Electrostatic and Electrodynamical Manipulation of Electrons on Cryogenic Substrates for Quantum Computing Applications

Master Thesis
Group D
Physics
Aalborg University
Submitted: 30-05-2025

TITLE SHEET

AALBORG UNIVERSITY
DENMARK

Final Year Physics
Department of Materials and Production
Aalborg 9220

Title: Electrostatic and Electrodynamical Manipulation of Electrons on Cryogenic Substrates for Quantum Computing Applications
Project: Master Thesis

Project Period: September 2024 - May 2025
Project Group: Group D
Participants: Jonatan Tøffner-Clausen
Supervisor: Thomas Garm Pedersen
Number of Pages: 114

Abstract:

The purpose of this thesis was to study aspects of the out-of-plane qubit control induced by external electric fields using quantum mechanical perturbation theory in both the time-dependent and time-independent cases for the electron-on-Helium qubit platform. By limiting ourselves to simpler models, we are able to apply both analytical and numerical approaches, which show the viability and ease of which qubits may be electrically controlled on this platform. We initially derive an expression for the image-charge induced Coulomb potential binding the electron using classical electrostatics. The energies and eigenstates of the electron are then found by solving the time-independent Schrödinger equation under the given potential. The corresponding dipole matrix elements and oscillator strengths are subsequently found. The model is then generalized in the form a Kratzer potential using a quantum defect parameter. The generalized energies, eigenstates, dipole matrix elements and oscillator strengths are then determined. The system is then perturbed by a static electric field and time-independent perturbation theory is used to find energy and eigenstate corrections with both the basis expansion method and Dalgarno-Lewis approach. The first few energy corrections and polarizabilities are subsequently found for a general n. The perturbative energy expansion is regularized using a hypergeometric approximant and its accuracy is verified with a numerical complex-scaled Sturmian expansion scheme. The model is compared with experimental results. A new harmonic field is then introduced as the perturbation and time-dependent perturbation theory is used to determine the eigenstate corrections with both the basis expansion method and a Sturmian expansion. The dynamic linear polarizability for Kratzer potential is found in closed form and plotted. The closed form Pockels polarizability for the Coulomb potential is shown.

Preface

This Master thesis has been carried out by a $9'$th/$10'$th semester physics student at the Department of Materials and Production, Aalborg University, in the period starting in September 2024 and ending in May 2025. The thesis was written under supervision of Thomas Garm Pedersen. The sources of the main text are referenced using the apalike formatting method and are found in the bibliography in alphabetical order. In accordance with AAU rules, generative A.I has been used for this project as a programming and grammar assistant at certain points.

I would like to thank Thomas for supervising this project. I would also like to thank Mikkel and Ali for helpful discussion and feedback.

Jonatan Tøffner-Clausen
jtaffn20@student.aau.dk

List of Figures

Contents

Chapter 1

Introduction

Quantum computing is a framework wherein quantum mechanical systems are used to perform computational tasks. Quantum computers aim to exploit certain quantum mechanical phenomena such as superposition and entanglement for the purposes of computation. The information stored on quantum computers is encoded in quantum bits, or qubits, rather than bits. As opposed to classical bits, qubits can be in a superposition of both zero and one and may be entangled with other qubits so that the state of one is correlated with the state of another [de Wolf, 2023]. Utilization of these properties in turn allows for the creation of quantum algorithms that offer significant computational advantages and speed-up for certain tasks in comparison with classical computer algorithms [Zhahir et al., 2023, de Wolf, 2023]. This includes the simulation of quantum systems, factorization of large numbers, searching in unstructured data, solving linear equations and more, which see applications in cryptography, finance, machine learning, physics research and other fields [Zhahir et al., 2023]. The field of quantum computation originally began in the early 1980s with the introduction of the first framework for a reversible quantum Turing machine [Benioff, 1980]. Richard Feynmann shortly thereafter proposed using these quantum computers to simulate quantum mechanical systems as a means of avoiding the exponential computational costs associated with classical computers [Feynman, 1982]. The viability of quantum computers for such tasks was subsequently confirmed and extended to more general systems [Lloyd, 1996, Deutsch, 1985]. The first quantum algorithms were then developed [Deutsch and Jozsa, 1992, Simon, 1997], which later inspired Shor's algorithm in 1994, which finds the prime factorization of an integer [Shor, 1997]. The first two-qubit quantum computer was built in 1997 and succesfully ran the algorithm reported by Deutsch and Jozsa [Chuang et al., 1998]. Later, in 2001, a five-qubit quantum computer was used to find the prime factorization of 15 using Shor's algorithm [Vandersypen et al., 2001]. Since then, interest and research in quantum computing have continued to grow. In 2025, IBM is planning to introduce the "Kookaburra" chip featuring 1.386 qubits [IBM Quantum, 2023].

The practical challenges involved in realizing a real-world large-scale quantum computer are very considerable [de Wolf, 2023, Dykman and Platzman, 2000]. A fully realized quantum computer requires an appreciable number of qubits, which can all interact with one another and yet not couple to other objects external to the system [Dykman and Platzman, 2000]. A computation in such a system involves manipulating the individual electrons, which includes changing their states and mutual interaction, on a time-scale less than their expected coherence time. Once a computation is completed, we must then be able to perform a read-out of the system so that the result of said computation may be extracted [Dykman and Platzman, 2000]. One of the most prominent problems facing a practical quantum computer is decoherence of the quantum states on which the qubits are encoded [Gill et al., 2025]. This occurs when the qubits inadvertently interact with the environment and causes the encoded information to be lost. This problem may be mitigated in several ways such as environmental control, like cooling, or quantum error-correction codes, but the practical implementation is still in its infancy [de Wolf, 2023]. Another significant problem is finding ways of effectively controlling and entangling qubits, and doing it on a scale where enough qubits are present to allow for meaningful computational capacity [Gill et al., 2025]. Numerous potential qubit platforms have been proposed as a result of the research conducted in response to these challenges including trapped-ion qubits [Bernardini et al., 2023], spin qubits [Harvey, 2022], superconducting qubits [Huang et al., 2020], photonic qubits [Romero and Milburn, 2024] and others, each of which come with their own set of strengths and weaknesses. The present work will focus on a particular type of these qubit platforms, namely the Electron-on-Helium qubit.

The Electron-on-Helium qubit was originally proposed by Platzman and Dykman in 1999 [Dykman and Platzman, 2000]. This platform proposes using electrons bound to a cryogenic substrate of Helium as qubits. The originally proposed setup involves placing a cryogenic substrate of Helium on a plate capacitor. The electron qubits would then be confined and controlled using small electrodes embedded in the Helium substrate [Dykman and Platzman, 2000]. The electron qubits' out-of-plane confinement would primarily be determined by the relative difference of permitivity between the substrate and the medium, usually a vacuum, which induces an image charge effect that binds the electron relatively weakly to the substrate. The negative electron affinity of the substrate also induces a finite potential barrier at the interface of roughly one electronvolt [Jennings et al., 2024], which in turn quantizes the out-of-plane states of the electrons. A possible read out mechanism could then be state-selective ionization induced by an electric field provided by the capacitor plates. The ground state and first excited state were originally proposed as the two qubit basis states. It has since been extended to Neon substrates [Chen et al., 2022]. The physics of such systems have been studied already in the 1960s [Bruschi et al., 1966, Cole and Cohen, 1969], but was only proposed for quantum computing purposes by the end of the millennium. Subsequent research of this platform has since been made [Dykman et al., 2003, Lyon, 2006, Kawakami et al., 2023]. The electron-on-Helium qubit platform faces certain challenges such as mechanical disturbances [Jennings et al., 2024], but also has several promising features including long coherence times, efficient qubit control and scalability [Dykman and Platzman, 2000].

The main focus of this thesis will be to study the electron-on-Helium qubit platform with particular emphasis on the out-of-plane qubit control using electric fields. In particular, we ultimately aim to describe aspects of the electron qubits' response due to both static and dynamic perturbations induced by external electric fields. We do this by first examining the details of the potential induced by the image charge effect and then determining the unperturbed out-of-plane electron states. The resulting theoretical framework is then subsequently extended using quantum defect theory as a means of accurately modelling experimental observations that are not immediately reproduced by the bare image charge effect. We then introduce external electric fields and study the induced effects, including Stark shifts and electric polarizability, using non-degenerate quantum mechanical perturbation theory for both the static and dynamic case. By restricting ourselves to single electron qubit systems and using simpler models for the out-of-plane states, we can use analytical and exact approaches to model the effects of perturbations induced by external electric fields. The results from perturbation theory are subsequently extended to non-perturbative regimes such as field-induced ionization processes. These approaches allow us to find out-of-plane electron states with energy eigenvalues agreeing with experimental findings, analytical Stark shifts and ionization rates and closed-form electric, frequency-dependent polarizabilities. The results from this work is compared with numerical methods and experimental findings. By doing so we aim to demonstrate the viability and ease for which qubits may be electrically controlled on this platform.

Chapter 2

Electron Qubit on Cryogenic Substrate

In this chapter, we construct the electron qubit system from the perspective of classical electromagnetism and derive the image potential binding the electron to the substrate. We then construct and solve the time-independent Schrödinger equation pertaining to the electron qubit system, which includes finding the energy eigenvalues and eigenstates in both the bound and unbound case. In addition, we find the diagonal and off-diagonal dipole matrix elements of the system, as well as the corresponding oscillator strengths and oscillator strength moments. We then discuss the energy transitions, reintroduce the electron as a quantum defect atom and thereby construct a phenomenological extension of the image potential. With the new potential, we solve the time-independent Schrödinger equation once more and determine the corresponding matrix elements, oscillator strength and oscillator strength moments.

2.1 One-dimensional Electron on Cryogenic Substrate

In this section, we initially wish to examine some basic properties of the electron-on-Helium qubit platform, which includes developing a simple one-dimensional model for the electron qubits floating on the cryogenic substrate composed of liquid Helium or solid Neon. This will allow us to determine the out-of-plane electron states yielding correct energy eigenvalues and ultimately prepare us for the later analysis concerning qubit control using external electric fields. The basic qubit platform is shown in Figure 2.1 below, where an electron of charge $q = -e$ is placed at a position $z_0\hat{\mathbf{z}}$ and lies in a medium with dielectric constant ε_1. The origin is placed in the boundary plane between the medium occupied by the electron qubit and the cryogenic substrate, which is treated as a separate medium with dielectric constant ε_2. In this way, the entire system is composed of two semi-infinite dielectric media separated by an interface lying in the xy-plane. The qubit platform has been proposed with both liquid Helium and solid Neon substrates, which have dielectric constants $\varepsilon_{\text{He}} = 1.057$ [Lide, 2005] and $\varepsilon_{\text{Ne}} = 1.244$ [Zhou et al., 2022], respectively. The medium occupied by the electron is usually chosen to be a vacuum so that $\varepsilon_1 = 1$.

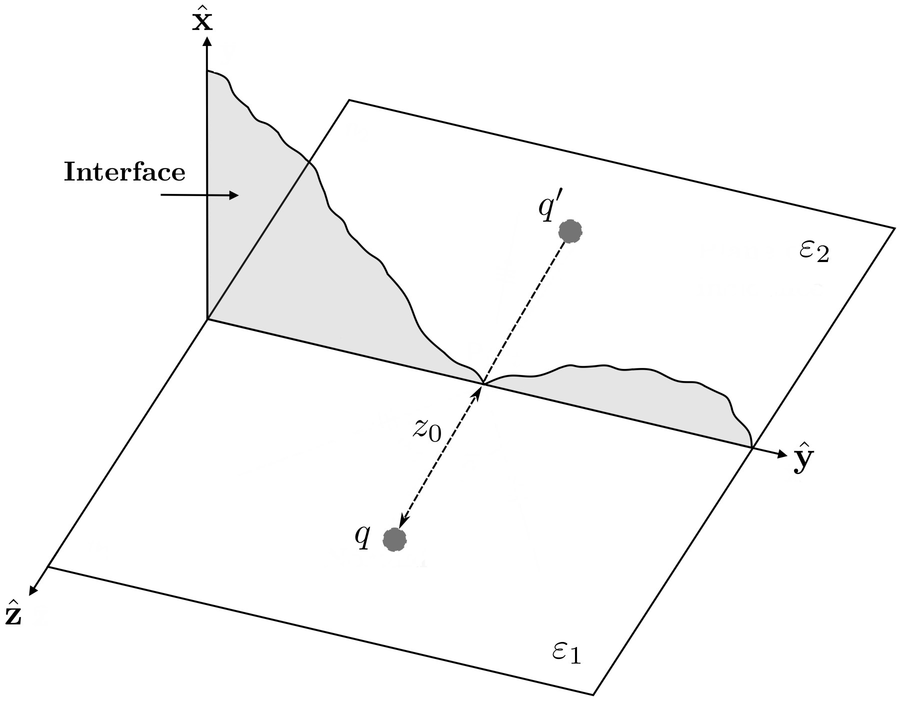

Figure 2.1: Sketch of Electron Lying Above Cryogenic Substrate. The electron lies in medium 1, which in the present case is composed of a vacuum and hence $\varepsilon_1 = 1$. The cryogenic substrate corresponds to the second medium of permitivity ε_2. Important examples of cryogenic substrates for the purposes of qubit platforms are liquid Helium and solid Neon, which have dielectric constants $\varepsilon_2 = 1.057$ and $\varepsilon_2 = 1.244$, respectively. The electron induces an image charge in the substrate due to the relative difference in permitivity. The electron and its image both lie a distance z_0 from the interface separating the two media, which in this case is the xy-plane. The figure is based on [Pedrotti et al., 2018].

The main idea now is that the electron induces an image charge q' placed at $-z_0\hat{\mathbf{z}}$ in the cryogenic substrate due to the relative difference in permitivity between the two media [Landau and Lifshitz, 1984]. The interaction between the electron qubit and its image in the cryogenic substrate in turn produces a Coulomb-like potential $\mathcal{V}$ that binds the electron qubit relatively weakly to the surface of the substrate [Dykman and Platzman, 2000]. In addition to the Coulomb-like potential, the electron also faces a finite potential barrier at the interface between the two media due to the negative electron affinity of the substrate [Jennings et al., 2024]. The magnitude of this barrier for a liquid Helium substrate is roughly $V_0 \simeq 1$ eV [Dykman and Platzman, 2000]. In the case of a solid Neon substrate, this barrier is around 30% lower [Zhou et al., 2022]. We may then write the potential confining the qubit electron to the substrate as

$$\mathcal{V}(z) = \begin{cases} V_0, & \text{if } z \leq 0 \\ V(z), & \text{if } z > 0 \end{cases}, \tag{2.1.1}$$

where V_0 is the magnitude of the potential barrier and $V(z)$ is the potential induced by interaction with the image charge. For the purposes of determining the electron qubits energy eigenstates, we will on account of the relatively large potential barrier assume that $V_0 \to \infty$ and hence restrict the problem to the positive z-axis [Jennings et al., 2024]. This corresponds to imposing a Dirichlet boundary condition and hence ignoring potential spill-out effects into the substrate. This ultimately allows us to determine the eigenstates and energy eigenvalues of the electron qubit by solving a problem similar to the one-dimensional Hydrogen atom. Before doing so, we must first determine the form of the potential $V(z)$ that binds the electron qubit to the substrate. We accomplish this using a classical electrodynamics approach described in [Landau and Lifshitz, 1984]. This involves determining the magnitude of the image charge q' using the method of images, which ensures that the two point charges satisfy the electromagnetic boundary conditions. In the first medium, we seek the electric potential ϕ_1 due to the two point charges q and q', whereas in the second medium we seek the electric potential ϕ_2 due to another fictitious charge q'' lying at $z_0\hat{\mathbf{z}}$. The total electric potential ϕ in the system may then be expressed as

$$\phi = \frac{1}{4\pi\varepsilon_0} \times \begin{cases} q''/\varepsilon_2 r, & \text{if } z \leq 0 \\ q/\varepsilon_1 r + q'/\varepsilon_1 r', & \text{if } z > 0 \end{cases}, \tag{2.1.2}$$

where r and r' are the distances from some point of observation to the charge and its image, respectively. We may now determine the two fictitious charges q' and q'' by ensuring that ϕ satisfies the electromagnetic boundary conditions, which in this case require that both the parallel component of the electric field $\mathcal{E}_\parallel$ and the perpendicular component of the displacement field $\mathcal{D}_\perp = \varepsilon\mathcal{E}_\perp$ are continuous across the interface shown in Figure 2.1. We can relate the electric field to the electric potential by $\mathcal{E} = -\nabla\phi$. In terms of the electric potential therefore, the boundary conditions read

$$\text{BCs} : \begin{cases} \phi_1 = \phi_2 \\ \varepsilon_1 \partial_z \phi_1 = \varepsilon_2 \partial_z \phi_2 \end{cases}. \tag{2.1.3}$$

We can now substitute the two electric potentials ϕ_1 and ϕ_2 from equation (2.1.2) into the electromagnetic boundary conditions given in equation (2.1.2), which gives

$$\left.\frac{q'}{4\pi\varepsilon_0\varepsilon_1\sqrt{x^2+y^2+(z+z_0)^2}}\right|_{z=0} = \left.\frac{1}{4\pi\varepsilon_0\sqrt{x^2+y^2+(z-z_0)^2}}\left(\frac{q''}{\varepsilon_2} - \frac{q}{\varepsilon_1}\right)\right|_{z=0}, \tag{2.1.4}$$

$$\left.\frac{-2\varepsilon_1 q'(z+z_0)}{8\pi\varepsilon_0\varepsilon_1\sqrt{x^2+y^2+(z+z_0)^2}}\right|_{z=0} = \left.\frac{-2(z-z_0)}{8\pi\varepsilon_0\sqrt{x^2+y^2+(z-z_0)^2}}\left(\frac{q''\varepsilon_2}{\varepsilon_2} - \frac{q\varepsilon_1}{\varepsilon_1}\right)\right|_{z=0}, \tag{2.1.5}$$

where we have evaluated both electric potentials at an arbitrary point along the interface, meaning $z = 0$. Substituting $z = 0$ then reduces equations (2.1.4) and (2.1.5) significantly and just leaves

$$\varepsilon_2(q + q') = \varepsilon_1 q'', \tag{2.1.6}$$

$$q - q' = q''. \tag{2.1.7}$$

The magnitude of the two fictitious charges q' and q'' can now be determined by solving the system given in equation (2.1.6) and (2.1.7). For the purposes of this calculation, we are only interested in the image charge q'. Solving the system above shows that

$$q' = \frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2} q. \tag{2.1.8}$$

We have now determined the magnitude of the image charge q', which in the present case of Helium and Neon are relatively weak. Specifically, we have $q' = -0.0277q$ and $q' = -0.1087q$ for Helium and Neon, respectively. Using equation (2.1.8), we are now able to determine the potential $V(z)$ by first calculating the image force $\mathbf{F}$ and then exploiting the fact that it is conservative. The standard Coulombic image force becomes

$$\begin{aligned} \mathbf{F} &= \frac{q^2}{4\pi\varepsilon_0\varepsilon_1(2z)^2}\frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2}\hat{\mathbf{z}} \\ &= \frac{q^2}{16\pi\varepsilon_0\varepsilon_1 z^2}\frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2}\hat{\mathbf{z}}. \end{aligned} \tag{2.1.9}$$

