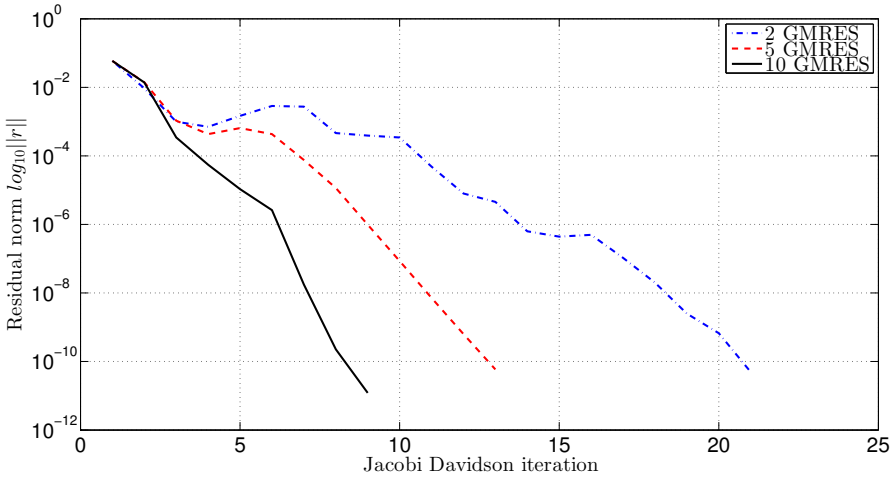

Figure 5.2: Residual history from the JD algorithm applied with 2, 5 and 10 inner iterations. The matrix is the same as above but with coupling coefficient $g = -1.57$. Note that overall the convergence is slower than in Figure 5.1.

The JD algorithm shows very fast convergence in all these cases. If the CEQ is solved with decent precision, the convergence appears to be faster than quadratical. Despite this excellent convergence, the Davidson method is faster as it requires less matrix-vector multiplications than the JD-algorithm. Figure 5.3 demonstrates the differences in the behaviour of the two algorithms. It is easy to see that the Davidson method outperforms the JD algorithm with respect to the number of total matrix-vector multiplications. The benefit of expanding the search space from the JOCC equation is not worth the computational effort required to solve the CEQ. The conclusion is that Davidson method is superior in this application.

In Figure 5.3, 10 inner iterations were used for the JD algorithm. Several attempts

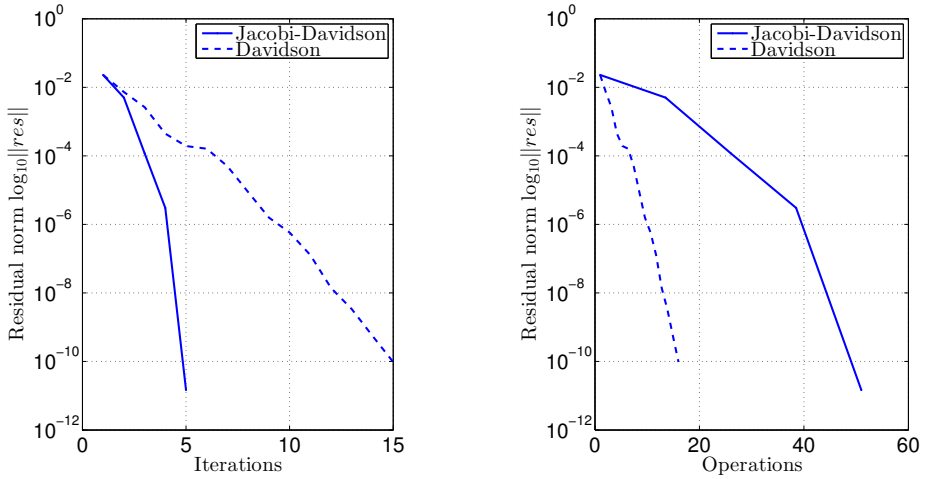

Figure 5.3: Residual histories for the JD algorithm using 10 inner iterations and the Davidson algorithm. The figure illustrates how the former converges fast per iteration, but the latter requires fewer matrix-vector multiplications.

were made for different matrices using different number of inner iterations, but the Davidson algorithm consistently converged with fewer operations. All results in the next section were therefore generated using the Davidson algorithm. However, in some cases the Davidson algorithm exhibits some unstable tendencies. This behaviour is presented in more detail in section 5.3.

5.2 Physical Results

The matrices studied are of dimensions between 25600 and 40000, corresponding to single-particle bases with 160 to 200 elements. Apart from this, the main parameter defining the system is the coupling coefficient g, which controls the interaction strength as described in Section 3.3.

With the ability to solve the eigenvalue problem with far less computational costs compared to previous methods, it is possible to study in more detail the behaviour of the model while sweeping the coupling coefficient. In figure 5.4, an example of this is presented.

The interaction energy of the two interacting particles presented in Figure 5.4 is the total energy of the resonant state minus the two single particle energies, i.e.

$$E_{int} = E_{res} - E_1 - E_2. \tag{5.1}$$

The tunneling rate is calculated according to equation (1.7).

The data presented in Figure 5.4 were generated by sweeping the coupling coefficient between -1.5 and 1.2 in steps of 0.03. Thus, the eigenvalue problem was solved 90 times, modifying the matrix in each step. Using the Davidson method this process involved roughly 1500 matrix-vector multiplications.

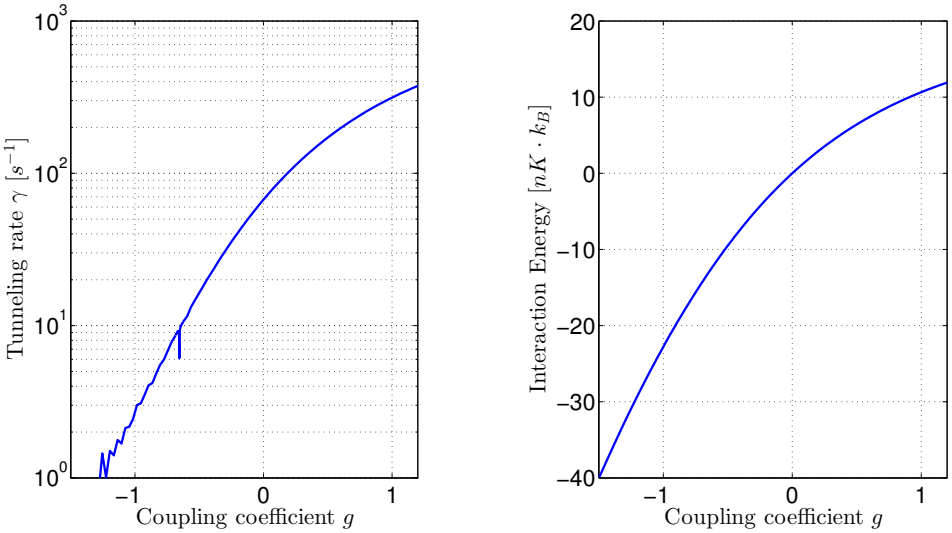

Figure 5.4: Tunneling frequency and interaction energy as a function of the dimensionless coupling coefficient g. The energy units presented are choosen as in previous results [3].

5.3 Beyond the Application

In some cases the residual history of the Davidson algorithm seemed rather volatile. Since the success of this algorithm is highly dependent on the diagonal dominance of the matrices, a matrix was modified to heavily decrease this property. While results from such a matrix have no physical significance, it poses an interesting opportunity for comparison. The JD algorithm could in principle just increase the inner iterations to counteract the smaller diagonal dominance. The Davidson algorithm has no such opportunity. In Figure 5.5 such a numerical experiment is presented.

As seen in Figure 5.5, the Davidson algorithm seems to be converging around iteration 150 but does not reach the given tolerance. This indicates that the JD algorithm has some advantages to the Davidson method in being more reliable and applicable to matrices that are less diagonally dominant. This is an important result since it shows the diversity of the JD algorithm compared to Davidsons and indicates that the JD algorithm can be used in applications where the matrix properties are less defined.