Because the image force is conservative, we know that $\mathbf{F} = -\nabla V$ [Kibble and Berkshire, 2004]. Because the image force acts solely along $\hat{\mathbf{z}}$, we can determine $V(z)$ by simple integration, which gives

$$\begin{aligned} V &= -\int \frac{q^2}{16\pi\varepsilon_0\varepsilon_1 z^2}\frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2}\mathrm{d}z \\ &= -\frac{q^2}{16\pi\varepsilon_0\varepsilon_1 z}\frac{\varepsilon_2 - \varepsilon_1}{\varepsilon_1 + \varepsilon_2}. \end{aligned} \tag{2.1.10}$$

We now substitute $\varepsilon_1 = 1$ and define $\varepsilon_2 \equiv \varepsilon$, where ε is the permitivity of the chosen substrate. This allows the potential in equation (2.1.10) to be written as

$$V(z) = -\frac{\mathcal{Z}e^2}{4\pi\varepsilon_0 z}, \quad \text{where } \mathcal{Z} = \frac{1}{4}\frac{\varepsilon - 1}{\varepsilon + 1}. \tag{2.1.11}$$

For the particular cases of Helium and Neon, we find that $\mathcal{Z}_{\mathrm{He}} = 0.006928$ and $\mathcal{Z}_{\mathrm{Ne}} = 0.027184$. The potential that binds the electron to the substrate is therefore quite weak, especially in comparison with the potential of the Hydrogen atom, which is two orders of magnitude stronger. With the potential of the electron qubit determined and the corresponding boundary conditions imposed, we are now ready to determine the out-of-plane energy eigenstates and corresponding energy eigenvalues. This is accomplished by solving the time-independent Schrödinger equation under the imposed one-dimensional potential, which reads

$$\hat{H}\psi = E\psi, \quad \text{where } \hat{H} = -\frac{\hbar^2}{2m}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{\mathcal{Z}e^2}{4\pi\varepsilon_0 z}, \tag{2.1.12}$$

where E is the energy eigenvalue and m is the mass of the electron. The eigenvalue problem in equation (2.1.12) is as mentioned above restricted to the positive z-axis due to the infinite potential barrier approximation at the interface. We want to solve equation (2.1.12) for both the bound states and unbound states. We start with the bound states $\psi = \psi_n$, which have a set of discrete energy levels $E = E_n < 0$ for each principal quantum number n. Before proceeding, we convert the problem at hand to modified atomic units. In particular, we define the modified Bohr radius and Hartree energy in terms of

$$\overline{a}_0 = \frac{4\pi\varepsilon_0\hbar^2}{me^2\mathcal{Z}}, \quad E_n = -\frac{\overline{\mathrm{Ha}}}{2n^2} = \frac{\hbar^2}{2m\overline{a}_0^2 n^2}, \tag{2.1.13}$$

which we will be our characteristic length and energy scales, respectively. In the presented units, we in the case of Helium have $\overline{a}_0 = 7.609$ nm and $\overline{\mathrm{Ha}} = 1.316$ meV $= 318.2$ GHz, and in the case of Neon have $\overline{a}_0 = 1.939$ nm and $E_0 = 20.267$ meV $= 4.901$ THz. We then note that the more realistic finite barrier $V_0 \simeq 1$ eV for the Helium substrate is approximately 1500 times larger than the ground state energy as provided by equation (2.1.13). Similarly for the Neon substrate, the finite barrier is roughly 70 times larger. Using the units outlined in equation (2.1.13), the eigenvalue problem reads

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z} - \frac{1}{2n^2}\right)\psi_n = 0. \tag{2.1.14}$$

The eigenvalue problem has now been converted appropriate units and we are ready to proceed. We can determine the eigenstates ψ_n by assuming that they are products of a number of simpler functions [Nieto, 2000]. These simpler functions are found by investigating by investigating the limiting behaviour of z. We start by considering the asymptotic behaviour of ψ_n as $z \to \infty$. In this limit, one finds that equation (2.1.14) reduces to

$$\left(\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{1}{n^2}\right)\psi_n \to 0, \quad \text{for } z \to \infty. \tag{2.1.15}$$

It is clear that equation (2.1.15) is asymptotic to both $\exp(z/n)$ and $\exp(-z/n)$. We will later require that $\langle\psi_n|\psi_m\rangle = \delta_{nm}$, where δ_{nm} is the Kronecker delta, and hence we discard $\exp(z/n)$ because it is not square-integrable on $\mathbb{R}_{\geq 0}$. Thus, we can then conclude that

$$\psi_n \sim \mathrm{e}^{-z/n}. \tag{2.1.16}$$

We now consider the opposite limiting behaviour equation (2.1.14), where instead $z \to 0$. In this limit, we find that the third term $-1/2n^2$ vanishes and hence

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z}\right)\psi_n \to 0, \quad \text{for } z \to 0. \tag{2.1.17}$$

As with the three-dimensional Hydrogen atom, we see that a factor of the eigenstates ψ_n is a power function z_n^p for some p. Substituting this factor into equation (2.1.17) above then shows that $p(p-1) = 0$ in the small z_n limit, from which we conclude that z is a factor. We now write the eigenstates as $\psi_n = fz\exp(-z/n)$, where $f = f(z)$ is an as of yet unknown function. Substituting this proposal into equation (2.1.14) then shows that

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z} - \frac{1}{2n^2}\right) fz\mathrm{e}^{-z/n} = 0. \tag{2.1.18}$$

We now expand the above equation by letting the differential operator act on the proposed solution, which in turn yields

$$\left(\frac{1}{2}z\frac{\mathrm{d}^2 f}{\mathrm{d}z^2} + \frac{\mathrm{d}f}{\mathrm{d}z} - \frac{z}{n}\frac{\mathrm{d}f}{\mathrm{d}z} + \frac{zf}{2n^2} - \frac{f}{n} + f - \frac{fz}{2n^2}\right)\mathrm{e}^{-z/n} = 0. \tag{2.1.19}$$

It is clear that the fourth and last terms in equation (2.1.19) cancel. Reducing further and multiplying through by $n\exp(z/n)$ then gives

$$z'\frac{\mathrm{d}^2 f}{\mathrm{d}z'^2} + (2-z')\frac{\mathrm{d}f}{\mathrm{d}z'} + (n-1)f = 0, \tag{2.1.20}$$

where we in addition have made the change of variable $z' = 2z/n$. The condition given in equation (2.1.20) requires that f is an associated Laguerre polynomial, as shown in equation (A.0.2). In this particular case, we have $a = 1$ and $b = n-1$, which in turn means that $f = L_{n-1}^{(1)}(2z/n)$. We can now formulate the complete eigenfunction by combining all three factors, which in turn yields

$$\psi_n = \mathcal{N}_n z\mathrm{e}^{-z/n}{}_1F_1\left(1-n; 2, \frac{2z}{n}\right), \tag{2.1.21}$$

where we have introduced the normalization constant $\mathcal{N}_n$ and applied equation (A.0.6) to express the eigenstates in terms of the confluent hypergeometric function. We note that the normalization constant has absorbed a factor n from equation (A.0.6). We now only need to determine $\mathcal{N}_n$, which we accomplish by enforcing the $m = n$ case of the orthonormalization condition, meaning $\langle\psi_n|\psi_n\rangle = 1$. Applying this to the wave function found above in equation (2.1.21) yields

$$\int_0^\infty \mathcal{N}_n^2 z^2\mathrm{e}^{-2z/n}\left[{}_1F_1\left(1-n; 2, \frac{2z}{n}\right)\right]^2 \mathrm{d}z = 1. \tag{2.1.22}$$

To evaluate the integral above, we first change the variable of integration so that $z' = 2z/n$, which in turn means that $\mathrm{d}z = n\mathrm{d}z'/2$ and hence

$$\frac{1}{8}n^3\mathcal{N}_n^2\int_0^\infty z'^2\mathrm{e}^{-z'}\left[{}_1F_1\left(1-n; 2, z'\right)\right]^2 \mathrm{d}z' = 1. \tag{2.1.23}$$

The integral in equation (2.1.23) is a special case of the W. Gordon integral shown in equation (A.0.10); $\mathrm{J}_2^{1,0}\left(1-n, 1-n; 1, 1, 1\right)$ in particular. Rearranging equation (2.1.23) above and applying the identity then shows that

$$\begin{aligned}
\mathcal{N}_n &= \left\{\frac{1}{8}n^3 \mathrm{J}_2^{1,0}\left(1-n, 1-n; 1, 1, 1\right)\right\}^{-1/2} \\
&= \left\{\frac{1}{8}n^3 \frac{\Gamma(3)(n-1)!}{(2)_{n-1}} {}_3F_2(1-n, -1, 2, 2, 1, 1)\right\}^{-1/2} \\
&= \left\{\frac{1}{8}n^3 \frac{2\Gamma(n)}{\Gamma(n+1)} \cdot n\right\}^{-1/2} \\
&= 2n^{-3/2}, \qquad (2.1.24)
\end{aligned}$$

where we applied the identity $\Gamma(x+1) = x\Gamma(x)$. Using the expression for the normalization constant $\mathcal{N}$ given in equation (2.1.24) then shows that the fully normalized eigenstate are

$$\psi_n = 2n^{-3/2} z \mathrm{e}^{-z/n} {}_1F_1\left(1-n; 2, \frac{2z}{n}\right). \qquad (2.1.25)$$

The bound eigenstates given above are products of an n'th order polynomial and a decaying exponential. Now that we have determined the normalized eigenstates, we will be able to calculate the probability density using $|\psi_n|^2 = \psi_n^\star \psi_n$, from which expectation values may be found. The wave function of the ground state and first three excited states are shown in Figure 2.2 below.

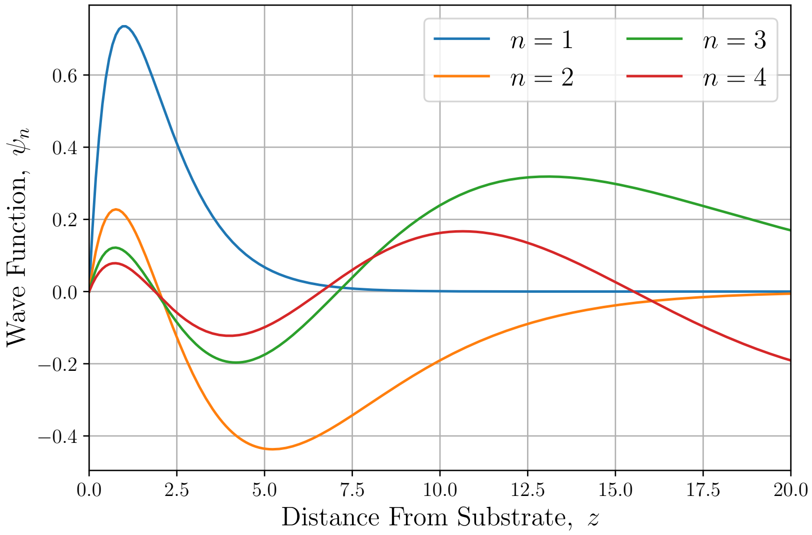

Figure 2.2: Wave Function of the bound electron plotted as a function of distance z from the interface, which lies in the xy-plane. The eigenstates corresponding to principal quantum numbers $n \in \{1, 2, 3, 4\}$ are shown. The eigenstates asymptotically settle to zero at infinity. In the case of a Helium substrate, we have $20\overline{a}_0 = 152.18$ nm, whilst in the case of a Neon substrate, we have $20\overline{a}_0 = 38.78$ nm.

Because the dielectric constant of Helium is close to that of air, then it is clear that $\mathcal{Z} \ll 1$ and hence the characteristic length $\overline{a}_0$ is relatively large. For this reason one finds that the probability density first vanishes for relatively large distance, especially for the excited states. As seen in Figure 2.2, the ground state decays at roughly $z = 7.5$, which in the case of Helium and Neon corresponds to $z = 57.1$ nm and $z = 14.5$ nm, respectively. Now that we have determined the bound eigenstates and their corresponding eigenvalues, we turn to the unbound eigenstates $\psi = \psi_k$. The unbound eigenstates form a continuum in the wavenumber $k > 0$ rather than the discrete set given in the bound case. The corresponding energy levels are $E = E_k > 0$. Like the bound eigenstates, we determine the unbound eigenstates by first converting the eigenvalue problem to atomic units using equation (2.1.13) and

$$E_k = \overline{\text{Ha}}\frac{k^2}{2} = \frac{\hbar^2 k^2}{2m\overline{a}_0^2}. \tag{2.1.26}$$

Using these units, the corresponding eigenvalue problem given in equation (2.1.12) then reads

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z} + \frac{1}{2}k^2\right)\psi_k = 0. \tag{2.1.27}$$

As with the bound eigenstates, we can now determine the unbound eigenstates by assuming that they may be written as a product of several other functions. The same line of reasoning, which we used in the previous case concerning the bound eigenstates, can be applied for the unbounded eigenstates to deduce that $\psi_k = fz\exp(-ikz)$, where $f = f(k)$ is an unknown function. The reason for this is that the unbound eigenstates, in terms of the governing differential equation, inhibit the same behaviour in both the $z \to 0$ and $z \to \infty$ limits . Although both $\exp(\pm ikz)$ are valid in this case, we choose the negative sign for the sake of consistency. Inserting the proposed solution into equation (2.1.27) above and expanding terms then gives

$$\left(\frac{1}{2}z\frac{\mathrm{d}^2 f}{\mathrm{d}z^2} - ikz\frac{\mathrm{d}f}{\mathrm{d}z} + \frac{\mathrm{d}f}{\mathrm{d}z} - \frac{1}{2}k^2 zf - ikf + f + \frac{1}{2}k^2 fz\right)\mathrm{e}^{-ikz} = 0. \tag{2.1.28}$$

Once again, the fourth and last terms cancel. Multiplying the resulting equation through by a factor $\exp(ikz)/ik$ then shows that

$$z'\frac{\mathrm{d}^2 f}{\mathrm{d}z'^2} + (2 - z')\frac{\mathrm{d}f}{\mathrm{d}z'} + \left(\frac{1}{ik} - 1\right)f = 0, \tag{2.1.29}$$

where we also have introduced the change of variable $z' = 2ikz$. This shows that f is an associated Laguerre polynomial, for which $a = 1$ and $b = -1 - i/k$ such that $f = L^{(1)}_{-1-i/k}(2ikz)$. Combining all three factors then allows the unbound eigenstates to be written as

$$\psi_k = \mathcal{N}_k z\mathrm{e}^{-ikz}{}_1F_1\left(1 + \frac{i}{k}, 2, 2ikz\right), \tag{2.1.30}$$

where the conversion factor from equation (A.0.6) has been absorbed into the normalization constant $\mathcal{N}_k$. It is important to note that the unbound eigenstates are not square-integrable and hence we cannot enforce the orthonormalization condition $\langle\psi_k|\psi_{k'}\rangle = \delta_{kk'}$ to find $\mathcal{N}_k$. Instead, we require that

$$\int_0^\infty \psi_k^\star \psi_{k'}\mathrm{d}z = \delta(k - k'), \tag{2.1.31}$$

where $\delta(k - k')$ is the Dirac delta function [Bethe and Salpeter, 1957]. Eigenstates satisfying equation (2.1.31) are said to be normalized on the k-scale or δ-normalized. Determining $\mathcal{N}_k$ so that this condition holds is a more involved calculation compared to the regular normalization procedure. An outline of how it is accomplished is described in the appendix, see section B. The resulting calculation shows that

$$\mathcal{N}_k = \frac{2k^{1/2}}{\sqrt{1 - \mathrm{e}^{-2\pi/k}}}. \tag{2.1.32}$$

Accordingly, the δ-normalized eigenstates may be written as

$$\psi_k = \frac{2k^{1/2}}{\sqrt{1 - \mathrm{e}^{-2\pi/k}}} z\mathrm{e}^{-ikz}{}_1F_1\left(1 + \frac{i}{k}, 2, 2ikz\right). \tag{2.1.33}$$

The hypergeometric series in the unbound eigenstates no longer truncates because the first argument is not a negative integer. The unbound eigenstates therefore consist of a hypergeometric series and a complex exponential, that asymptotically approaches the wave function of a free particle. Similarly to the bound states, using the eigenstates given in equation (2.1.33) now allows us to compute the corresponding probability density $|\psi_k| = \psi_k^\star \psi_k$ for the unbound states. The unbound wave functions with wavenumbers $k = 1/4$, $k = 1/8$ and $k = 1/12$ are shown in Figure 2.3 below.

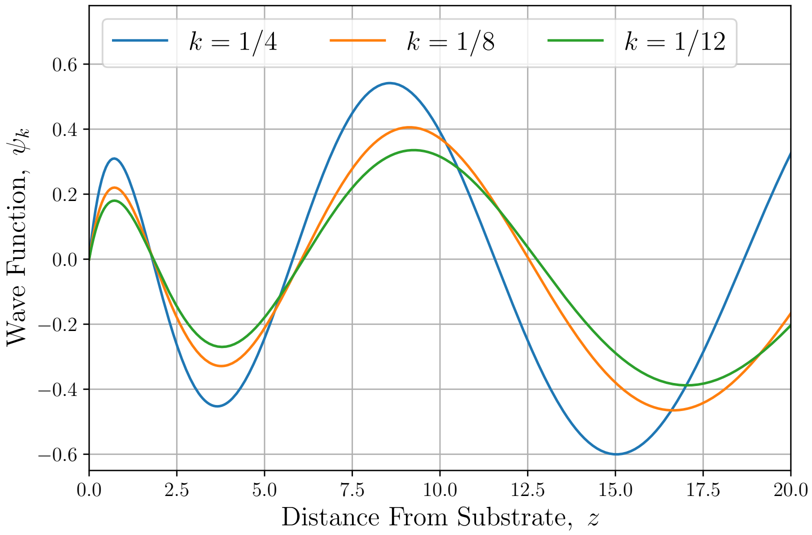

Figure 2.3: Wave function of the unbound electron plotted as a function of distance from the interface, which lies in the xy-plane. The eigenstates corresponding to wavenumbers $k \in \{1/4, 1/8, 1/12\}$ are shown. The eigenstates asymptotically settle to free particle waves of unit amplitude. This takes significantly longer for the smaller wavenumber eigenstates.

As a consequence of the unbound eigenstates not being normalizable, we find that the probability density never decays to zero, but rather asymptotically approaches the wave function of a free particle. This is shown for three particular eigenstates in Figure 2.3 above, where the wave function simply oscillates. The original eigenvalue problem given in equation (2.1.12) has now been solved for both the bound and unbound states. In the following section, we turn to calculating the expectation values and matrix elements, which will also be put to use later in the project.

2.1.1 Qubit Electron Matrix Elements & Oscillator Strengths

Now that both the bound and unbound eigenstates of the electron-on-Helium qubit system have been found, we are able to apply the results and determine the dipole matrix elements of the described system. The resulting calculations provides both the expectation value $Z_{nn} = \langle \psi_n | z | \psi_n \rangle$ of the position coordinate z and the more general off-diagonal matrix element $Z_{nm} = \langle \psi_n | z | \psi_m \rangle$ that quantifies the strength of an electric dipole transition under an electric field [Sakurai and Napolitano, 2017], both of which will useful later. There are in general three types of matrix elements, namely the bound-bound, bound-unbound and unbound-unbound cases. The unbound-unbound elements are difficult to calculate and will not be presented in this thesis. An attempt to determine them using the methods presented in [Gordon, 1929, Ji et al., 2024] was tried, but later abandoned due to the scope of the thesis. We start with the bound-bound matrix elements $Z_{nm} = \langle \psi_n | z | \psi_m \rangle$ pertaining to the position operator z. In the case of $m = n$, we retrieve the diagonal matrix elements corresponding to the expectation value of the positional coordinate z. Using the bound electron's eigenstates as given in equation (2.1.25), one finds that a direct calculation of the inner product yields

$$\begin{aligned} Z_{nn} &= 4n^{-3}\int_0^\infty z^2 z\mathrm{e}^{-2z/n}\left[{}_1F_1\left(1-n;2,\frac{2z}{n}\right)\right]^2\mathrm{d}z \\ &= \frac{1}{4}n\int_0^\infty z'^3\mathrm{e}^{-z'}\left[{}_1F_1\left(1-n;2,z'\right)\right]^2\mathrm{d}z', \end{aligned} \tag{2.1.34}$$

where we have used the change of variable of integration such that $z' = 2z/n$. The integral given in equation (2.1.34) is a special case of the W. Gordon integral, the details of which is described in the appendix, see section A. To evaluate it, we make use of equation (A.0.15) given in the appendix. Comparing to the aforementioned formula, we then find that

$$\begin{aligned} Z_{nn} &= \frac{1}{4}n\mathrm{J}_2^{2,0}\left(1-n,1-n;1,1,1\right) \\ &= \frac{1}{4}n\frac{\Gamma(4)(n-1)!}{(2)_{n-1}}{}_3F_2(1-n,-2,3,2,1,1) \\ &= \frac{3n\Gamma(n)}{2\Gamma(n+1)}\cdot n^2 \\ &= \frac{3}{2}n^2, \end{aligned} \tag{2.1.35}$$

which agrees with the standard result of the Hydrogen atom in the $\ell = 0$ case [Shankar, 1994]. We now move onto the off-diagonal matrix elements Z_{nm}, for which $m \neq n$. A direct calculation initially shows that

$$\begin{aligned} Z_{nm} &= 4(nm)^{-3/2}\int_0^\infty z\mathrm{e}^{-z/n}{}_1F_1\left(1-n;2,\frac{2z}{n}\right)z z\mathrm{e}^{-z/m}{}_1F_1\left(1-m;2,\frac{2z}{m}\right)\mathrm{d}z \\ &= \frac{n^{5/2}}{4m^{3/2}}\int_0^\infty z'^3\mathrm{e}^{-\frac{n+m}{2m}z'}{}_1F_1\left(1-n;2,z'\right){}_1F_1\left(1-m;2,\frac{n}{m}z'\right)\mathrm{d}z', \end{aligned} \tag{2.1.36}$$

where we have changed the variable of integration to z'. The integral found in equation (2.1.36) is another example of the W. Gordon integral. This allows us to write the off-diagonal matrix elements as

$$Z_{nm} = \frac{n^{5/2}}{4m^{3/2}}\mathrm{J}_2^{2,0}\left(1-n,1-m,1,\frac{n}{m}\right). \tag{2.1.37}$$

To evaluate this integral, we make use of the recurrence relation given in equation (A.0.14). This requires us to compute a $\mathrm{J}_2^{1,0}$ integral and two $\mathrm{J}_2^{0,0}$ integrals. For the given integral, we first note that the parameter

$$\begin{aligned} \mathcal{A} &= \frac{1}{2}\cdot 2\left(1-\frac{n}{m}\right)-(1-n)+\frac{n}{m}(1-m) \\ &= 0. \end{aligned} \tag{2.1.38}$$

It is then clear that the integral $\mathrm{J}_2^{1,0}$ vanishes and has no contribution to the matrix element. Substituting this into equation (A.0.14) then leaves

$$\begin{aligned} Z_{nm} &= \frac{n^{5/2}}{4m^{3/2}}\frac{4}{1-(n/m)^2}\left\{(2-2(1-m))\,\mathrm{J}_2^{0,0}\left(1-n,1-m,1,\frac{n}{m}\right)+2(1-m)\mathrm{J}_2^{0,0}\left(1-n,2-m,1,\frac{n}{m}\right)\right\} \\ &= \frac{2n^{5/2}m^{1/2}}{m^2-n^2}\left\{m\mathrm{J}_2^{0,0}\left(1-n,1-m,1,\frac{n}{m}\right)+(1-m)\mathrm{J}_2^{0,0}\left(1-n,2-m,1,\frac{n}{m}\right)\right\}. \end{aligned} \tag{2.1.39}$$

We now use equation (A.0.10) to compute the $\mathrm{J}_2^{0,0}$ integrals. The first of these, for which the second argument is $1-m$, becomes

$$\begin{aligned} \mathrm{J}_2^{0,0}\left(1-n,1-m,1,\frac{n}{m}\right) &= \frac{2^2\Gamma(2)(1+n/m)^{1-n+1-m-2}}{(n/m-1)^{1-n}(1-n/m)^{1-m}}{}_2F_1\left(1-n,1-m,2,\frac{-4n/m}{(n/m-1)^2}\right) \\ &= \frac{4(-1)^{n-1}m^2}{(m-n)^2}\left(\frac{m-n}{m+n}\right)^{n+m}{}_2F_1\left(1-n,1-m,2,\frac{-4nm}{(n-m)^2}\right). \end{aligned} \tag{2.1.40}$$

In identical fashion, the second integral becomes

$$\mathrm{J}_2^{0,0}\left(1-n,2-m,1,\frac{n}{m}\right)=\frac{4(-1)^{n-1}m^2}{(m-n)^2}\left(\frac{m-n}{m+n}\right)^{m+n-1}{}_2F_1\left(1-n,2-m,2,\frac{-4nm}{(n-m)^2}\right). \tag{2.1.41}$$

Substitution both integrals back into equation (2.1.39) then yields

$$\begin{aligned}Z_{nm}=\frac{2n^{5/2}m^{1/2}}{m^2-n^2}\Bigg\{&m\frac{4(-1)^{n-1}m^2}{(m-n)^2}\left(\frac{m-n}{m+n}\right)^{n+m}{}_2F_1\left(1-n,1-m,2,\frac{-4nm}{(n-m)^2}\right)\\&+(1-m)\frac{4(-1)^{n-1}m^2}{(m-n)^2}\left(\frac{m-n}{m+n}\right)^{m+n-1}{}_2F_1\left(1-n,2-m,2,\frac{-4nm}{(n-m)^2}\right)\Bigg\}.\end{aligned} \tag{2.1.42}$$

We can now collect similar terms in equation (2.1.42) and finally express the off-diagonal matrix elements as

$$\begin{aligned}Z_{nm}=&\frac{8(-1)^n(nm)^{5/2}}{(m-n)^4}\left(\frac{m-n}{m+n}\right)^{m+n}\\&\times\left\{m\frac{n-m}{m+n}{}_2F_1\left(1-n,1-m,2,\frac{-4mn}{(m-n)^2}\right)+(m-1){}_2F_1\left(1-n,2-m,2,\frac{-4mn}{(m-n)^2}\right)\right\}.\end{aligned} \tag{2.1.43}$$

We now have obtained all the matrix elements for the bound states of the electron. The matrix elements Z_{nm} are plotted for particular n, m in Figure 2.3 below.

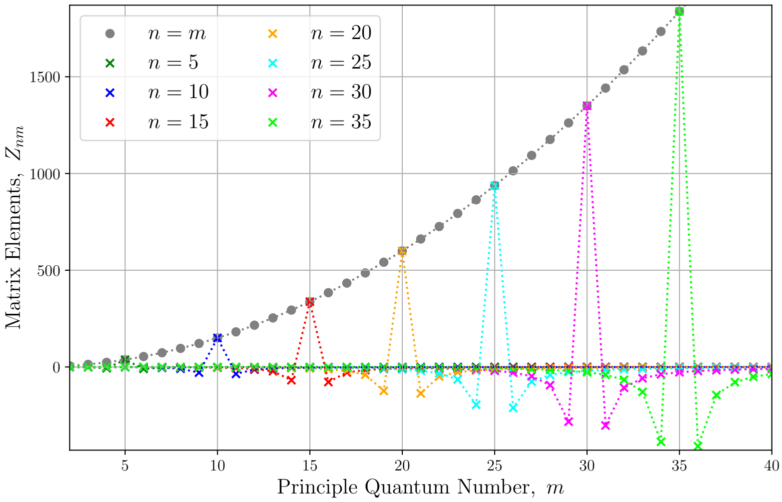

Figure 2.4: The bound-bound matrix elements Z_{nm} for fixed $n \in \{5, 10, 15, 20, 25, 30, 35\}$ are plotted for every $1 \leq m \leq 40$. In addition, the diagonal matrix elements $Z_{nn} = 3n^2/2$ are plotted alongside. The dotted lines are meant as a visual aid to see the pattern and not represent any matrix elements because they are of course discrete. The off-diagonal matrix elements spike around $n = m$ and decays as m recedes.

As seen in Figure 2.3, the off-diagonal matrix elements attain non-zero values only within a short interval around the diagonal $n = m$ case, which grow quadratically with n. The same general pattern holds for all the off-diagonal matrix elements, which grow in step with the expectation value of z. With the bound-bound matrix elements calculated, we now move on to the bound-unbound matrix elements Z_{nk}.

Using the unbound states found in equation (2.1.33), the immediate calculation gives

$$\begin{aligned}
Z_{km} &= \frac{4m^{-3/2}k^{1/2}}{\sqrt{1-\mathrm{e}^{-2\pi/k}}}\int_0^\infty z\mathrm{e}^{ikz}{}_1F_1\left(1-\frac{i}{k},2,-2ikz\right)zz\mathrm{e}^{-z/m}{}_1F_1\left(1-m;2,\frac{2z}{m}\right)\mathrm{d}z \\
&= \frac{m^{5/2}k^{1/2}}{4\sqrt{1-\mathrm{e}^{-2\pi/k}}}\int_0^\infty z'^3\mathrm{e}^{-(1-ikm)z'/2}{}_1F_1\left(1-\frac{i}{k},2,-ikmz'\right){}_1F_1\left(1-m;2,z'\right)\mathrm{d}z' \\
&= \frac{m^{5/2}k^{1/2}}{4\sqrt{1-\mathrm{e}^{-2\pi/k}}}\mathrm{J}_2^{2,0}\left(1-\frac{i}{k},1-m,-ikm,1\right), \qquad (2.1.44)
\end{aligned}$$

where we have introduced $z' = 2z/m$ and introduced the W. Gordon integral representation. We now apply the same procedure as with the bound-bound matrix elements. We note that $\mathscr{A} = 0$ once more and hence we can evaluate the integral using equation (A.0.14), where the $\mathrm{J}_2^{1,0}$ integral vanishes. This leaves

$$\begin{aligned}
Z_{km} &= \frac{m^{5/2}k^{1/2}}{4\sqrt{1-\mathrm{e}^{-2\pi/k}}}\frac{4}{(-ikm)^2-1}\left\{(2-2(1-m))\,\mathrm{J}_2^{0,0}\left(1-\frac{i}{k},1-m,-ikm,1\right)\right. \\
&\quad\left.+2(1-m)\mathrm{J}_2^{0,0}\left(1-\frac{i}{k},2-m,-ikm,1\right)\right\} \\
&= \frac{-2m^{5/2}k^{1/2}}{(1+k^2m^2)\sqrt{1-\mathrm{e}^{-2\pi/k}}}\left\{m\mathrm{J}_2^{0,0}\left(1-\frac{i}{k},1-m,-ikm,1\right)\right. \\
&\quad\left.+(1-m)\mathrm{J}_2^{0,0}\left(1-\frac{i}{k},2-m,-ikm,1\right)\right\}. \qquad (2.1.45)
\end{aligned}$$

We now calculate the two $\mathrm{J}_2^{0,0}$ integrals using equation (A.0.10). For the first integral with second argument $1-m$, we acquire

$$\begin{aligned}
\mathrm{J}_2^{0,0}\left(1-\frac{i}{k},1-m,-ikm,1\right) &= \frac{2^2\Gamma(2)(-ikm+1)^{1-i/k+1-m-2}}{(1-(-ikm))^{1-i/k}(-ikm-1)^{1-m}}{}_2F_1\left(1-\frac{i}{k},1-m,2,\frac{-4-ikm}{(1-(-ikm))^2}\right) \\
&= \frac{4(-1)^{m-1}}{(1-ikm)^2}\left(\frac{1-ikm}{1+ikm}\right)^{2-i/k-m}{}_2F_1\left(1-\frac{i}{k},1-m,2,\frac{4ikm}{(1+ikm)^2}\right),
\end{aligned} \qquad (2.1.46)$$

whilst for the second integral with second argument $2-m$, we get

$$\mathrm{J}_2^{0,0}\left(1-\frac{i}{k},2-m,-ikm,1\right) = \frac{4(-1)^m}{(1-ikm)^2}\left(\frac{1-ikm}{1+ikm}\right)^{3-i/k-m}{}_2F_1\left(1-\frac{i}{k},2-m,2,\frac{4ikm}{(1+ikm)^2}\right). \qquad (2.1.47)$$

Substituting both integrals back into equation (2.1.45) then yields

$$\begin{aligned}
Z_{km} &= \frac{-2m^{5/2}k^{1/2}}{(1+k^2m^2)\sqrt{1-\mathrm{e}^{-2\pi/k}}}\left\{m\frac{4(-1)^{m-1}}{(1-ikm)^2}\left(\frac{1-ikm}{1+ikm}\right)^{2-i/k-m}{}_2F_1\left(1-\frac{i}{k},1-m,2,\frac{4ikm}{(1+ikm)^2}\right)\right. \\
&\quad\left.+(1-m)\frac{4(-1)^m}{(1-ikm)^2}\left(\frac{1-ikm}{1+ikm}\right)^{3-i/k-m}{}_2F_1\left(1-\frac{i}{k},2-m,2,\frac{4ikm}{(1+ikm)^2}\right)\right\}. \qquad (2.1.48)
\end{aligned}$$

Reducing now shows that

$$\begin{aligned}
Z_{km} &= \frac{8(-1)^m m^{5/2}k^{1/2}}{(1+ikm)^4\sqrt{1-\mathrm{e}^{-2\pi/k}}}\left(\frac{1-ikm}{1+ikm}\right)^{-i/k-m} \qquad (2.1.49) \\
&\quad\times\left\{m\frac{1+ikm}{1-ikm}{}_2F_1\left(1-\frac{i}{k},1-m,2,\frac{4ikm}{(1+ikm)^2}\right)+(m-1){}_2F_1\left(1-\frac{i}{k},2-m,2,\frac{4ikm}{(1+ikm)^2}\right)\right\},
\end{aligned}$$

which is the final expression of the unbound-bound matrix elements. As with the bound-bound matrix elements, we now also plot the bound-unbound matrix elements. This is shown for particular k, m in Figure 2.3 below.

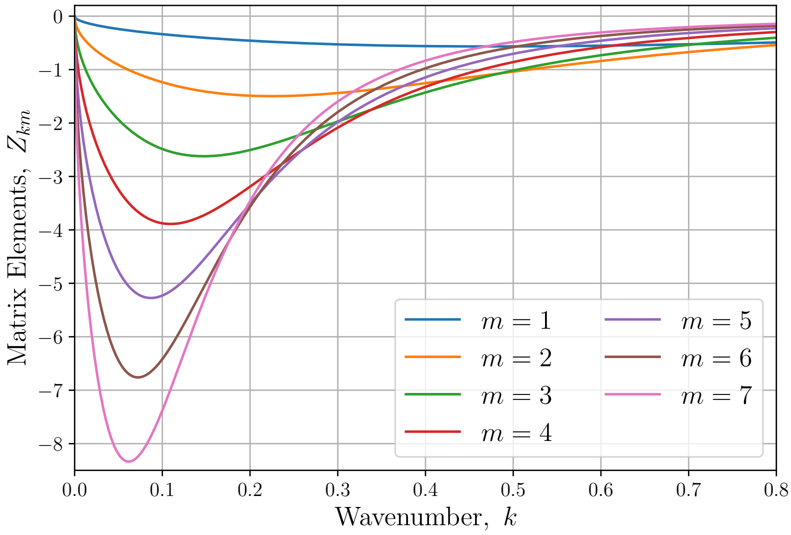

Figure 2.5: The bound-unbound matrix elements Z_{km} for fixed $n \in \{1, 2, 3, 4, 5, 6, 7\}$ are plotted in the range $0 \leq k \leq 0.8$.