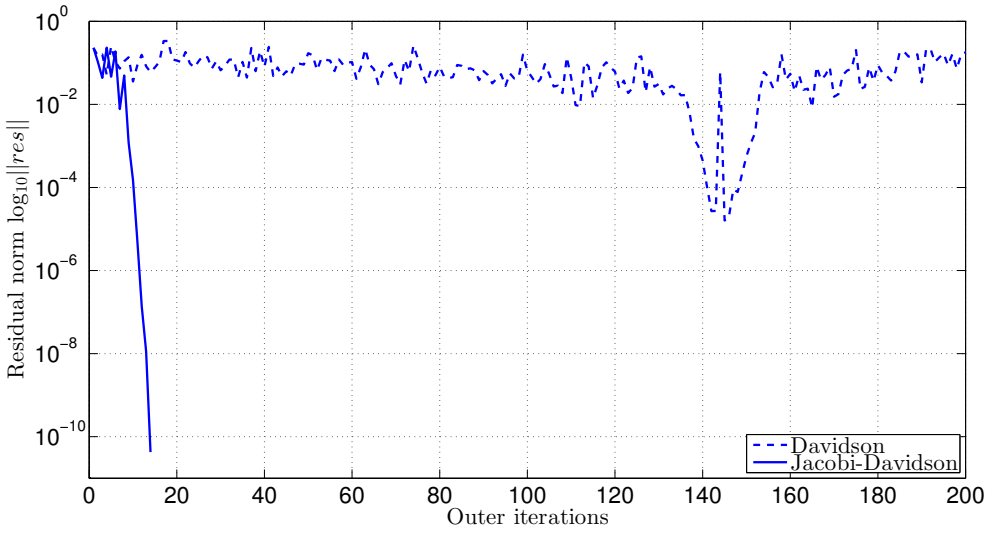

Figure 5.5: Residual histories of JD and Davidson algorithms applied to a matrix modified to decrease its diagonal dominance. The JD algorithm still converges with relatively few outer iterations, by using 50 inner iterations. The Davidson algorithm fails to converge to the desired tolerance.

6

Discussion

The Davidson method proved superior for the purpose of locating resonance eigenvalues due to the algorithm's low computational cost per outer iteration. However, the erratic behaviour of the residual before reaching convergence suggests that it is not as robust as the Jacobi-Davidson algorithm. For other applications, the JD algorithm could prove more viable. In Section 6.1, we continue to discuss some of the differences between the two algorithms. Section 6.2 explores ways of selecting the number of inner iterations in the Generalized Minimal Residual method.

In our application an essential part of both the Davidson and JD algorithms is the ability to target a specific eigenvalue in the vicinity of the pole approximation. As explained in Section 4.3 this is achieved through selection of Ritz pairs. Section 6.3 discusses some alternative selection techniques.

6.1 Advantages of the Jacobi-Davidson

In equation (4.7) in the Davidson algorithm, the matrix A has been replaced with its diagonal elements making the equation entirely diagonal. As a result, the computational cost of calculating $\mathbf{t}$ is extremely low. However, in a more general adaptation of the Davidson algorithm, where the approximation of replacing A with its diagonal elements $\mathrm{D_A}$ is too crude, and equation (4.7) would be formulated as

$$(\mathrm{K} - \theta \mathrm{I})\mathbf{t} = -\mathbf{r}, \tag{6.1}$$

where K is some other approximation of A. The equation is then no longer a diagonal equation, and if the choice of K is such that the left-hand side is not easy to invert an iterative solver would have to be used. As a result, the computational cost of this step greatly increases. Assuming equal computational effort is used in solving (6.1) as the Jacobi-Davidson correction equation, the main advantage of the Davidson method is no longer in effect. This clarifies further that the superiority of the Davidson algorithm compared to the JD algorithm that has been demonstrated in Section 5.1 is very specific to applications where D_A is a good approximation of A.

6.2 Accuracy of Correction Equation

The importance of CEQ was discussed already in Section 4.4 and the results show that with high accuracy in the CEQ solution the convergence of JD algorithm was fast in terms of the number of outer iterations. As noted above though, higher accuracy rapidly increases the computational cost of each JD-iteration, so much so that for our purposes, the Davidson method becomes more viable. Due to the size of the full matrices only matrix-vector multiplications need to be considered [1]. As can be seen from Algorithm 1 this only happens once in an outer JD-iteration (not including the CEQ), when calculating the expansion to W. The inner GMRES loop uses one matrix-vector multiplication per iteration, as can be seen in Algorithm 2 on line 4. This means the total number of operations P scales as

$$P(n) \propto \texttt{outer}(1 + \texttt{inner})n^2 + \mathcal{O}(n). \tag{6.2}$$

This gives a sense of what happens to computation times when the inner iterations are increased. The results of Section 5.1 show that too few inner iterations can lead to unstable convergence behaviour and possibly even stagnation. Thus the question arises of how to optimally choose the number of inner iterations. In Figure 5.5, and to some extent also Figures 5.1 and 5.2, we note how the JD residual fluctuates in the beginning. This could suggest that higher CEQ accuracy is needed in the early iterations, meaning that the optimal number of inner iterations could perhaps be implemented as inversely proportional to the number of the current iteration.