Similarly to the bound-bound matrix elements, we see that the bound-unbound matrix elements grow with increasing principle quantum number and eventually decays. This is shown in Figure 2.3 above, where the matrix elements from between the first few bound states and a portion of the continuum are plotted. We have now calculated the desired matrix elements and aim to apply them. In particular, we want to calculate the associated oscillator strengths g_{nm} that will be used extensively in later sections. In addition, they help verify the calculations in this section through sum rules. The oscillator strength is a dimensionless measure of the degree to which a transition from one state to another is allowed [Sakurai and Napolitano, 2017] and is in atomic units defined as

$$g_{mn} = 2(E_n - E_m)\,|Z_{nm}|^2\,. \tag{2.1.50}$$

To showcase the simplest examples of this quantity, we can consider the oscillator strengths from the ground state to an excited state lying either in the discrete or continuous spectra. For the bound-bound transitions from the ground state, we need the matrix elements Z_{n1} using the expression above. These reduce to

$$\begin{aligned} Z_{n1} &= \frac{8(-1)^n (n)^{5/2}}{(1-n)^4}\left(\frac{1-n}{1+n}\right)^{1+n} \\ &\quad\times \left\{1\frac{n-1}{1+n}{}_2F_1\left(1-n, 1-1, 2, \frac{-4mn}{(1-n)^2}\right) + (1-1)\,{}_2F_1\left(1-n, 2-1, 2, \frac{-4mn}{(1-n)^2}\right)\right\} \\ &= \frac{-8n^{5/2}}{(n-1)^4}\left(\frac{n-1}{n+1}\right)^{n+2}, \end{aligned} \tag{2.1.51}$$

where we note that the first hypergeometric function truncates immediately because it has argument zero. Using equation (2.1.50), the bound-bound oscillator strength becomes

$$\begin{aligned} g_{1n} &= 2\left(-\frac{1}{2n^2} - \left(-\frac{1}{2}\right)\right)\left|\frac{-8n^{5/2}}{(n-1)^4}\left(\frac{n-1}{n+1}\right)^{n+2}\right|^2 \\ &= \frac{n^2-1}{n^2}\,\frac{64n^5}{(n-1)^8}\left(\frac{n-1}{n+1}\right)^{2n+4} \\ &= \frac{64n^3}{(n^2-1)^3}\left(\frac{n-1}{n+1}\right)^{2n}\,. \end{aligned} \tag{2.1.52}$$

Similarly, we now compute the bound-unbound transitions, we instead require the matrix elements Z_{k1}, which we calculated above. For transitions from the ground state, we acquire

$$\begin{aligned}
Z_{k1} &= \frac{8(-1)^1 1^{5/2} k^{1/2}}{(1+ik)^4\sqrt{1-\mathrm{e}^{-2\pi/k}}}\left(\frac{1-ik}{1+ik}\right)^{-i/k-1} \\
&\quad\times\left\{1\frac{1+ik}{1-ik}{}_2F_1\left(1-\frac{i}{k},1-1,2,\frac{4ikm}{(1+ikm)^2}\right)+(1-1)_2F_1\left(1-\frac{i}{k},2-1,2,\frac{4ikm}{(1+ikm)^2}\right)\right\} \\
&= \frac{-8k^{1/2}}{(k^2+1)^2\sqrt{1-\mathrm{e}^{-2\pi/k}}}\left(\frac{1-ik}{1+ik}\right)^{-i/k}. \qquad (2.1.53)
\end{aligned}$$

We can reduce the second fraction in equation (2.1.53) by introducing polar form. Because $|1 \pm ik| = \sqrt{1+k^2}$, it is clear that the fraction has unit modulus. Using $\arg(z_1/z_2) = \arg z_1 - \arg z_2$, we may then write

$$\begin{aligned}
\arg\left(\frac{1-ik}{1+ik}\right) &= \arg(1-ik) - \arg(1+ik) \\
&= -2\arg(1+ik) \\
&= -2\tan^{-1} k, \qquad (2.1.54)
\end{aligned}$$

where we also have applied $\arg(z) = -\arg(z^\star)$. It is then clear that

$$\begin{aligned}
\left(\frac{1-ik}{1+ik}\right)^{-i/k} &= \left(\mathrm{e}^{-2i\tan^{-1}k}\right)^{-i/k} \\
&= \mathrm{e}^{-2k^{-1}\tan^{-1}k}. \qquad (2.1.55)
\end{aligned}$$

Substituting this representation back into equation (2.1.53) then yields

$$Z_{k1} = \frac{-8k^{1/2}\mathrm{e}^{-2k^{-1}\tan^{-1}k}}{(k^2+1)^2\sqrt{1-\mathrm{e}^{-2\pi/k}}}. \qquad (2.1.56)$$

Using equation (2.1.57), we may then write the bound-unbound oscillator strengths as

$$\begin{aligned}
g_{1k} &= 2\left(\frac{1}{2}k^2-\left(-\frac{1}{2}\right)\right)\left|\frac{-8k^{1/2}\mathrm{e}^{-2k^{-1}\tan^{-1}k}}{(k^2+1)^2\sqrt{1-\mathrm{e}^{-2\pi/k}}}\right|^2 \\
&= \frac{64k\mathrm{e}^{-4k^{-1}\tan^{-1}k}}{(k^2+1)^3(1-\mathrm{e}^{-2\pi/k})}. \qquad (2.1.57)
\end{aligned}$$

Equations (2.1.52) and (2.1.57) form all the oscillator strengths from the ground state. From these we may construct various sum rules by means of the oscillator strength moments, which we define as

$$\mathcal{G}_{m,p} = \sum_{\substack{n=1\\n\neq m}}^{\infty} g_{mn}\left(E_n - E_m\right)^p + \int_0^\infty g_{mk}\left(E_k - E_m\right)^p \mathrm{d}k. \qquad (2.1.58)$$

The most important case is $p = 0$. For the ground state oscillator strengths derived above, we find that $\sum_{n>1} g_{1n} \approx 0.33701694$ and $\int_0^\infty g_{1k}\mathrm{d}k \approx 0.66298244$ such that $\mathcal{G}_{1,0} = 1$. This is the Thomas–Reiche–Kuhn sum rule, see [Bethe and Salpeter, 1957]. The expected result verifies the above matrix element calculations. The observation that the continuum contribution to $\mathcal{G}_{1,0}$ is roughly twice as large as the discrete contribution tells us that transitions to the continuum in some sense are more accessible. In other words, ionization plays a significant role and is a testament to the electron's weak coupling to the substrate. This concludes the present section and we can now move on to the next, in which we discuss the transition energies.

2.2 Electron as Quantum Defect Atom

At this point, we want to consider the transition energies of the electron lying above a cryogenic substrate. We can do this now using the energy expression given in equations (2.1.13) and (2.1.26). Accordingly, the transition energy ΔE from the ground state to the n'th excited state is

$$\begin{aligned} \Delta E_{1n} &= |E_1| - |E_n| \\ &= \overline{\mathrm{Ha}} \left(1 - \frac{1}{n^2} \right) . \end{aligned} \tag{2.2.1}$$

Electrons bound the liquid Helium substrates have been predicted to have a bound spectrum corresponding to a one-dimensional hydrogen atom [Nieto, 2000]. Experiments show however, that the transition energies are roughly $\sim$ 7 GHz too large as discussed in [Nieto, 2000]. This discrepancy demonstrates that the image potential model alone may be inadequate in describing the system. Fortunately, it can be remedied analytically using quantum defect theory, which in this case is a phenomenological extension of the principle quantum number n. In particular, we introduce the quantum defect parameter $\delta = \delta(\ell)$, which ordinarily is a function of the orbital quantum number ℓ. For the purposes of studying the electron system however, it is approximately constant. The quantum defect parameter is an empirical parameter used to fit potentials so that the observed energy eigenvalues are reproduced [Seaton, 1958]. Using the quantum defect parameter, we define the effective principle quantum number $\mathscr{n} = n - \delta$, which in turn allows us to rewrite the transition energies above as

$$\begin{aligned} \Delta E_{1n} &= \overline{\mathrm{Ha}} \left(\frac{1}{(1-\delta)^2} - \frac{1}{(n-\delta)^2} \right) \\ &= \overline{\mathrm{Ha}} \left\{ \left(1 - \frac{1}{n^2} \right) + 2\delta \left(1 - \frac{1}{n^3} \right) + 3\delta^2 \left(1 - \frac{1}{n^4} \right) + \dots \right\} , \end{aligned} \tag{2.2.2}$$

where we in the second line have expanded the transition energy expression in a Taylor series. The image potential model may have certain limitations for various reasons. For instance, a small energy shift, analogous to the linear term $2\delta\overline{\mathrm{Ha}}(1 - n^{-3})$ but significantly weaker in magnitude, arises due to radiative corrections, as discussed in [Shakeshaft and Spruch, 1980] and [Grotch, 1981]. Additionally, the presence of an infinite barrier cut-off at the substrate interface introduces further modifications to the energy levels, as treated in [Grimes et al., 1976]. A simple estimate of the quantum defect parameter that will attempt to account for these effects can be made by fitting equation (2.2.2) to the observed transition energies provided in [Grimes et al., 1976] or [Lambert and Richards, 1981]. Using the 1 → 2 transition $\Delta E_{12} = 125.9$ GHz and the 1 → 3 transition $\Delta E_{13} = 148.6$ GHz of the former, we find

$$\begin{aligned} 125.9\ \mathrm{GHz} &= \overline{\mathrm{Ha}} \left(\frac{1}{(1-\delta)^2} - \frac{1}{(2-\delta)^2} \right) \\ 148.6\ \mathrm{GHz} &= \overline{\mathrm{Ha}} \left(\frac{1}{(1-\delta)^2} - \frac{1}{(3-\delta)^2} \right) \end{aligned} \quad \Rightarrow \quad \begin{aligned} \overline{\mathrm{Ha}} &= 158.4\ \mathrm{GHz} \\ \delta &= 0.0237 \end{aligned} . \tag{2.2.3}$$

This corresponds to a one-dimensional Rydberg atom. Using this to calculate the transition energies, one finds an increase of $\sim$ 7.8 GHz, which is in accordance with the reported values [Nieto, 2000]. The modified Hartree is reproduced within 0.44% in the case of Helium and only requires a small quantum defect parameter. In addition to altering the energy eigenvalues, the effective principal quantum $\mathscr{n}$ also affects the corresponding eigenstates. We cannot simply let $n \longrightarrow n - \delta$ and apply it to the eigenstates as given in equation (2.1.25) since $\mathscr{n} = n - \delta$ is not an integer. Instead, we exploit a mathematical observation used in quantum defect theory, such as in [Nieto, 2000]. The key insight here is that the solutions of the radial hydrogen atom do not actually require that both the principal quantum number n and orbital quantum number ℓ be integers, but rather only that $n - \ell \in \mathbb{Z}_{\geq 0}$. In other words, the factor $\ell(\ell + 1)$ in the r^{-2} effective potential term found in radial Hydrogen atom need not have $\ell \in \mathbb{Z}_{\geq 0}$ for a finite order solution to exist. For the purposes of the electron lying on a cryogenic substrate, this allows us to incorporate the effective principal quantum number by phenomenologically introducing the potential

$$V_\delta(z) = -\frac{\mathcal{Z}e^2}{z} + \frac{\hbar^2}{2m}\frac{(-\delta)((-\delta)+1)}{z^2}, \quad z > 0, \tag{2.2.4}$$

where we effectively have set $\ell = -\delta$ if we compare with the ordinary radial hydrogen atom. This is an example of the Kratzer potential, see for instance [Flügge, 1999], which essentially is a Coulomb potential with an added $1/z^2$ contribution. The $\delta = 0$ case returns the Coulomb potential. In the case of the potential given in (2.2.4), it is clear that the condition $n - \ell \in \mathbb{Z}_{>0}$ is satisfied if we let $n \longrightarrow \mathscr{n}$ given that $\ell = -\delta$ and $\mathscr{n} = n - \delta$. We can now find the corresponding eigenstates by solving the eigenvalue problem as given in equation (2.1.12) under the new potential introduced above. This is accomplished in a similar manner to that of the original problem. We start with the bound states, for which the corresponding eigenvalue problem reads

$$\left(-\frac{\hbar^2}{2m}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{\mathcal{Z}e^2}{4\pi\varepsilon_0 z} - \frac{\hbar^2}{2m}\frac{\delta(1-\delta)}{z^2}\right)\psi_{\mathscr{n}} = E_{\mathscr{n}}\psi_{\mathscr{n}}, \quad z > 0, \tag{2.2.5}$$

where $E_{\mathscr{n}} < 0$. As with the original eigenvalue problem, we move forward by first converting the problem to atomic units as given in equation (2.1.13), which in turn gives

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z} + \frac{\delta(1-\delta)}{2z^2} - \frac{1}{2\mathscr{n}^2}\right)\psi_{\mathscr{n}} = 0. \tag{2.2.6}$$

Once again, we assume that $\psi_{\mathscr{n}}$ is a product of several other functions whose form we determine by investigating the limiting behaviour of $\psi_{\mathscr{n}}$ for small and large z. Looking at equation (2.2.6), it is clear that the quantum defect eigenstates exhibit the same asymptotic behaviour and hence we can conclude that $\exp(-z/\mathscr{n})$ is a factor of the final eigenstate. In the opposite limit, we again assume that the eigenstates behave like a power function z^p. In the $z \longrightarrow 0$ limit, equation (2.2.6) reduces to

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{\delta(1-\delta)}{2z^2}\right)\psi_{\mathscr{n}} \longrightarrow 0, \quad \text{for } z \longrightarrow 0. \tag{2.2.7}$$

Substituting the power function proposal into equation (2.2.7) then shows that $p(p-1) = \delta(\delta - 1)$, which is satisfied for both $p = \delta$ and $p = 1 - \delta$. We want to ensure that the quantum defect eigenstates reduce to the original eigenstates given in equation (2.1.25) once $\delta = 0$. For this reason we must choose $p = 1 - \delta$ and hence we conclude that $z^{1-\delta}$ is a factor of the eigenstates. As such, we deduce that the eigenstates take the form $\psi_{\mathscr{n}} = fz^{1-\delta}\exp(-z/\mathscr{n})$, where $f = f(z)$ is an unknown function. Substituting this proposal into the eigenvalue problem given in equation (2.2.6) then gives

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z} + \frac{\delta(1-\delta)}{2z^2} - \frac{1}{2\mathscr{n}^2}\right)fz^{1-\delta}\mathrm{e}^{-z/\mathscr{n}} = 0. \tag{2.2.8}$$

We can now let the differential operator act on the product in the above equation and expand terms. Doing so and reducing then eventually allows us to write

$$z'\frac{\mathrm{d}^2 f}{\mathrm{d}z'^2} + (2 - 2\delta - z')\frac{\mathrm{d}f}{\mathrm{d}z'} + (\mathscr{n} - 1 + \delta)f = 0, \tag{2.2.9}$$

where we have introduced the substitution $z' = 2z/\mathscr{n}$. This implies that f is another associated Laguerre polynomial with $a = 1 - 2\delta$ and $b = \mathscr{n} - 1 + \delta = n - 1$ as given in equation (A.0.2), meaning $f = L_{n-1}^{(1-2\delta)}(2z/\mathscr{n})$. As such, the bound eigenstates take the form

$$\psi_{\mathscr{n}} = \mathcal{N}_{\mathscr{n}} z^{1-\delta}\mathrm{e}^{-z/\mathscr{n}}\,{}_1F_1\left(1 - n, 2 - 2\delta, \frac{2z}{\mathscr{n}}\right), \tag{2.2.10}$$

where the conversion factor from equation (A.0.6) has been absorbed into the newly introduced normalization constant $\mathcal{N}_n$. We determine the normalization constant $\mathcal{N}_n$ by enforcing the normalization condition for the bound states, $\langle \psi_m | \psi_n \rangle = \delta_{mn}$. The direct calculation initially shows that

$$\left(\frac{n}{2}\right)^{3-2\delta} \mathcal{N}_n^2 \int_0^\infty z'^{2-2\delta} \mathrm{e}^{-z'} \left[{}_1F_1\left(1-n, 2-2\delta, z'\right) \right]^2 \mathrm{d}z', \tag{2.2.11}$$

where we again have made the substitution $z' = 2z/n$. The integral in equation (2.2.11) is a special case of the W. Gordon integral as written in equation (A.0.10), which when applied allows us to write the normalization constant as

$$\begin{aligned} \mathcal{N}_n &= \left\{ \left(\frac{n}{2}\right)^{3-2\delta} \int_0^\infty z'^{2-2\delta} \mathrm{e}^{-z'} \left[{}_1F_1\left(1-n, 2-2\delta, z'\right) \right]^2 \mathrm{d}z' \right\}^{-1/2} \\ &= \left\{ \left(\frac{n}{2}\right)^{3-2\delta} \mathrm{J}^{1,0}_{2-2\delta}\left(1-n, 1-n; 1, 1, 1\right) \right\}^{-1/2} \\ &= \left\{ \left(\frac{n}{2}\right)^{3-2\delta} \frac{\Gamma(2-2\delta+1)(n-1)!}{(2-2\delta)_{n-1}} {}_3F_2(1-n, -1, 2, 2-2\delta, 1, 1) \right\}^{-1/2} \\ &= \left\{ \left(\frac{n}{2}\right)^{3-2\delta} \frac{\Gamma(2-2\delta+1)(n-1)!}{(2-2\delta)_{n-1}} \frac{n-\delta}{1-\delta} \right\}^{-1/2} \\ &= \frac{2^{1-\delta} (n)^{1/2}_{1-2\delta}}{n^{2-\delta}\Gamma(2-2\delta)}, \end{aligned} \tag{2.2.12}$$

where we in the final step have applied the identity $\Gamma(x+1) = x\Gamma(x)$ and the definition of the Pochhammer symbol. Using the normalization constant given in equation (2.2.12), we then find that the eigenstates may be written as

$$\psi_n = \frac{2^{1-\delta} (n)^{1/2}_{1-2\delta}}{n^{2-\delta}\Gamma(2-2\delta)} z^{1-\delta} \mathrm{e}^{-z/n} {}_1F_1\left(1-n, 2-2\delta, \frac{2z}{n}\right). \tag{2.2.13}$$

We have now determined the bound eigenstates of the quantum defect electron. We can see that the eigenstates reduce to those given in equation (2.1.25) as the quantum defect parameter δ goes to zero. To compare these new eigenstates to the originals, we have plotted the wave functions of the ground state and the first excited state for a select few quantum defect parameters δ. This is shown in Figure 2.6 below.

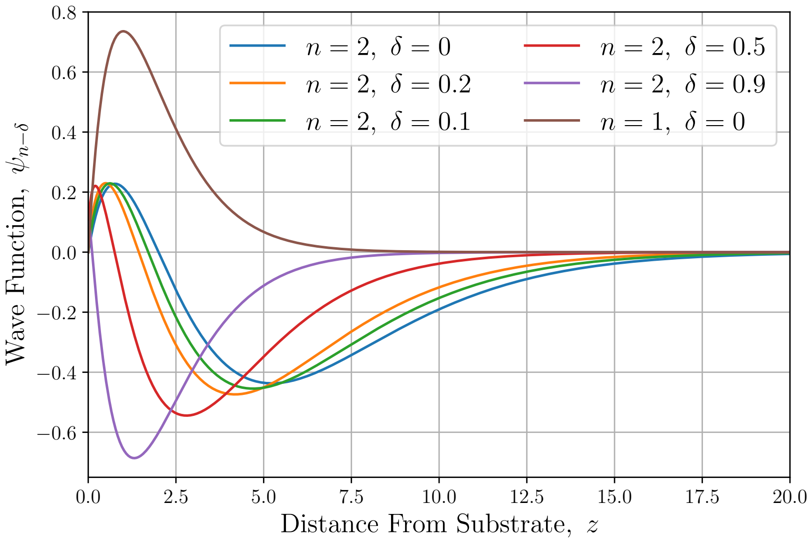

Figure 2.6: Wave function of the bound quantum defect electron plotted as a function of distance from the substrate, which lies in the xy-plane. The first excited state is included for various quantum defect parameters as well as the Coulomb ground state. The quantum defect parameters included are $\delta \in \{0, 0.1, 0.2, 0.5, 0.9\}$. As the quantum defect parameter increases, one finds that the electron is more strongly bound to the interface, meaning the eigenstate decays closer to the origin. As the quantum defect parameter approaches unity for the first excited state, we find that $\psi_{2-\delta} \to -\psi_1$. In terms of the probability density $\psi^\star_{2-\delta}\psi_{2-\delta}$, this "transition" to the lower state is smooth.