6.3 Selecting Ritz Pair

In the JD and Davidson algorithms (algorithm 1 line 4), one has to select a Ritz pair as an approximation to the target eigenpair. As presented by Sleijpen [4], the algorithm chooses an eigenpair closest to some value τ. In our case this is not desirable, and we therefore implement a different method where we choose the Ritz pair with largest overlap to the pole approximation.

When applying the algorithms on matrices with stronger coupling coefficient, this selection seems to have some effect on the convergence behaviour. As illustrated in Figure 6.1, in some iterations the selected Ritz pair does not seem to be the desired choice. While the algorithm always chooses the Ritz pair with largest overlap, the pair with the second largest overlap is sometimes much closer to the previous approximation (and the final result).

To bypass this problem, an alternative solution could be to use a more advanced selection method. A possible approach is selecting some Ritz pairs with large overlap with the pole approximation and amongst them choosing the one closest to the previous approximation. The results from applying such a selection to the same matrix as in the example above is presented in Figure 6.2. While the more distant guesses made in Figure 6.1 are avoided (note the difference in scale of the y-axis in the two figures), the choice still does not seem optimal. It turns out the selection process gets harder for higher iterations since the search space gets bigger. This generates more Ritz values in total and the distance between them seems to become smaller.

[1] Matrix-matrix multiplication is infeasible due to the computational costs

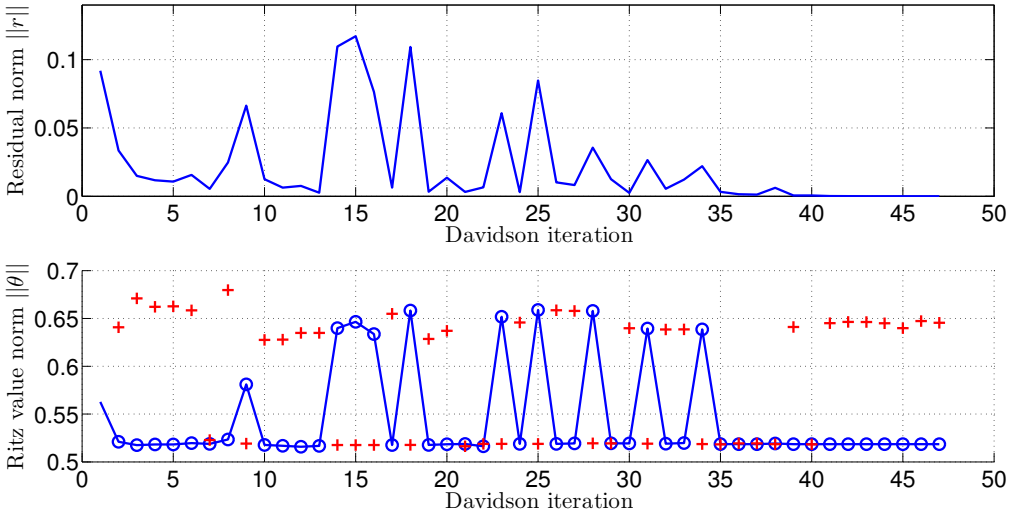

Figure 6.1: The Davidson method applied to a matrix with extremely large coupling coefficient. The marked spots represents the Ritz values with largest (circle) and second largest (cross) overlap with the pole approximation. Notice the strong correlation between a large residual and choosing a more distant Ritz pair.

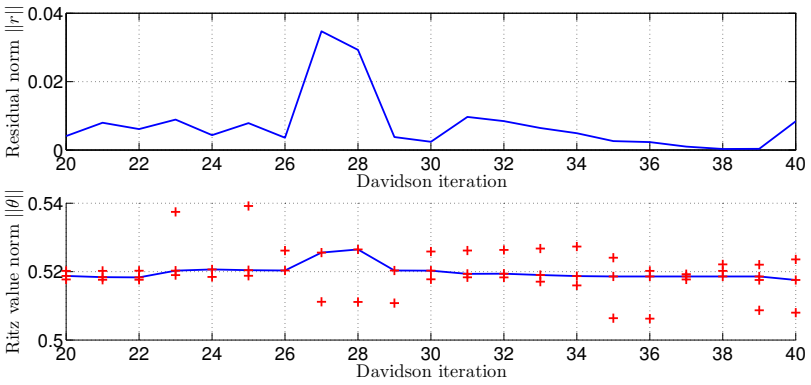

Figure 6.2: The Davidson method with a different selection rule applied to the same matrix as above. The crosses represents Ritz values with large overlap to the pole approximation. Here, 4 Ritz values where selected and then the one closest to the previous approximation were used.