The quantum defect parameter δ can generally be chosen freely, even negative values that alter that sign of the potential. The only exceptions are the positive integers and those satisfying $n + 1 - 2\delta \in \mathbb{Z}_{\leq 0}$, which induce poles in the gamma functions found in the normalization constant. An example of this is provided in Figure 2.6, where $\psi_{2-\delta} \to -\psi_1$ as the quantum defect parameter approaches unity. In these cases however, another δ can be chosen, which produces an equivalent potential. In addition, we see that the electron effectively becomes more strongly bound to the surface of the substrate as the quantum defect parameter δ increases. We now turn to the unbound states. Here, we refrain from defining an effective wavenumber because the unbound eigenstates already form a continuum in k and because it introduces complications when verifying sum rules. As such, we again introduce energy eigenvalues $E_k = \overline{\mathrm{H}}\mathrm{a}k^2/2$ for the Kratzer potential. Using the previously introduced atomic units, the eigenvalue problem for the unbound states eventually read

$$\left(\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z} + \frac{\delta(1-\delta)}{2z^2} + \frac{1}{2}k^2\right)\psi_k = 0. \tag{2.2.14}$$

We can now apply the same argumentation as with the bound case to deduce that $\psi_k = f z^{1-\delta}\exp(-ikz)$, where we again choose $\exp(-ikz)$ for consistency. Substituting the proposed expression back into equation (2.2.5) above and reducing then eventually shows that f must satisfy

$$z'\frac{\mathrm{d}^2 f}{\mathrm{d}z'^2} + (2 - 2\delta - z')\frac{\mathrm{d}f}{\mathrm{d}z'} + \left(\delta - 1 - \frac{i}{k}\right)f = 0, \tag{2.2.15}$$

where we also have introduced the change of variable $z' = 2ikz$. This corresponds to an associated Laguerre polynomial for $a = 1 - 2\delta$ and $b = \delta - 1 - i/k$ as given in equation (A.0.2). This allows the unbound eigenstates to be written as

$$\psi_k = \mathcal{N}_k z^{1-\delta}\mathrm{e}^{-ikz}{}_1F_1\left(1 - \delta + \frac{i}{k}, 2 - 2\delta, 2ikz\right), \tag{2.2.16}$$

where we again have absorbed the conversion factor from equation (A.0.6) into the normalization constant $\mathcal{N}_k$. We now δ-normalize the unbound states given in equation (2.2.16) according to

$$\int_0^\infty \psi_k^\star \psi_{k'} \mathrm{d}z = \delta(k - k'). \tag{2.2.17}$$

Using the argumentation outlined in section B shows that the normalization constant is

$$\mathcal{N}_k = \frac{(2k)^{1-\delta} \left|\Gamma(1 - \delta - ik^{-1})\right|}{\sqrt{2\pi} \mathrm{e}^{-\pi/2k} \Gamma(2 - 2\delta)}. \tag{2.2.18}$$

Applying equation (2.2.18) then allows us write the δ-normalized eigenstates as

$$\psi_k = \frac{(2k)^{1-\delta} \left|\Gamma(1 - \delta - ik^{-1})\right|}{\sqrt{2\pi} \mathrm{e}^{-\pi/2k} \Gamma(2 - 2\delta)} z^{1-\delta} \mathrm{e}^{-ikz} {}_1F_1\left(1 - \delta + \frac{i}{k}, 2 - 2\delta, 2ikz\right). \tag{2.2.19}$$

Using equation (2.2.19), we may now determine the corresponding probability density $|\psi_k|^2 = \psi_k^\star \psi_k$. The wave function of the unbound states with wavenumber $k = 1/16$ are shown in Figure 2.7 below for variety of quantum defect parameters.

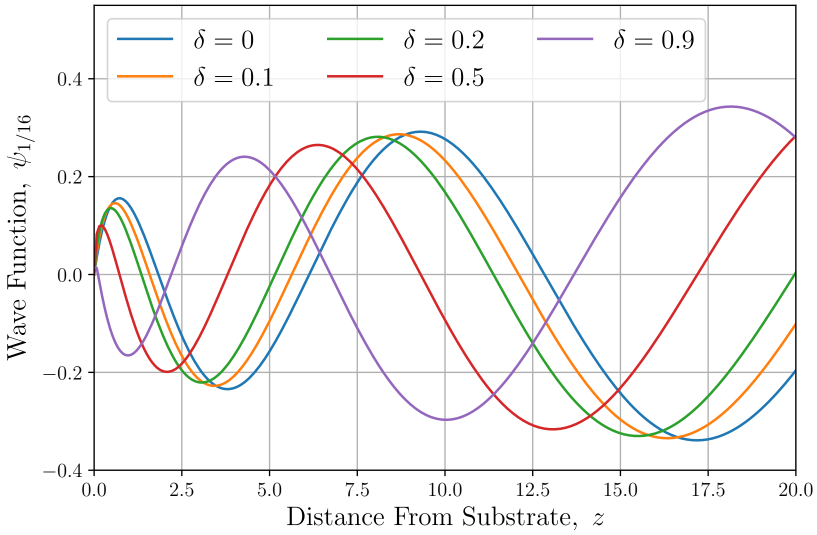

Figure 2.7: Wave function of the unbound quantum defect electron plotted as a function of distance from the cryogenic substrate, which lies in the xy-plane. The principle quantum number is fixed to $k = 1/16$, whilst the quantum defect parameters $\delta \in \{0, 0.1, 0.2, 0.5, 0.9\}$ are chosen. Again, the larger the quantum defect parameter, the more tightly bound the electron is to the substrate as indicated by the fact the eigenstates settle to free states more slowly.

As with the original unbound eigenstates, the eigenstates eventually settle to that of a free particle as we move far enough away from the substrate. As the quantum defect parameter δ increases, it takes longer for the probability density to settle to its final amplitude. We have now extended the original Coulombic model to a Kratzer potential and determined the corresponding eigenstates and energy eigenvalues. Similarly to the original potential, we now move forward and compute the matrix elements.

2.2.1 Quantum Defect Matrix Elements & Oscillator Strengths

We now turn to calculating the matrix elements for the quantum defect electron, where we make use of the eigenstates calculated in the previous section. Again, we focus on the bound-bound and bound-unbound matrix elements. We start with the bound-bound matrix elements $Z_{nm} = \langle \psi_n \, |z| \, \psi_m \rangle$, where ψ_n are given in equation (2.2.13). We initially consider the $n = m$ case, which corresponds to the expectation value of the position coordinate z. A direct calculation gives

$$\begin{aligned} Z_{nn} &= \frac{4^{1-\delta}(n)_{1-2\delta}}{n^{4-2\delta}\Gamma^2(2-2\delta)} \int_0^\infty z^{3-2\delta} \mathrm{e}^{-2z/n} \left[{}_1F_1\left(1-n, 2-2\delta, \frac{2z}{n}\right)\right]^2 \mathrm{d}z \\ &= \frac{(n)_{1-2\delta}}{4\Gamma^2(2-2\delta)} \int_0^\infty z'^{3-2\delta} \mathrm{e}^{-z'} \left[{}_1F_1\left(1-n, 2-2\delta, z'\right)\right]^2 \mathrm{d}z', \end{aligned} \tag{2.2.20}$$

where we have changed the variable of integration so that $z' = 2z/n$. The integral in equation (2.2.20) above is another W. Gordon integral, which may be evaluated by applying equation (A.0.15). This in turn yields

$$\begin{aligned} Z_{nn} &= \frac{(n)_{1-2\delta}}{4\Gamma^2(2-2\delta)} \mathrm{J}^{2,0}_{2-2\delta}\left(1-n, 1-n; 1, 1, 1\right) \\ &= \frac{(n)_{1-2\delta}}{4\Gamma^2(2-2\delta)} \frac{\Gamma(4-2\delta)(n-1)!}{(2-2\delta)_{n-1}} {}_3F_2(1-n, -2, 3, 2-2\delta, 1, 1) \\ &= \frac{(n)_{1-2\delta}}{4\Gamma^2(2-2\delta)} \frac{\Gamma(4-2\delta)(n-1)!}{(2-2\delta)_{n-1}} \frac{3n^2 + \delta(2\delta+1-6n)}{(\delta-1)(2\delta-3)} \\ &= \frac{3}{2}n^2 - \delta\left(3n - \frac{1}{2} - \delta\right), \end{aligned} \tag{2.2.21}$$

which clearly reduces to the original result when δ approaches zero. We now turn to the corresponding off-diagonal matrix elements Z_{nm}, for which $n \neq m$. Calculating the matrix elements directly gives

$$\begin{aligned} Z_{nm} &= \frac{4^{1-\delta}(n)^{1/2}_{1-2\delta}(m)^{1/2}_{1-2\delta}}{(nm)^{2-\delta}\Gamma^2(2-2\delta)} \int_0^\infty z^{1-\delta} \mathrm{e}^{-z/n} {}_1F_1\left(1-n, 2-2\delta, \frac{2z}{n}\right) z z^{1-\delta} \mathrm{e}^{-z/m} \\ &\times {}_1F_1\left(1-m, 2-2\delta, \frac{2z}{m}\right) \mathrm{d}z \\ &= \frac{n^{2-\delta}(n)^{1/2}_{1-2\delta}(m)^{1/2}_{1-2\delta}}{4m^{2-\delta}\Gamma^2(2-2\delta)} \int_0^\infty z'^{3-2\delta} \mathrm{e}^{-\frac{n+m}{2m}z'} {}_1F_1\left(1-n, 2-2\delta, z'\right) \mathrm{e}^{-z/m} \qquad (2.2.22) \\ &\times {}_1F_1\left(1-m, 2-2\delta, \frac{nz'}{m}\right) \mathrm{d}z' \\ &= \frac{n^{2-\delta}(n)^{1/2}_{1-2\delta}(m)^{1/2}_{1-2\delta}}{4m^{2-\delta}\Gamma^2(2-2\delta)} \mathrm{J}^{2,0}_{2-2\delta}\left(1-n, 1-m, 1, \frac{n}{m}\right). \qquad (2.2.23) \end{aligned}$$

where we have changed the variable of integration so that $z' = 2z/n$ and introduced the W. Gordon integral representation. Again, we use equation (A.0.14) to reduce the integral to a $\mathrm{J}^{1,0}_{2-2\delta}$ integral and two $\mathrm{J}^{0,0}_{2-2\delta}$ integrals. We immediately note that the parameter

$$\begin{aligned} \mathcal{A} &= \frac{1}{2}(2-2\delta)\left(1-\frac{n}{m}\right) - 1(1-n) + \frac{n}{m}(1-m) \\ &= 0, \end{aligned} \tag{2.2.24}$$

which again means that the contribution from $\mathrm{J}^{1,0}_{2-2\delta}$ vanishes. Using equation (A.0.14), this just leaves

$$\begin{aligned} Z_{nm} &= \frac{n^{2-\delta}(n)^{1/2}_{1-2\delta}(m)^{1/2}_{1-2\delta}}{4m^{2-\delta}\Gamma^2(2-2\delta)} \frac{4}{1-(n/m)^2} \left\{ (2-2\delta-2(1-m))\, \mathrm{J}^{0,0}_{2-2\delta}\left(1-n, 1-m, 1, \frac{n}{m}\right) \right. \\ &+ \left. 2(1-m)\mathrm{J}^{0,0}_{2-2\delta}\left(1-n, 2-m, 1, \frac{n}{m}\right)\right\} \\ &= \frac{2n^{2-\delta}(n)^{1/2}_{1-2\delta}(m)^{1/2}_{1-2\delta}}{(m^2-n^2)m^{-\delta}\Gamma^2(2-2\delta)} \left\{ m\mathrm{J}^{0,0}_{2-2\delta}\left(1-n, 1-m, 1, \frac{n}{m}\right) \right. \\ &+ \left. (1-m)\mathrm{J}^{0,0}_{2-2\delta}\left(1-n, 2-m, 1, \frac{n}{m}\right)\right\}. \end{aligned} \tag{2.2.25}$$

We now turn to calculating the $\mathrm{J}^{0,0}_{2-2\delta}$ integrals using equation (A.0.10). For the first integral with second argument $1-m$, we acquire

$$\begin{aligned}\mathrm{J}^{0,0}_{2-2\delta}\left(1-n,1-m,1,\frac{n}{m}\right) &= \frac{2^{2-2\delta}\Gamma(2-2\delta)(1+n/m)^{1-n+1-m-2+2\delta}}{(n/m-1)^{1-n}(1-n/m)^{1-m}}\,{}_2F_1\left(1-n,1-m,2-2\delta,\frac{-4n/m}{(n/m-1)^2}\right)\\ &= \frac{4^{1-\delta}(-1)^{n-1}\Gamma(2-2\delta)}{m^{2\delta-2}(m-n)^{2-2\delta}}\left(\frac{m-n}{m+n}\right)^{m+n}\\ &\quad\times {}_2F_1\left(1-n,1-m,2-2\delta,\frac{-4nm}{(n-m)^2}\right). \qquad (2.2.26)\end{aligned}$$

Similarly for the second integral with second argument $2-m$, we acquire

$$\begin{aligned}\mathrm{J}^{0,0}_{2-2\delta}\left(1-n,2-m,1,\frac{n}{m}\right) &= \frac{4^{1-\delta}(-1)^{n-1}\Gamma(2-2\delta)}{m^{2\delta-2}(m-n)^{2-2\delta}}\left(\frac{m-n}{m+n}\right)^{n+m-1}\\ &\quad\times {}_2F_1\left(1-n,2-m,2-2\delta,\frac{-4nm}{(n-m)^2}\right). \qquad (2.2.27)\end{aligned}$$

Substituting back into equation (2.2.25) then yields

$$\begin{aligned}Z_{nm} &= \frac{2n^{2-\delta}(n)^{1/2}_{1-2\delta}(m)^{1/2}_{1-2\delta}}{(m^2-n^2)m^{-\delta}\Gamma^2(2-2\delta)}\left\{m\frac{4^{1-\delta}(-1)^{n-1}\Gamma(2-2\delta)}{m^{2\delta-2}(m-n)^{2-2\delta}}\left(\frac{m-n}{m+n}\right)^{m+n}\right.\\ &\quad\times {}_2F_1\left(1-n,1-m,2-2\delta,\frac{-4nm}{(n-m)^2}\right)+(1-m)\frac{4^{1-\delta}(-1)^{n-1}\Gamma(2-2\delta)}{m^{2\delta-2}(m-n)^{2-2\delta}}\\ &\quad\left.\times\left(\frac{m-n}{m+n}\right)^{n+m-1}{}_2F_1\left(1-n,2-m,2-2\delta,\frac{-4nm}{(n-m)^2}\right)\right\}\\ &= \frac{2^{3-2\delta}(-1)^n(nm)^{2-\delta}(n)^{1/2}_{1-2\delta}(m)^{1/2}_{1-2\delta}}{\Gamma(2-2\delta)(m-n)^{4-2\delta}}\left(\frac{m-n}{m+n}\right)^{m+n} \qquad (2.2.28)\\ &\quad\times\left\{m\frac{n-m}{m+n}{}_2F_1\left(1-n,1-m,2-2\delta,\frac{-4nm}{(n-m)^2}\right)+(m-1){}_2F_1\left(1-n,2-m,2-2\delta,\frac{-4nm}{(n-m)^2}\right)\right\},\end{aligned}$$

We have now found the matrix elements of the bound states and move to the bound-unbound matrix elements Z_{nk}. The immediate calculation yields

$$\begin{aligned}Z_{km} &= \frac{(4k)^{1-\delta}(m)^{1/2}_{1-2\delta}\left|\Gamma(1-\delta-ik^{-1})\right|}{\sqrt{2\pi}m^{2-\delta}\mathrm{e}^{-\pi/2k}\Gamma^2(2-2\delta)}\int_0^\infty z^{1-\delta}\mathrm{e}^{ikz}{}_1F_1\left(1-\delta-\frac{i}{k},2-2\delta,-2ikz\right)zz^{1-\delta}\mathrm{e}^{-z/m}\\ &\quad\times {}_1F_1\left(1-m,2-2\delta,\frac{2z}{m}\right)\mathrm{d}z\\ &= \frac{m^{2-\delta}(m)^{1/2}_{1-2\delta}\left|\Gamma(1-\delta-ik^{-1})\right|}{4\sqrt{2\pi}k^{\delta-1}\mathrm{e}^{-\pi/2k}\Gamma^2(2-2\delta)}\int_0^\infty z'^{3-2\delta}\mathrm{e}^{-(1-ikm)z'/2}{}_1F_1\left(1-\delta-\frac{i}{k},2-2\delta,-ikmz'\right)\\ &\quad\times {}_1F_1\left(1-m,2-2\delta,z'\right)\mathrm{d}z'\\ &= \frac{m^{2-\delta}(m)^{1/2}_{1-2\delta}\left|\Gamma(1-\delta-ik^{-1})\right|}{4\sqrt{2\pi}k^{\delta-1}\mathrm{e}^{-\pi/2k}\Gamma^2(2-2\delta)}\mathrm{J}^{2,0}_{2-2\delta}\left(1-\delta-\frac{i}{k},1-m,-ikm,1\right), \qquad (2.2.29)\end{aligned}$$

where we have introduced $z'=2z/m$ and introduced the W. Gordon integral representation. Here, we again apply equation (A.0.14) to compute the integral. We note that the parameter $\mathscr{A}=0$ and hence the $\mathrm{J}^{2,0}_{2-2\delta}$

integral can be represented using two $\mathrm{J}^{0,0}_{2-2\delta}$ integrals. In particular, we find

$$\begin{aligned}
Z_{k\mathscr{m}} &= \frac{\mathscr{m}^{2-\delta}(m)^{1/2}_{1-2\delta}\left|\Gamma(1-\delta-ik^{-1})\right|}{4\sqrt{2\pi}k^{\delta-1}\mathrm{e}^{-\pi/2k}\Gamma^2(2-2\delta)}\frac{4}{(-ik\mathscr{m})^2-1}\Bigg\{(2-2\delta-2(1-m)) \\
&\quad\times\mathrm{J}^{0,0}_{2-2\delta}\left(1-\delta-\frac{i}{k},1-m,-ik\mathscr{m},1\right)+2(1-m)\mathrm{J}^{0,0}_{2-2\delta}\left(1-\delta-\frac{i}{k},2-m,-ik\mathscr{m},1\right)\Bigg\} \\
&= \frac{-2\mathscr{m}^{2-\delta}(m)^{1/2}_{1-2\delta}\left|\Gamma(1-\delta-ik^{-1})\right|}{\sqrt{2\pi}k^{\delta-1}(1+k^2\mathscr{m}^2)\mathrm{e}^{-\pi/2k}\Gamma^2(2-2\delta)}\Bigg\{\mathscr{m}\mathrm{J}^{0,0}_{2-2\delta}\left(1-\delta-\frac{i}{k},1-m,-ik\mathscr{m},1\right) \\
&\quad+(1-m)\mathrm{J}^{0,0}_{2-2\delta}\left(1-\delta-\frac{i}{k},2-m,-ik\mathscr{m},1\right)\Bigg\}.
\end{aligned}\tag{2.2.30}$$

We now turn to the two $\mathrm{J}^{0,0}_{2-2\delta}$ integrals, which we calculate using equation (A.0.10). The first integral with second argument $1-m$ becomes

$$\begin{aligned}
\mathrm{J}^{0,0}_{2-2\delta}\left(1-\delta-\frac{i}{k},1-m,-ik\mathscr{m},1\right) &= \frac{2^{2-2\delta}\Gamma(2-2\delta)((-ik\mathscr{m})+1)^{1-\delta-i/k+1-m-2+2\delta}}{(1-(-ik\mathscr{m}))^{1-\delta-i/k}((-ik\mathscr{m})-1)^{1-m}} \\
&\quad\times{}_2F_1\left(1-\delta-\frac{i}{k},1-m,2-2\delta,\frac{-4((-ik\mathscr{m}))}{(1-(-ik\mathscr{m}))^2}\right) \\
&= \frac{4^{1-\delta}(-1)^{m-1}\Gamma(2-2\delta)}{(1+ik\mathscr{m})^{2-2\delta}}\left(\frac{1-ik\mathscr{m}}{1+ik\mathscr{m}}\right)^{-i/k-m} \\
&\quad\times{}_2F_1\left(1-\delta-\frac{i}{k},1-m,2-2\delta,\frac{4ik\mathscr{m}}{(1+ik\mathscr{m})^2}\right).
\end{aligned}\tag{2.2.31}$$

Similarly for the second integral with second argument $2-m$, we acquire

$$\begin{aligned}
\mathrm{J}^{0,0}_{2-2\delta}\left(1-\delta-\frac{i}{k},2-m,-ik\mathscr{m},1\right) &= \frac{4^{1-\delta}(-1)^{m}\Gamma(2-2\delta)}{(1+ik\mathscr{m})^{2-2\delta}}\left(\frac{1-ik\mathscr{m}}{1+ik\mathscr{m}}\right)^{1-i/k-m} \\
&\quad\times{}_2F_1\left(1-\delta-\frac{i}{k},2-m,2-2\delta,\frac{4ik\mathscr{m}}{(1+ik\mathscr{m})^2}\right).
\end{aligned}\tag{2.2.32}$$

Substituting the two integrals back into equation (2.2.30) then yields

$$\begin{aligned}
Z_{k\mathscr{m}} &= \frac{-2\mathscr{m}^{2-\delta}(m)^{1/2}_{1-2\delta}\left|\Gamma(1-\delta-ik^{-1})\right|}{\sqrt{2\pi}k^{\delta-1}(1+k^2\mathscr{m}^2)\mathrm{e}^{-\pi/2k}\Gamma^2(2-2\delta)}\Bigg\{\mathscr{m}\frac{4^{1-\delta}(-1)^{m-1}\Gamma(2-2\delta)}{(1+ik\mathscr{m})^{2-2\delta}}\left(\frac{1-ik\mathscr{m}}{1+ik\mathscr{m}}\right)^{-i/k-m} \\
&\quad\times{}_2F_1\left(1-\delta-\frac{i}{k},1-m,2-2\delta,\frac{4ik\mathscr{m}}{(1+ik\mathscr{m})^2}\right)+(1-m)\frac{4^{1-\delta}(-1)^{m}\Gamma(2-2\delta)}{(1+ik\mathscr{m})^{2-2\delta}} \\
&\quad\times\left(\frac{1-ik\mathscr{m}}{1+ik\mathscr{m}}\right)^{1-i/k-m}{}_2F_1\left(1-\delta-\frac{i}{k},2-m,2-2\delta,\frac{4ik\mathscr{m}}{(1+ik\mathscr{m})^2}\right)\Bigg\} \\
&= \frac{2^{3-2\delta}(-1)^m\mathscr{m}^{2-\delta}(m)^{1/2}_{1-2\delta}\left|\Gamma(1-\delta-ik^{-1})\right|}{\sqrt{2\pi}k^{\delta-1}(1+ik\mathscr{m})^{4-2\delta}\mathrm{e}^{-\pi/2k}\Gamma(2-2\delta)}\left(\frac{1-ik\mathscr{m}}{1+ik\mathscr{m}}\right)^{-i/k-m}\Bigg\{\mathscr{m}\frac{1+ik\mathscr{m}}{1-ik\mathscr{m}} \\
&\quad\times{}_2F_1\left(1-\delta-\frac{i}{k},1-m,2-2\delta,\frac{4ik\mathscr{m}}{(1+ik\mathscr{m})^2}\right)+(m-1){}_2F_1\left(1-\delta-\frac{i}{k},2-m,2-2\delta,\frac{4ik\mathscr{m}}{(1+ik\mathscr{m})^2}\right)\Bigg\},
\end{aligned}\tag{2.2.33}$$

which is the final expression of the bound-unbound matrix elements. In conjunction with equation (2.2.28), equation (2.2.33) yields all the matrix elements for the quantum defect electron. We note that both reduce to equations (2.1.43) and (2.1.49) in the $\delta = 0$ case as they should. In addition, the Thomas–Reiche–Kuhn is still satisfied for arbitrary δ. The contribution from bound versus unbound states however do depend on δ. In the ground state case for instance, one finds that $\sum_{n>1} g_{1n} \approx 0.02230407$ and $\int_0^\infty g_{1k}\mathrm{d}k \approx 0.97769592$ for $\delta = 0.8$, whereas $\sum_{n>1} g_{1n} \approx 0.59997693$ and $\int_0^\infty g_{1k}\mathrm{d}k \approx 0.40002301$ for $\delta = -0.8$. Moving forward, when referring to various quantities such as energies E_n, eigenstates ψ_k or matrix elements Z_{nk} and so on, we implicitly mean the quantum defect atom variants, for which $\delta = 0$ is a special case, meaning we abandon the italic subscripts n. We have now fully developed the one-dimensional model describing the out-of-plane eigenstates and energy eigenvalues for the electron-on-Helium qubit system. With these results, we now move forward and begin introducing external electric fields as a means of qubit control.

Chapter 3

Electrostatic Perturbation of Qubit

In this chapter, we develop the results of Chapter one further by studying the effects of introducing an external electrostatic field to the electron qubit system using perturbative methods. By expanding the energies and eigenstates in perturbative series, we find the resulting corrections due to the external field to various orders using two methods. Energy corrections and static polarizabilities are then found. The divergence of the resulting energy expansion is considered and a means of regularization is introduced, which also allows ionization rates to be found. A numerical approach is used to verify the results.

3.1 Electrostatic Perturbation of Qubit

We now aim to apply and further develop the results of chapter one, where we developed the one-dimensional quantum defect model for the electron-on-Helium qubit system. Using these results, we are now able to begin describing qubit control by perturbing the qubit platform with external electric fields. In this chapter, we focus on the electrostatic perturbations of the form $\boldsymbol{\mathcal{E}} = \mathcal{E}_0\hat{\mathbf{z}}$. The field is chosen so that it points away from the substrate, as shown in Figure 3.1 below. In terms of the modified atomic units introduced in the previous chapter, the electric field is measured in units of $\overline{\text{Ha}}/e\overline{a}_0$, which in the case of Helium and Neon corresponds to $\bar{\mathcal{E}}_0 = 0.173$ V/μm and $\bar{\mathcal{E}}_0 = 10.45$ V/μm, respectively.

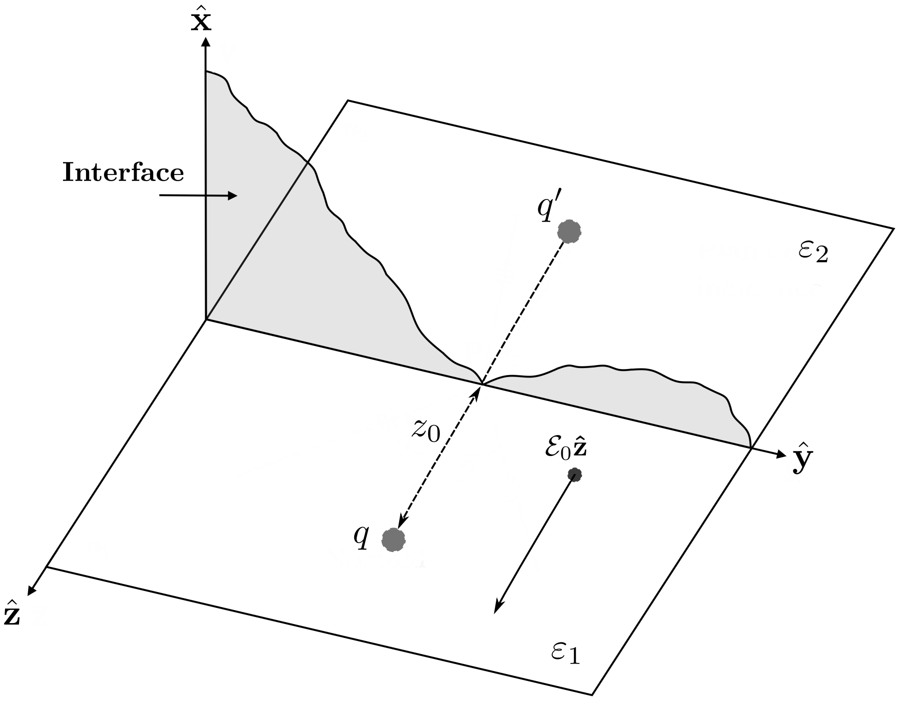

Figure 3.1: Electron-on-Helium qubit system perturbed by the electrostatic field $\boldsymbol{\mathcal{E}} = \mathcal{E}_0\hat{\mathbf{z}}$, which is chosen so that it points away from the substrate. The figure is based on [Pedrotti et al., 2018].

To describe how the electric field affects the Hamiltonian, we will employ the dipole approximation, which is valid when the electric field is approximately spatially uniform over the atom or molecule in question [Pedersen, 2022]. The electron qubit will in the presence of the electric field produce an induced dipole moment $\mathbf{p} = qd\hat{\mathbf{z}} = -ez\hat{\mathbf{z}}$. To this dipole moment we, in dipole approximation, associate an energy $U = -\mathbf{p}\cdot\mathcal{E} = e\mathcal{E}_0 z$. For the quantum mechanical treatment of this problem, we treat the positional coordinate as an operator and hence the contribution to the Hamiltonian due to the electric field is $e\mathcal{E}_0 z$. The modified potential in the presence of an electrostatic field distorts the wave function and alters energy eigenvalues. Physically, this skews the electron cloud in the direction opposite of the field. In modified atomic units, we may write the new full problem as

$$\left\{\hat{H}_0 + \hat{H}_1\right\}\tilde{\psi} = \tilde{E}\tilde{\psi}, \tag{3.1.1}$$

where $\hat{H}_0$ is the original, unperturbed Hamiltonian and $\hat{H}_1 = \mathcal{E}_0 z$ is the so-called perturbation. Because the field is chosen to point away from the substrate, we expect tunnelling to be induced for negative field strengths $\mathcal{E}_0 < 0$ [Pedersen, 2024]. The reason for this is that the Kratzer potential effectively becomes barrier for the electrons to tunnel through. The ease of which this may accomplished increases with field strength. The problem given in equation (3.1.1) is impractical to solve analytically, so we instead employ perturbative methods. Perturbation theory is a selection of methods used to solve a particular class of problems such as this, where an initially exactly solvable problem is perturbed by some small amount. The application of perturbation theory in the context of problems such as this will involve expanding the unknown quantities in powers series' in the perturbation. Before explaining this further, we will for the sake of convenience introduce the notation

$$\begin{gathered} s^{i,j}_{n,m} = \langle\tilde{\psi}_{n,i}|\tilde{\psi}_{m,j}\rangle, \quad S^{i,j}_{n,m} = \langle\tilde{\Psi}_{n,i}|\tilde{\Psi}_{m,j}\rangle, \\ E_{nm} = E_n - E_m, \quad \mathcal{H}^{i,j}_{n,m} = \left\langle\tilde{\psi}_{n,i}\left|\hat{H}_0\right|\tilde{\psi}_{m,j}\right\rangle, \quad \tilde{\mathcal{H}}^{i,j}_{n,m} = \left\langle\tilde{\psi}_{n,i}\left|\hat{H}_1\right|\tilde{\psi}_{m,j}\right\rangle. \end{gathered} \tag{3.1.2}$$

The starting point is the perturbed time-independent Schrödinger equation given in equation (3.1.1), which we will eventually solve using two different perturbative methods. The first step lies in expanding the unknown eigenstate and corresponding energy eigenvalue in a perturbative expansions of the form

$$\tilde{\psi}_n = \sum_{j=0}^{\infty}\tilde{\psi}_{n,j} \quad \wedge \quad \tilde{E}_n = \sum_{j=0}^{\infty}\tilde{E}_{n,j}. \tag{3.1.3}$$

The initial state of the qubit system will be a bound state and hence the unbound states will not be of importance in this derivation. The j'th term in the power series' given in equation (3.1.3) can be thought of as the correction term of order j to the unperturbed energy or eigenstate due to the applied field. In fact, the perturbative expansions given above are power series' in $\mathcal{E}_0$. The first correction is then the linear response, the second correction is quadratic response and so on. The correction order is denoted by the second subscript j. We note that the zeroth order correction terms are simply the unperturbed eigenfunctions and eigenvalues pertaining to the original Hamiltonian $\hat{H}_0$, meaning $\tilde{\psi}_{n,0} = \psi_n$ and $\tilde{E}_{n,0} = E_n$, respectively. We now proceed by substituting both perturbative expansions into equation (3.1.1), which after rearranging terms gives

$$\left(\hat{H}_0 + \hat{H}_1 - \tilde{E}_{n,0} - \tilde{E}_{n,1} - \tilde{E}_{n,2} - \dots\right)\left(\tilde{\psi}_{n,0} + \tilde{\psi}_{n,1} + \tilde{\psi}_{n,2} + \dots\right) = 0. \tag{3.1.4}$$

We now want to determine all the corrections to the eigenstate and energy eigenvalue in such a way that equation (3.1.4) holds for all z and $\mathcal{E}_0$. We accomplish this by grouping terms of equal order, meaning $\hat{H}_i\tilde{\psi}_{n,j}$ and $\tilde{E}_{n,i}\tilde{\psi}_{n,j}$ such that $i + j = p$ for some perturbative order p, and requiring each grouping to vanish. This ensures that every correction holds for all $\mathcal{E}_0$. The reason for this is that every correction of order p is proportional to $\mathcal{E}_0^p$ and hence the $\mathcal{E}_0$-dependence can be divided out in each grouping when requiring them to vanish. This approach allows us to construct a sequence of iteratively solvable equations starting at first order. Expanding (3.1.4) and grouping terms of equal order then gives

$$\left(\hat{H}_0\tilde{\psi}_{n,0} - \tilde{E}_{n,0}\tilde{\psi}_{n,0}\right) + \left(\hat{H}_0\tilde{\psi}_{n,1} + \hat{H}_1\tilde{\psi}_{n,0} - \tilde{E}_{n,0}\tilde{\psi}_{n,1} - \tilde{E}_{n,1}\tilde{\psi}_{n,0}\right) + \cdots = 0. \tag{3.1.5}$$

The first grouping corresponds to the unperturbed system, which we have already solved. For every other perturbative order, we find that

$$\hat{H}_0\tilde{\psi}_{n,p} + \hat{H}_1\tilde{\psi}_{n,p-1} - \sum_{j=0}^{p} \tilde{E}_{n,j}\tilde{\psi}_{n,p-j} = 0, \quad \text{for } p > 0. \tag{3.1.6}$$

We now aim to solve (3.1.6) for each p by determining the corrections of the eigenstate and corresponding energy eigenvalues. The two aforementioned perturbative methods that will be put to use mainly differ in terms of finding the eigenstate corrections. Before proceeding to both methods therefore, we first use equation (3.1.6) to find $\tilde{E}_{n,p}$ for all p. To do so, we multiply equation (3.1.6) through by ψ_n from the left and integrate over $\mathbb{R}_{\geq 0}$, which in turn yields

$$\mathcal{H}_{n,n}^{0,p} + \tilde{\mathcal{H}}_{n,n}^{0,p-1} - \sum_{j=0}^{p} \tilde{E}_{n,j} s_{n,n}^{0,p-j} = 0. \tag{3.1.7}$$

We now exploit the fact that $\hat{H}_0$ is a Hermitian operator and deduce that $\mathcal{H}_{n,n}^{0,p} = E_n s_{n,n}^{0,p}$, where we have used the unperturbed eigenvalue problem $\hat{H}_0\psi_n = E_n\psi_n$ to eliminate the Hamiltonian. It is then clear that the second term in equation (3.1.7) cancels with the $j = 0$ term of the sum. In addition, we see that the $j = p$ term reads $\tilde{E}_{n,p}s_{n,n}^{0,0} = \tilde{E}_{n,p}$ because ψ_n is normalized. In the $p = 1$ case therefore, the energy correction is simply $\tilde{E}_{n,1} = \tilde{\mathcal{H}}_{n,n}^{0,0}$. In other words, the first-order energy correction is the expectation value of the perturbation. For the higher-order corrections, we instead have

$$\tilde{E}_{n,p} = \tilde{\mathcal{H}}_{n,n}^{0,p-1} - \sum_{j=1}^{p-1} \tilde{E}_{n,j} s_{n,n}^{0,p-j}, \quad \text{for } p > 1. \tag{3.1.8}$$

The expression in equation (3.1.8) will become usable once the eigenstate corrections are known. Before determining them using the two methods mentioned earlier, we must first construct a set of conditions that ensure that these eigenstate corrections keep the full eigenstate $\tilde{\psi}_n$ normalized, meaning $\langle\tilde{\psi}_n|\tilde{\psi}_n\rangle = 1$. We accomplish this by substituting the perturbative expansion into the normalization condition, which in turn yields

$$\sum_{j=0}^{\infty}\sum_{l=0}^{\infty} s_{n,n}^{j,l} = 1. \tag{3.1.9}$$

The very first term in the double sum given in equation (3.1.9) above is $s_{n,n}^{0,0} = 1$ because ψ_n is normalized. This cancels with the right-hand side and hence the remaining terms mush vanish. By grouping terms of equal order, we can ensure that the normalization condition is enforced for all $\mathcal{E}_0$. By expanding equation (3.1.9) and grouping terms of equal order, we then find the condition

$$\sum_{j=0}^{p} s_{n,n}^{j,p-j} = 0, \quad \text{for } p > 0. \tag{3.1.10}$$

For every p, equation (3.1.10) places a constraint on $\tilde{\psi}_{n,p}$. To make this condition more explicit, we now extract the first and last terms of the sum above. These two terms are each others complex conjugate and hence $s_{n,n}^{0,p} + s_{n,n}^{p,0} = 2\mathrm{Re}\ s_{n,n}^{0,p}$. Subtracting the remaining sum over to the right-hand side then leaves the final condition. In the $p = 1$ case, we have $\mathrm{Re}\ s_{n,n}^{0,1} = 0$, whereas for every other perturbative order, the condition reads

$$\mathrm{Re}\ s_{n,n}^{0,p} = -\frac{1}{2}\sum_{j=1}^{p-1} s_{n,n}^{j,p-j}, \quad \text{for } p > 1. \tag{3.1.11}$$

With the conditions provided by equation (3.1.11), we are now ready to move forward and determine the eigenstate corrections. This is done using the two methods mentioned above, which will be presented in two subsequent sections.