Although we can not see any particular advantage of using such a selection in our application (apart from a slightly smoother convergence), it might be useful for other applications of the JD and Davidson algorithms.

Something else that might be useful in other applications of the JD or Davidson algorithms is the method of *harmonic Ritz values* presented by Sleijpen [4]. Since this method did not seem compatible with our implemented selection method it was not implemented in our case.

6.4 Conclusions and Outlook

The Jacobi-Davidson algorithm proved highly effective for locating the resonance eigenvalues in this specific application. Even with rather limited accuracy in solving the correction equation, the convergence is almost quadratical. The fast convergence combined with the ability to directly target the resonance eigenvalue allows the Jacobi-Davidson algorithm to greatly decrease the computation time for locating resonances compared to a Krylov subspace solver. Calculations previously requiring several hours were completed in only a few minutes. It turns out, however, that in this application, the Davidson method alone is sufficient and the very low computational cost of the algorithm makes it more effective for our purpose, with some further reduction to computation times. Both the Davidson algorithm and the Jacobi-Davidson algorithm only require one copy of the matrix in the memory which mimimizes memory demands.

These algorithms allowed for a detailed calculation of how the tunneling rates are related to the interaction strengths. They could also help produce more accurate results since using more basis states, and hence larger Hamiltonians, allows more accurate models, and the size of the matrices has little impact on the behaviour of the algorithms.

While the Davidson method was better suited for this specific application, the Jacobi-Davidson algorithm could prove very useful for several other applications. This thesis was mainly concerned with a system of two particles but in a many body system the Hamiltonian becomes sparse and various incomplete decompositions could be used as preconditioners. This could allow the Jacobi-Davidson to take over where the Davidson method is no longer effective.

Bibliography

[1] G. Zürn, *Few-fermion systems in one dimension*, Ph.D. thesis, Heidelberg University, Ruperto Carola (2012).

[2] G. Zürn, A. N. Wenz, S. Murmann, A. Bergschneider, T. Lompe, S. Jochim, *Pairing in Few-Fermion Systems with Attractive Interactions*, Phys. Rev. Lett. 111 (2013) 175–302.

[3] R. Lundmark, *Tunneling Theory for Few-Body Systems in One-Dimensional Traps*, Master's thesis, Chalmers University of Technology, Sweden (2014).

[4] G. L. Sleijpen, H. A. van der Vorst, *A Jacobi–Davidson iteration method for linear eigenvalue problems*, SIAM Review 42 (2) (2000) 267–293.

[5] E. R. Davidson, *The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real symmetric matrices*, J. Comput. Phys 17 (10) (1975) 87–94.

[6] C. G. J. Jacobi, *Über ein leichtes Verfahren die in der Theorie der Säcularstörungen vorkommenden Gleichungen numerisch aufzulösen*, Journal für die reine und angewandte Mathematik 51–94.

[7] P. Arbenz, M. Hochstenbach, *A Jacobi–Davidson Method for Solving Complex Symmetric Eigenvalue Problems*, SIAM Journal on Scientific Computing 25 (5) (2004) 1655–1673.

[8] J. J. Sakurai, J. Napolitano, *Modern Quantum Mechanics*, Addison-Wesley, 2011.

[9] T. Berggren, *On the use of resonant states in eigenfunction expansions of scattering and reaction amplitudes*, Nuclear Physics A 109 (10) (1985) 265–287.

[10] M. T. Heath, *Scientific Computing: An Introductory Survey*, 2nd Edition, McGraw-Hill Higher Education, 1996.

[11] W. H. Dickhoff, D. van Neck, *Many-Body Theory Exposed!*, World Scientific Publishing, 2005.

[12] MATLAB, version 8.1.0.604 (R2013a), The MathWorks Inc., Natick, Massachusetts, 2013.