3.1.1 Basis Expansion Method

In this section, we present the first method of determining the eigenstate corrections, which we for the purposes of this thesis will be denoted the basis expansion method. In this more traditional method, we assume that the eigenstate corrections can be written in terms of an expansion of the unperturbed eigenstates that form a basis under $\hat{H}_0$. This involves summing over the bound states and integrating over the unbound states such that

$$\begin{aligned}\tilde{\psi}_{n,p} &= \sum_{m=1}^{\infty} c_{m,p}\psi_m + \int_0^{\infty} c_{k,p}\psi_k \mathrm{d}k \\ &\equiv \sum\!\!\!\!\!\!\int_{j\in I} c_{j,p}\psi_j,\end{aligned} \tag{3.1.12}$$

where $c_{j,p}$ is the set of basis expansion coefficients associated with the p'th correction to the eigenstate and I is the set of all eigenstates, bound and unbound. This representation allows us to determine the eigenstates if only we determine the basis expansion coefficients. We start with the equal-index coefficients, meaning $c_{n,p}$, which are determined using the normalization condition found in equation (3.1.11). Substituting the above expansion into said condition then yields

$$\mathrm{Re}\int_0^{\infty} \psi_n \sum\!\!\!\!\!\!\int_{j\in I} c_{j,p}\psi_j \mathrm{d}z = -\frac{1}{2}\sum_{j=1}^{p-1} s_{n,n}^{j,p-j}. \tag{3.1.13}$$

We now multiply the eigenstate ψ_n into the sum-integral on the left-hand side. This allows us to interchange the outer spatial integral with the sum-integral. Using the notation given in (3.1.12) and recognizing the resulting inner products, we may write

$$\mathrm{Re}\sum\!\!\!\!\!\!\int_{j\in I} c_{j,p} s_{n,j}^{0,0} = -\frac{1}{2}\sum_{j=1}^{p-1} s_{n,n}^{j,p-j}. \tag{3.1.14}$$

By orthonormality, it is clear that the left-hand side of equation (3.1.14) vanishes entirely except for the $j = n$ case. In the $p = 1$ case, the right-hand side of equation (3.1.14) vanishes and hence $c_{n,1} = 0$. For higher perturbative orders, we instead have

$$c_{n,p} = -\frac{1}{2}\sum_{j=1}^{p-1} s_{n,n}^{j,p-j}, \quad \text{for } p > 1, \tag{3.1.15}$$

which gives all the equal-index coefficients. We now turn to the remaining coefficients, which are found using equation (3.1.6). Substituting in the basis expansion yields

$$\sum\!\!\!\!\!\!\int_{j\in I} \left\{ \hat{H}_0 c_{j,p} + \hat{H}_1 c_{j,p-1} - \sum_{l=0}^{p} \tilde{E}_{n,l} c_{j,p-l} \right\} \psi_j = 0, \tag{3.1.16}$$

Letting the unperturbed Hamiltonian act on the i'th eigenstate in equation (3.1.16) returns the corresponding energy eigenvalue E_i. We can then multiply through by ψ_m for some $m \in I\backslash n$ from the left and integrate over $\mathbb{R}_{\geq 0}$, which after interchanging integrals and sums gives

$$\sum\!\!\!\!\!\!\!\int_{j\in I} \left\{ E_j c_{j,p} s^{0,0}_{m,j} + c_{j,p-1}\tilde{\mathcal{H}}^{0,0}_{m,j} - \sum_{l=0}^{p} \tilde{E}_{n,l} c_{j,p-l} s^{0,0}_{m,j} \right\} = 0. \tag{3.1.17}$$

By orthonormality, we know that $s^{0,0}_{m,j} = \delta_{mj}$ and hence vanishes for all but $j = m$, where the inner product returns unity. Extracting the $l = 0$ term from the sum then yields

$$E_m c_{m,p} + \sum\!\!\!\!\!\!\!\int_{j\in I} c_{j,p-1}\tilde{\mathcal{H}}^{0,0}_{m,j} - E_n c_{m,p} - \sum_{l=1}^{p} \tilde{E}_{n,l} c_{m,p-l} = 0. \tag{3.1.18}$$

We can now solve for the coefficient $c_{m,p}$, which gives

$$c_{m,p} = \frac{1}{E_{mn}} \sum_{l=1}^{p} \tilde{E}_{n,l} c_{m,p-l} - \frac{1}{E_{mn}} \sum\!\!\!\!\!\!\!\int_{j\in I} c_{j,p-1}\tilde{\mathcal{H}}^{0,0}_{m,j}. \tag{3.1.19}$$

Equation (3.1.19) holds for all $m \neq n$, so in conjunction with equation (3.1.15), we have now found every coefficient $c_{m,p}$ for every $m \in I$ and $p > 0$. For the sake of completeness, we note that the zeroth-order coefficients $c_{m,0}$ are delta functions in m that pick out exactly ψ_m in the basis expansion. We are now able to construct a recursive algorithm that calculates all $\tilde{E}_{n,p}$ and $\tilde{\psi}_{n,p}$. Starting at $p = 1$, we compute the first-order energy correction $\tilde{E}_{n,1} = \tilde{\mathcal{H}}^{0,0}_{n,n}$. The corresponding eigenstate correction $\tilde{\psi}_{n,1}$ can then be found by calculating its expansion coefficients in the $p = 1$ case of equations (3.1.15) and (3.1.19). We can then move on to the next order of perturbation, where the same procedure is repeated but for energy corrections found using equation (3.1.8). In this way, we can keep calculating corrections of higher order using all the previously computed corrections. In particular, the order of calculation is given in the sequence below:

$$\tilde{E}_{n,1} \Rightarrow c_{m,1} \Rightarrow \tilde{\psi}_{n,1} \Rightarrow \tilde{E}_{n,2} \Rightarrow c_{m,2} \Rightarrow \tilde{\psi}_{n,2} \Rightarrow \tilde{E}_{n,3} \Rightarrow \dots . \tag{3.1.20}$$

To showcase the procedure in a specific example, we consider the perturbation of the $\delta = 0$ ground state, for which $\psi_1 = 2z\mathrm{e}^{-z}$, and compute up to the second-order energy. The first-order energy correction is

$$\begin{aligned} \tilde{E}_{1,1} &= \tilde{\mathcal{H}}^{0,0}_{1,1} \\ &= \mathcal{E}_0 Z_{11} \\ &= \frac{3}{2}\mathcal{E}_0, \end{aligned} \tag{3.1.21}$$

where we have applied the diagonal matrix element given in equation (2.1.35). The first-order equal index coefficient is $c_{1,1} = 0$. Using equation (3.1.19), the remaining first-order coefficients are

$$\begin{aligned} c_{m,1} &= \frac{1}{E_{m1}} \tilde{E}_{1,1} c_{m,0} - \frac{1}{E_{m1}} \sum\!\!\!\!\!\!\!\int_{j\in I} c_{j,0}\tilde{\mathcal{H}}^{0,0}_{m,j} \\ &= \frac{\tilde{E}_{1,1}}{E_{m1}}\delta_{1m} - \frac{\tilde{\mathcal{H}}^{0,0}_{m,1}}{E_{m1}} \\ &= \frac{\tilde{\mathcal{H}}^{0,0}_{m,1}}{E_{1m}} \\ &= \mathcal{E}_0 \frac{Z_{m1}}{E_{1m}}, \end{aligned} \tag{3.1.22}$$

where we have used the fact that $m \neq 1$ to eliminate the first term in the second line. Using the basis expansion in conjunction with the above coefficient, we then have

$$\begin{aligned}
\tilde{\psi}_{1,1} &= \mathop{\sum\!\!\!\!\!\!\!\int}_{\substack{j\in I\\ j\neq 1}} c_{j,1}\psi_j + c_{1,1}\psi_1 \\
&= \mathcal{E}_0 \mathop{\sum\!\!\!\!\!\!\!\int}_{\substack{j\in I\\ j\neq 1}} \frac{Z_{j1}}{E_{1j}}\psi_j,
\end{aligned} \tag{3.1.23}$$

where the off-diagonal matrix elements are given in equations (2.2.28) and (2.2.33). Moving up to perturbative order $p = 2$, the second-order energy correction is given by equation (3.1.8), which yields

$$\begin{aligned}
\tilde{E}_{1,2} &= \tilde{\mathcal{H}}^{0,1}_{1,1} - \tilde{E}_{1,1} s^{0,1}_{1,1} \\
&= \mathop{\sum\!\!\!\!\!\!\!\int}_{\substack{j\in I\\ j\neq 1}} \left\{ \mathcal{E}_0 \frac{Z_{j1}}{E_{1j}} \tilde{\mathcal{H}}^{0,0}_{1,j} - E_{1,1}\mathcal{E}_0 \frac{Z_{j1}}{E_{1j}} s^{0,0}_{1,j} \right\} \\
&= \mathcal{E}_0^2 \mathop{\sum\!\!\!\!\!\!\!\int}_{\substack{j\in I\\ j\neq 1}} \frac{|Z_{j1}|^2}{E_{1j}} \\
&= \frac{1}{2}\mathcal{E}_0^2 \mathop{\sum\!\!\!\!\!\!\!\int}_{\substack{j\in I\\ j\neq 1}} \frac{g_{j1}}{E_{1j}E_{j1}} \\
&= -\frac{1}{2}\mathcal{E}_0^2 \mathcal{G}_{1,-2},
\end{aligned} \tag{3.1.24}$$

where we have introduced the oscillator strength moment. The next terms to calculate would be the second-order coefficients $c_{m,2}$ that provide the second-order eigenstate correction. This second-order correction becomes tricky to calculate because it contains unbound-unbound matrix elements $Z_{kk'}$. The method also quickly becomes more computationally expensive because an additional sum-integral is introduced at every higher order. Additionally, due to the complex form of the matrix elements, the sum-integrals generally cannot be evaluated analytically in closed form. As a result, this approach is, for all practical purposes, numerical. We now move onto the second method for computing the eigenstate corrections.

3.1.2 Dalgarno-Lewis Method

In the second method, which is attributed to Dalgarno and Lewis [Dalgarno and Lewis, 1955], we instead determine the eigenstate corrections by solving equation (3.1.6) directly. An advantage of this is that we are able to provide closed-form eigenstate corrections instead of the basis expansions found in the previous section. The initial trick to accomplish this is to assume that $\tilde{\psi}_{n,p} = f_p\psi_n$ for every p, where ψ_n is the unperturbed eigenstate and f_p is the so-called field factor. This allows us to determine the eigenstate corrections if only we can find f_p. We accomplish this by first substituting the proposed form of the eigenstate correction into equation (3.1.6), which in turn yields

$$\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + V\right) f_p\psi_n + \hat{H}_1 f_{p-1}\psi_n - \sum_{j=0}^{p} \tilde{E}_{n,j} f_{p-j}\psi_n = 0, \tag{3.1.25}$$

where we have written the unperturbed Hamiltonian $\hat{H}_0$ explicitly in terms of its kinetic and potential energy operators. We now let the differential operator in equation (3.1.25) act on the product $f_p\tilde{\psi}_n$ and expand terms. Rearranging the resulting terms then shows that

$$f_p\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + V - E_n\right)\psi_n - \frac{1}{2}\frac{\mathrm{d}^2 f_p}{\mathrm{d}z^2}\psi_n - \frac{\mathrm{d}f_p}{\mathrm{d}z}\frac{\mathrm{d}\psi_n}{\mathrm{d}z} + \hat{H}_1 f_{p-1}\psi_n - \sum_{j=1}^{p} \tilde{E}_{n,j} f_{p-j}\psi_n = 0, \tag{3.1.26}$$

where we have moved the first term of the sum into the parenthesis. An important observation to make at this point is that the parenthesis must vanish because it simply is the Hamiltonian minus its expectation value, see equation (2.2.6). This then leaves

$$\frac{\mathrm{d}^2 f_p}{\mathrm{d}z^2}\psi_n + 2\frac{\mathrm{d}f_p}{\mathrm{d}z}\frac{\mathrm{d}\psi_n}{\mathrm{d}z} = 2\left[\hat{H}_1 f_{p-1}\psi_n - \sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j}\psi_n\right], \tag{3.1.27}$$

where we have multiplied through by a factor 2. We now divide equation (3.1.27) through by ψ_n. Using logarithmic differentiation, this allows us to write

$$\frac{\mathrm{d}f_p'}{\mathrm{d}z} + 2f_p'\frac{\mathrm{d}\ln\psi_n}{\mathrm{d}z} = 2\left[\hat{H}_1 f_{p-1} - \sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j}\right], \tag{3.1.28}$$

where $f_p' = \mathrm{d}f_p/\mathrm{d}z$ is the derivative. We note that the first term on the right-hand side reduces to $\hat{H}_1 f_{p-1}$ because the perturbation is a multiplicatively commutative operator. Equation (3.1.28) is exactly solvable since it is a first-order linear differential equation in f_p'. To solve it, we start by multiplying the equation through by an integrating factor $\exp(2\int\{\ln\psi_n\}'\,\mathrm{d}z) = \psi_n^2$, which in turn yields

$$\frac{\mathrm{d}f_p'\psi_n^2}{\mathrm{d}z} = 2\left[\hat{H}_1 f_{p-1} - \sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j}\right]\psi_n^2, \tag{3.1.29}$$

where we have factored the left-hand side using the product rule of differentiation. We can now find the derivative f_p' by integrating equation (3.1.29) and dividing through by a factor ψ_n^2, which in turn gives

$$f_p' = 2\int\left[\hat{H}_1 f_{p-1} - \sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j}\right]\psi_n^2\mathrm{d}z\psi_n^{-2} + 2C_p^{(1)}\psi_n^{-2}, \tag{3.1.30}$$

where $C_p^{(1)}$ is the first of two integration constants. We now have the derivative f_p', which means that the field factor f_p is found by integrating once more. This yields

$$f_p = 2\iint\left[\hat{H}_1 f_{p-1} - \sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j}\right]\psi_n^2\mathrm{d}z\psi_n^{-2}\mathrm{d}z + 2C_p^{(1)}\int\psi_n^{-2}\mathrm{d}z + 2C_p^{(2)}, \tag{3.1.31}$$

where $C_p^{(2)}$ is the second constant of integration. We now want to determine the two constants of integration. By recalling $\psi_n \sim \exp(-z/n)$, it is clear that ψ_n^{-2} is not square-integrable on $\mathbb{R}_{\geq 0}$. We cannot then ensure that $\tilde{\psi}_{n,p}$ is normalizable unless $C_p^{(1)} = 0$. Using $\tilde{\psi}_{n,p} = f_p\psi_n$, we may for the moment express the p'th order eigenstate correction as

$$\tilde{\psi}_{n,p} = 2\iint\left[\hat{H}_1 f_{p-1} - \sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j}\right]\psi_n^2\mathrm{d}z\psi_n^{-2}\mathrm{d}z\psi_n + 2C_p^{(2)}\psi_n. \tag{3.1.32}$$

The remaining integration constant $C_p^{(2)}$ is determined by requiring that $\tilde{\psi}_{n,p}$ satisfies the normalization condition (3.1.11). Substituting in the eigenstate as given in equation (3.1.32), we then get

$$2\mathrm{Re}\int_0^\infty\left\{\iint\left[\hat{H}_1 f_{p-1} - \sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j}\right]\psi_n^2\mathrm{d}z\psi_n^{-2}\mathrm{d}z + C_p^{(2)}\right\}\psi_n^2\mathrm{d}z = -\frac{1}{2}\sum_{j=1}^{p-1} s_{n,n}^{j,p-j}. \tag{3.1.33}$$

Because unperturbed eigenstates ψ_n are normalized, it means that the second term on the left-hand side simply evaluates to $2C_p^{(2)}$. This allows us to solve for the second coefficient and hence

$$C_p^{(2)} = \mathrm{Re}\int_0^\infty\iint\left[\sum_{j=1}^{p}\tilde{E}_{n,j} f_{p-j} - \hat{H}_1 f_{p-1}\right]\psi_n^2\mathrm{d}z\psi_n^{-2}\mathrm{d}z\psi_n^2\mathrm{d}z - \frac{1}{4}\sum_{j=1}^{p-1} s_{n,n}^{j,p-j}. \tag{3.1.34}$$

Assuming the integrals given in equations (3.1.32) and (3.1.34) can be evaluated analytically, we are now able express the eigenstate corrections in closed form. We note that the second sum in equation (3.1.34) vanishes for the $p = 1$ case. The remaining corrections can now be found by following an algorithm similar to the one presented in the basis expansion method:

$$\tilde{E}_{n,1} \;\Rightarrow\; C_1^{(2)} \;\Rightarrow\; f_1 \;\Rightarrow\; \tilde{\psi}_{n,1} \;\Rightarrow\; \tilde{E}_{n,2} \;\Rightarrow\; C_2^{(2)} \;\Rightarrow\; f_2 \;\Rightarrow\; \dots . \tag{3.1.35}$$

As with the previous method, we now showcase the procedure for the $\delta = 0$ ground state up to the second-order energy correction. Here, we have $\psi_1 = 2z\mathrm{e}^{-z}$ and $\tilde{E}_{1,1} = 3\mathcal{E}_0/2$. The integration constant $C_{1,1}$ is then found with (3.1.34), which gives

$$\begin{aligned}
C_1^{(2)} &= \mathrm{Re}\int_0^\infty \iint \left[\frac{3}{2}\mathcal{E}_0 - \mathcal{E}_0 z\right] \psi_1^2 \mathrm{d}z \psi_1^{-2} \mathrm{d}z \psi_1^2 \mathrm{d}z \\
&= \frac{1}{16}\mathcal{E}_0 \int_0^\infty \iint \left[3 - z'\right] z'^2 \mathrm{e}^{-z'} \mathrm{d}z' z'^{-2} \mathrm{e}^{z'} \mathrm{d}z' z'^2 \mathrm{e}^{-z'} \mathrm{d}z' \\
&= \frac{1}{16}\mathcal{E}_0 \int_0^\infty \int z'^3 \mathrm{e}^{-z'} \times z'^{-2} \mathrm{e}^{z'} \mathrm{d}z' z'^2 \mathrm{e}^{-z'} \mathrm{d}z' \\
&= \frac{1}{16}\mathcal{E}_0 \int_0^\infty \int z' \mathrm{d}z' z'^2 \mathrm{e}^{-z'} \mathrm{d}z' \\
&= \frac{1}{32}\mathcal{E}_0 \int_0^\infty z'^4 \mathrm{e}^{-z'} \mathrm{d}z' \\
&= \frac{3}{4}\mathcal{E}_0,
\end{aligned} \tag{3.1.36}$$

where the substitution $z' = 2z$ has been introduced. We can now substitute this integration constant into equation (3.1.32), which allows us to compute the corresponding eigenstate correction as

$$\begin{aligned}
\tilde{\psi}_{1,1} &= 2\iint \left[\mathcal{E}_0 z - \frac{3}{2}\mathcal{E}_0\right] \psi_1^2 \mathrm{d}z \psi_1^{-2} \mathrm{d}z \psi_1 + 2 \cdot \frac{3}{4}\mathcal{E}_0 \psi_1 \\
&= \frac{1}{4}\mathcal{E}_0 \left\{\iint \left[z' - 3\right] \psi_1^2 \mathrm{d}z' \psi_1^{-2} \mathrm{d}z' + 6\right\} \psi_1 \\
&= \frac{1}{2}\left(3 - z^2\right) \mathcal{E}_0 \psi_1 .
\end{aligned} \tag{3.1.37}$$