[13] E. Romero, J. E. Roman, *A Parallel Implementation of Davidson Methods for Large-scale Eigenvalue Problems in SLEPc*, ACM Trans. Math. Softw. 40 (2) (2014) 13:1–13:29.

[14] B. P. Kürschner, *Two-sided Eigenvalue Algorithms for Modal Approximation* Master theisis Max planck institute for dynamics of complex technical systems.

[15] R. B. Morgan, *Davidson's method and preconditioning for generalized eigenvalue problems*, Journal of Computational Physics 89 (1) (1990) 241 – 245.

[16] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, H. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems*, `http://web.eecs.utk.edu/~dongarra/etemplates/book.html` (2000).

[17] Y. Saad, M. H. Schultz, *GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems*, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 856–869.

Glossary

CEQ The Correction Equation. Used for expanding the search space in the Jacobi-Davidson method, derived in Section 4.2.

Coupling coefficient A dimensionless parameter g proportional to the inter-particle interaction in the modeled system. For more information see [3].

GMRES An iterative method of solving a linear equation. The GMRES method is described in Appendix C and implemented in Algorithm 2.

Inner iteration A GMRES iteration inside the Jacobi-Davidson solver..

JOCC The Jacobi Orthogonal Component Correction is an iterative method for calculating eigenpairs presented in Section 4.2.

Outer iteration A full Jacobi-Davidson or Davidson iteration as described in Algorithm 1.

Pole approximation The quantum mechanical state where both particles are in their individual resonance state. This is described in Section 3.4.

Residual history The norm of the residual vector $\mathbf{r}$ for each step in an iterative eigensolving method.

Search space The subspace in which the eigensolving method is approximating the target eigenvector.

A

The Krylov Subspace

In linear algebra, a Krylov subspace $\mathcal{K}_k$ of order k, generated from the matrix A, is spanned by the images of a vector $\mathbf{b}$ under the first $k-1$ powers of A.

$$\mathcal{K}_k(\mathrm{A},\mathbf{b}) = \mathrm{span}\{\mathbf{b}, \mathrm{A}^1\mathbf{b}, \mathrm{A}^2\mathbf{b}, ... \mathrm{A}^{k-1}\mathbf{b}\} \tag{A.1}$$

The main advantages of using this kind of subspace comes from the Cayley–Hamilton theorem, which states that the inverse of a matrix can be expressed as a sum of its powers. This makes the subspace useful in iterative methods for dealing with linear equation systems or eigenvalue problems. Another useful property when generating the subspace is that it avoids matrix-matrix multiplication by first calculating $\mathbf{v}_1 = \mathrm{A}\mathbf{b}$, followed by $\mathbf{v}_2 = \mathrm{A}\mathbf{v}_1$ and so on. This property is essential when dealing with very large matrices when matrix-matrix operations become very costly.

Some successful Krylov subspace methods for finding eigenvalues include the Arnoldi, Lanczos and Conjugate Gradient methods [16].

B

Arnoldi Iteration

While the vectors obtained when expanding the Krylov subspace are useful, they tend to become almost lineary dependant for large k. To avoid this, most methods involving Krylov subspaces generates an orthogonal set of vectors $\mathbf{q}_0$, $\mathbf{q}_1$, $\mathbf{q}_2$, ... spanning this subspace. This can be done with Arnoldi iteration, which uses a modified Gram-Schmidt process to remove any projections of the previously generated vectors $\mathbf{q}_i$, $0 \leqslant i \leqslant k-1$ from the new vector $\mathbf{v}_k$ when generating the Krylow subspace. The projections obtained in the process are stored in an (k+1)-by-(k) upper Hessenberg matrix

$$H_k = \begin{bmatrix} \mathbf{q}_0^* \mathbf{v}_1 & \mathbf{q}_0^* \mathbf{v}_2 & \cdots & \mathbf{q}_0^* \mathbf{v}_k \\ \sqrt{\mathbf{v}_1^* \mathbf{v}_1} & \mathbf{q}_1^* \mathbf{v}_2 & \cdots & \mathbf{q}_1^* \mathbf{v}_k \\ 0 & \sqrt{\mathbf{v}_2^* \mathbf{v}_2} & \cdots & \mathbf{q}_2^* \mathbf{v}_k \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{q}_{k-1}^* \mathbf{v}_k \\ 0 & 0 & \cdots & \sqrt{\mathbf{v}_k^* \mathbf{v}_k} \end{bmatrix}. \tag{B.1}$$

This matrix combined with the matrix $Q_k = [\mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_k]$ has the property

$$\mathrm{AQ}_k = \mathrm{Q}_{k+1} \mathrm{H}_k. \tag{B.2}$$

The Arnoldi iteration algoritm is described in detail in Algorithm 3 below.

Algorithm 3 Construct and orthogonalize a Krylov subspace, span{$\mathbf{q}_1$, $\mathbf{q}_2$ ...}, and the upper Hessenberg matrix H with respect to A. The diagonalization process is a modified Gram-Schmidt and takes place on line 3 to 6.

1: **for** k = 2, 3, ... **do**
2: $\quad \mathbf{q}_k = \mathrm{A}\mathbf{q}_{k-1}$
3: $\quad$ **for** i = 1 to k-1 **do**
4: $\quad\quad h_{i,k-1} = \mathbf{q}_i{}^*\mathbf{q}_k$
5: $\quad\quad \mathbf{q}_k = \mathbf{q}_k$ - $h_{j,k-1}\mathbf{q}_i$
6: $\quad$ **end for**
7: $\quad h_{k,k-1} = \|\mathbf{q}_k\|$
8: $\quad \mathbf{q}_k = \frac{\mathbf{q}_k}{h_{k,k-1}}$
9: **end for**

C

Generalized Minimal Residual Method

The Generalized Minimal Residual (GMRES) method is a method based on Krylov subspaces for iteratively solving linear equation systems and is commonly used when the matrix involved is not symmetric, e.g. non-hermitian [17].

Denote the linear equation system to be solved by GMRES

$$\mathrm{A}\mathbf{x} = \mathbf{b}, \tag{C.1}$$

where A is a non-singular $n \times n$ matrix. The first vector $\mathbf{v}_1$ spanning the Krylov subspace $\mathcal{K}_1$ is the residual when applying $\mathbf{x}_0 = 0$ as solution to the system,

$$\mathbf{v}_1 = \mathbf{r} = \mathrm{A}\mathbf{x}_0 - \mathbf{b} = \mathbf{b}, \tag{C.2}$$

building a Krylov subspace as

$$\mathcal{K}_k(\mathrm{A},\mathbf{b}) = \mathrm{span}\{\mathbf{b}, \mathrm{A}^1\mathbf{b}, \mathrm{A}^2\mathbf{b}, ... \mathrm{A}^{k-1}\mathbf{b}\} = \mathrm{span}\{\mathbf{v}_1, \mathbf{v}_2, ...\mathbf{v}_k\}. \tag{C.3}$$

In each iterative step k the algorithm expands the Krylov search space to $\mathcal{K}_{k+1}$, orthogonalizes this space through the Arnoldi iteration described in appendix B and sets up an orthonormal basis for the search space. These basis vectors are then appended as columns to the $n \times k$ matrix V_k. An approximation of the solution $\mathbf{x}$ projected onto $\mathcal{K}_k$ can now written as $\hat{\mathbf{x}} = \mathrm{V}_k\mathbf{y}$, $\mathbf{y}$ being a vector containing the coefficients for for each base vector in V_k.

With $\hat{\mathbf{x}}$ on this form, consider the residual norm of the system, $||\mathbf{b} - \mathrm{A}\hat{\mathbf{x}}||$. Using the property of the Arnoldi iteration in equation (B.2), this can be rewritten as

$$||\mathbf{b} - \mathrm{A}\hat{\mathbf{x}}|| = ||\mathbf{b} - \mathrm{A}\mathrm{V}_k\mathbf{y}|| = ||\beta\mathbf{v}_1 - \mathrm{V}_{k+1}\mathrm{H}_k\mathbf{y}|| = ||\mathrm{V}_{k+1}(\beta\mathbf{e}_1 - \mathrm{H}_k\mathbf{y})||, \tag{C.4}$$

where β is the norm of $\mathbf{b}$ and $\mathbf{e}_1 = [1, 0, 0,...]^T$. Thus, minimizing the residual is eqvivalent to minimizing $||\beta\mathbf{e}_1 - \mathrm{H}_k\mathbf{y}||$, which will be a least-squares problem of dimension $k + 1$ and therefore easy to solve for $\mathbf{y}$. As the search space V is then expanded the approximate solution $\hat{\mathbf{x}}$ will converge to $\mathbf{x}$.