The field factor is also easily found in the form given in equation (3.1.37). We have now acquired the first-order corrections and can use them to calculate the second-order correction to the energy using equation (3.1.8), which yields

$$\begin{aligned}
\tilde{E}_{1,2} &= \tilde{\mathcal{H}}_{1,1}^{0,1} - \tilde{E}_{1,1} s_{1,1}^{0,1} \\
&= \frac{1}{8}\mathcal{E}_0^2 \int_0^\infty \left\{6 - \frac{1}{2}z'^2\right\} z' \psi_1^2 \mathrm{d}z' - \frac{3}{8}\mathcal{E}_0^2 \int_0^\infty \left\{6 - \frac{1}{2}z'^2\right\} \psi_1^2 \mathrm{d}z' \\
&= \frac{1}{16}\mathcal{E}_0^2 \int_0^\infty \left(6 - \frac{1}{2}z'^2\right) (z' - 3) z'^2 \mathrm{e}^{-z'} \mathrm{d}z' \\
&= -\frac{3}{2}\mathcal{E}_0^2 .
\end{aligned} \tag{3.1.38}$$

The procedure is readily repeated to higher-order corrections for the Coulomb potential. In comparison with the basis expansion method, this is one of the clear advantages of the Dalgarno-Lewis approach. We acquire closed-form corrections even to high perturbative order, but we also avoid dealing with the unbound-unbound matrix elements. In particular, the next four energy corrections of the Coulomb ground state become

$$\begin{aligned}
\tilde{E}_{1,3} &= \frac{27}{4}\mathcal{E}_0^3, \quad \tilde{E}_{1,4} = -\frac{795}{16}\mathcal{E}_0^4, \\
\tilde{E}_{1,5} &= \frac{3843}{8}\mathcal{E}_0^5, \quad \tilde{E}_{1,6} = -5583\mathcal{E}_0^6,
\end{aligned} \tag{3.1.39}$$

and similarly, the next two eigenstate corrections become

$$\begin{aligned}\tilde{\psi}_{1,3} &= \left(-\frac{1}{48}z^6 - \frac{1}{12}z^5 - \frac{3}{16}z^4 - \frac{1}{2}z_1^3 + z^2 + \frac{585}{16}\right)\mathcal{E}_0^3\psi_1,\\ \tilde{\psi}_{1,4} &= \left(\frac{1}{384}z^8 + \frac{1}{48}z^7 + \frac{31}{288}z^6 + \frac{11}{24}z^5 + \frac{39}{32}z^4 + \frac{57}{16}z^3 - \frac{243}{32}z^2 - \frac{23415}{64}\right)\mathcal{E}_0^4\psi_1.\end{aligned} \tag{3.1.40}$$

The process of calculating these corrections is automated using the Python script attached in the appendix. For the Kratzer-like potential, the indefinite integration required is more difficult, in particular for symbolic integration packages within Python and Mathmatica. The eigenstate corrections are clearly closely related to the Laguerre polynomials and hence an attempt was made to evaluate the indefinite integrals above using the methods introduced in [Conway, 2021, Conway, 2015]. This was ultimately unsuccessful because a fitting integral identity could not be recovered. We therefore opted for a slightly different, albeit less elegant, approach, which involves simply constructing the corrections from equation (3.1.6) directly. We can do this because the general form of the corrections $\tilde{\psi}_{n,p}$ can be deduced from the Dalgarno-Lewis approach. By computing a number of corrections and studying how the Dalgarno-Lewis corrections are constructed, we find that $\tilde{\psi}_{n,p} \propto z^{1-\delta}\mathrm{e}^{-z/\mathcal{n}}\mathcal{P}_{n,p}(z)$, where $\mathcal{P}_{n,p}(z)$ is a polynomial of degree $2p+n-1$ [Pedersen, 2024]. The Coulomb potential and Kratzer-like potential share the same general form. Once this is known, we can substitute a general polynomial of degree $2p+n-1$ with coefficients $\mathcal{a}_k$ into equation (3.1.6) and group terms in equal power of z. Equating every grouping to zero and solving for each a_k gives the solution. Although this approach in principle works for any perturbative order, we shall focus on the first- and second-order corrections in this section. We start with the $p=1$ case of equation (3.1.6), which reads

$$\hat{H}_0\tilde{\psi}_{n,1} + \hat{H}_1\psi_n - E_n\tilde{\psi}_{n,1} - \tilde{E}_{n,1}\psi_n = 0. \tag{3.1.41}$$

We can now substitute the unperturbed Hamiltonian of the Kratzer-like potential as given in equation (2.2.5), the perturbation $\hat{H}_1 = \mathcal{E}_0 z$, the unperturbed energy $E_n = -1/2\mathcal{n}^2$ and the first-order energy correction $\tilde{E}_{n,1} = \mathcal{E}_0 Z_{nn}$ into equation (3.1.41) so that

$$\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{1}{z} - \frac{\delta(1-\delta)}{2z^2} + \frac{1}{2\mathcal{n}^2}\right)\tilde{\psi}_{n,1} + (z - Z_{nn})\,\mathcal{E}_0\psi_n = 0. \tag{3.1.42}$$

In terms of the associated Laguerre polynomials, the unperturbed eigenstates read

$$\begin{aligned}\psi_n &= \mathcal{M}_n z^{1-\delta}\mathrm{e}^{-z/\mathcal{n}}L_{n-1}^{(1-2\delta)}\left(\frac{2z}{\mathcal{n}}\right)\\ &= \mathcal{M}_n z^{1-\delta}\mathrm{e}^{-z/\mathcal{n}}\sum_{k=0}^{n-1}\binom{n-2\delta}{n-1-k}\frac{(-1)^k}{k!}\left(\frac{2z}{\mathcal{n}}\right)^k,\end{aligned} \tag{3.1.43}$$

where $\mathcal{M}_n = 2^{1-\delta}\mathcal{n}^{\delta-2}(n)_{1-2\delta}^{-1/2}$. Because we are working with the first-order equation, we have $p=1$ and hence $\mathcal{P}_{n,1}(z)$ is a polynomial of degree $n+1$. To ensure that both the normalization factor $\mathcal{M}_n$ and field strength $\mathcal{E}_0$ cancel, we choose to write

$$\begin{aligned}\tilde{\psi}_{n,1} &= \mathcal{M}_n\mathcal{E}_0 z^{1-\delta}\mathrm{e}^{-z/\mathcal{n}}\mathcal{P}_{n,1}\left(\frac{2z}{\mathcal{n}}\right)\\ &= \mathcal{M}_n\mathcal{E}_0 z^{1-\delta}\mathrm{e}^{-z/\mathcal{n}}\sum_{k=0}^{n+1}\mathcal{a}_k\left(\frac{2z}{\mathcal{n}}\right)^k\end{aligned} \tag{3.1.44}$$

where we for convenience also have matched the argument $2z/\mathcal{n}$ of the Laguerre polynomial in equation (3.1.43). Substituting both eigenstates into equation (3.1.42) then yields

$$\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{1}{z} - \frac{\delta(1-\delta)}{2z^2} + \frac{1}{2\mathcal{n}^2}\right)z^{1-\delta}\mathrm{e}^{-z/\mathcal{n}}\mathcal{P}_{n,1} + (z - Z_{nn})\,z^{1-\delta}\mathrm{e}^{-z/\mathcal{n}}L_{n-1}^{(1-2\delta)}\left(\frac{2z}{\mathcal{n}}\right) = 0. \tag{3.1.45}$$

Applying the differential operator and reducing then eventually shows that

$$-\frac{1}{2}\mathscr{n}z\frac{\mathrm{d}^2\mathcal{P}_{n,1}}{\mathrm{d}z^2}+z\frac{\mathrm{d}\mathcal{P}_{n,1}}{\mathrm{d}z}-(1-\delta)\mathscr{n}\frac{\mathrm{d}\mathcal{P}_{n,1}}{\mathrm{d}z}-(n-1)\mathcal{P}_{n,1}+(z-Z_{nn})\,\mathscr{n}zL_{n-1}^{(1-2\delta)}\left(\frac{2z}{\mathscr{n}}\right)=0. \tag{3.1.46}$$

We now substitute the sum forms of $\mathcal{P}_{n,1}$ and $L_{n-1}^{(1-2\delta)}(2z/\mathscr{n})$ as given in equations (3.1.43) and (3.1.44) and group terms in equal power of z. For some grouping of terms of order z^k, where $k < n+1$, we find the recurrence relation

$$\mathscr{a}_{k+1}=\frac{k+1-n}{(k+1)(k+2-2\delta)}\mathscr{a}_k+\frac{(-1)^k k\mathscr{n}^2}{2(k+1)(k+2-2\delta)k!}\left(\frac{\mathscr{n}(k-1)(k-2\delta)}{2(n+1-k)}+Z_{nn}\right)\binom{n-2\delta}{n-k}. \tag{3.1.47}$$

A general grouping of terms of order z^k contains the coefficients $\mathscr{a}_k$, $\mathscr{a}_{k+1}$, but importantly the final grouping corresponding to $k = n+1$ only contains $\mathscr{a}_{n+1}$, which allows it to be determined immediately. It turns out this gives $\mathscr{a}_{n+1} = (-1)^n\mathscr{n}^3/8\Gamma(n)$. This allows every coefficient $\mathscr{a}_k$ to be determined save $\mathscr{a}_0$, which is found from the first-order normalization condition $s_{n,n}^{0,1} = 0$. By substituting the $k = 0$ case into the $k = 1$ case, the $k = 1$ case into the $k = 2$ case and so on, the coefficients become of the form $\mathscr{a}_k = \mathscr{a}_0\mathscr{a}_k^{(1)} - \mathscr{a}_k^{(2)}$, which in turn allows us to write the first-order normalization condition as

$$\int_0^\infty\sum_{k=0}^{n+1}\left(\mathscr{a}_0\mathscr{a}_k^{(1)}-\mathscr{a}_k^{(2)}\right)\left(\frac{2z}{\mathscr{n}}\right)^k z^{2-2\delta}\mathrm{e}^{-2z/\mathscr{n}}{}_1F_1\left(1-n,2-2\delta,\frac{2z}{\mathscr{n}}\right)\mathrm{d}z=0. \tag{3.1.48}$$

We now split the sum in two and interchange sums and integrals. Substituting the change of variable $z' = 2z/\mathscr{n}$ and applying equation (A.0.16) then allows us to express the first coefficient as

$$\mathscr{a}_0=\frac{\sum_{k=0}^{n+1}\mathscr{a}_k^{(2)}\Gamma(k+3-2\delta){}_2F_1\left(1-n,k+3-2\delta,2-2\delta,1\right)}{\sum_{k=0}^{n+1}\mathscr{a}_k^{(1)}\Gamma(k+3-2\delta){}_2F_1\left(1-n,k+3-2\delta,2-2\delta,1\right)}. \tag{3.1.49}$$

The ${}_2F_1$ functions can be removed using the identity ${}_2F_1(1-n,b,c,1) = (c-b)_{n-1}/(c)_{n-1}$. This gives every coefficient and allows perturbative eigenstate corrections to be determined. Using this method, we find the first-order eigenstate correction and second-order energy correction

$$\begin{aligned}\tilde{\psi}_{1,1}&=\frac{1}{4}(1-\delta)\left((1-\delta)^2(2-\delta)(3-2\delta)-2z^2\right)\psi_n\mathcal{E}_0,\\ \tilde{E}_{1,2}&=-\frac{1}{4}(1-\delta)^4(2-\delta)(3-2\delta)\mathcal{E}_0^2,\end{aligned} \tag{3.1.50}$$

which correctly reduce to the Coulomb cases when $\delta = 0$. This generalizes to higher order, but requires the previous corrections. Finding the $\mathcal{P}_{n,2}$ polynomial, which is a polynomial of degree $n+3$, for instance requires substituting both ψ_n, $\tilde{\psi}_{n,1}$ into the $p = 2$ case of equation (3.1.6) and following a similar procedure. We have now computed the energy and eigenstate corrections for both methods. Before we apply these results, we first generalize the energy correction results and relate them to electrostatic polarizability.

3.2 Electrostatic Energy Corrections & Polarizability

We have at this point determined the corrections to the energy and eigenstate using both the Basis Expansion Method and the Dalgarno-Lewis approach. We now aim to generalize the energy correction results provided by the Dalgarno-Lewis approach and relate them to the electrostatic polarizabilities, which will prove useful later. We begin by examining the energy corrections, focusing on terms up to fourth order. It turns out that expressions for each energy correction can be found for a general n. This is a consequence of the general form of the eigenstate corrections found from the Dalgarno-Lewis approach and how the higher-order energy corrections are calculated using equation (3.1.8). We know that the eigenstate corrections take the form $\tilde{\psi}_{n,p} \propto z^{1-\delta} \mathrm{e}^{-z/n} \mathcal{P}_{n,p}(z)$ and that the energy corrections are found by integrating products of these corrections over the positive real axis. Using the substitution $z' = 2z/n$, it becomes clear that $\tilde{E}_{n,p}$ becomes a polynomial in n. This is a feature that we can exploit. Before doing so, we, for the sake of reducing calculations, simplify the energy correction formulas generated by equation (3.1.8). By employing the normalization conditions provided by equation (3.1.10) and generalizing equation (3.1.7), it turns out that we can express higher-order energy corrections exclusively in terms of lower-order eigenstates. In particular, instead of multiplying equation (3.1.6) through by the unperturbed eigenstate ψ_n and integrating over $\mathbb{R}_{\geq 0}$, we multiply through by $\tilde{\psi}_{n,q}$ for some $q \in \mathbb{Z}_{\geq 0}$ and integrate. The resulting equation then reads

$$\mathcal{H}^{q,p}_{n,n} + \tilde{\mathcal{H}}^{q,p-1}_{n,n} - \sum_{j=0}^{p} \tilde{E}_{n,j} s^{q,p-j}_{n,n} = 0. \tag{3.2.1}$$

By combining equation (3.2.1) for various parameters q, p and applying conjugation, we can construct a number of useful identities. We already have the zeroth- and first-order energy corrections for general n. Starting with the second-order energy correction then, the $p = 1$ case of equation (3.1.10) provides the normalization condition $s^{0,1}_{n,n} = 0$. In turn, the $p = 2$ case of equation (3.1.8) yields

$$\begin{aligned} \tilde{E}_{n,2} &= \tilde{\mathcal{H}}^{0,1}_{n,n} - \tilde{E}_{n,1} s^{0,1}_{n,n} \\ &= \langle \psi_n | z | \tilde{\psi}_{n,1} \rangle \mathcal{E}_0. \end{aligned} \tag{3.2.2}$$

We now move onto the third-order energy. We now employ the $p = 2$ case of equation (3.1.10), which reads $2\mathrm{Re} s^{0,2}_{n,n} + s^{1,1}_{n,n} = 0$. The $p = 3$ case of equation (3.1.8) then yields

$$\begin{aligned} \tilde{E}_{n,3} &= \tilde{\mathcal{H}}^{0,2}_{n,n} - \tilde{E}_{n,1} s^{0,2}_{n,n} - \tilde{E}_{n,2} s^{0,1}_{n,n} \\ &= \langle \psi_n | z | \tilde{\psi}_{n,2} \rangle \mathcal{E}_0 + \frac{1}{2} Z_{nn} \langle \tilde{\psi}_{n,1} | \tilde{\psi}_{n,1} \rangle \mathcal{E}_0, \end{aligned} \tag{3.2.3}$$

where we again have applied the normalization condition $s^{0,1}_{n,n} = 0$ and substituted the first-order energy correction $\tilde{E}_{n,1} = Z_{nn} \mathcal{E}_0$. To reduce further, we consider the $(q, p) = (2, 1)$ case and conjugate of the $(q, p) = (1, 2)$ case of equation (3.2.1), which read

$$\mathcal{H}^{2,1}_{n,n} + \tilde{\mathcal{H}}^{2,0}_{n,n} - E_n s^{2,1}_{n,n} - \tilde{E}_{n,1} s^{2,0}_{n,n} = 0, \tag{3.2.4}$$

$$\mathcal{H}^{2,1}_{n,n} + \tilde{\mathcal{H}}^{1,1}_{n,n} - E_n s^{2,1}_{n,n} - \tilde{E}_{n,1} s^{1,1}_{n,n} - \tilde{E}_{n,2} s^{0,1}_{n,n} = 0. \tag{3.2.5}$$

Subtracting equation (3.2.5) from equation (3.2.4) and reducing then yields

$$\tilde{\mathcal{H}}^{2,0}_{n,n} - \tilde{\mathcal{H}}^{1,1}_{n,n} + \tilde{E}_{n,1} s^{1,1}_{n,n} - \tilde{E}_{n,1} s^{2,0}_{n,n} = 0, \tag{3.2.6}$$

where we again have applied $s^{0,1}_{n,n} = 0$. By substituting the second-order normalization condition into equation (3.2.6) along with the first-order energy correction $\tilde{E}_{n,1} = Z_{nn} \mathcal{E}_0$, one eventually finds that

$$\langle \psi_n | z | \tilde{\psi}_{n,2} \rangle = \frac{1}{2} \langle \tilde{\psi}_{n,1} | 2z - 3Z_{nn} | \tilde{\psi}_{n,1} \rangle. \tag{3.2.7}$$

Substituting this relation back into equation (3.2.3) then yields

$$\tilde{E}_{n,3} = \langle \tilde{\psi}_{n,1} | z - Z_{nn} | \tilde{\psi}_{n,1} \rangle \mathcal{E}_0, \tag{3.2.8}$$

which only requires first-order information to compute. Finally, we turn to the fourth-order energy correction. Here, we require the $(q, p) = (0, 4)$, $(q, p) = (1, 3)$ and $(q, p) = (3, 1)$ cases of equation (3.2.1), which take the forms

$$\mathcal{H}^{0,4}_{n,n} + \tilde{\mathcal{H}}^{0,3}_{n,n} - E_n s^{0,4}_{n,n} - \tilde{E}_{n,1} s^{0,3}_{n,n} - \tilde{E}_{n,2} s^{0,2}_{n,n} - \tilde{E}_{n,3} s^{0,1}_{n,n} - \tilde{E}_{n,4} s^{0,0}_{n,n} = 0, \tag{3.2.9}$$

$$\mathcal{H}^{1,3}_{n,n} + \tilde{\mathcal{H}}^{1,2}_{n,n} - E_n s^{1,3}_{n,n} - \tilde{E}_{n,1} s^{1,2}_{n,n} - \tilde{E}_{n,2} s^{1,1}_{n,n} - \tilde{E}_{n,3} s^{1,0}_{n,n} = 0, \tag{3.2.10}$$

$$\mathcal{H}^{3,1}_{n,n} + \tilde{\mathcal{H}}^{3,0}_{n,n} - E_n s^{3,1}_{n,n} - \tilde{E}_{n,1} s^{3,0}_{n,n} = 0. \tag{3.2.11}$$

We now subtract the conjugate of equation (3.2.11) from equation (3.2.10), which in turn yields

$$\tilde{\mathcal{H}}^{1,2}_{n,n} - \tilde{\mathcal{H}}^{0,3}_{n,n} + \tilde{E}_{n,1} s^{0,3}_{n,n} - \tilde{E}_{n,1} s^{1,2}_{n,n} - \tilde{E}_{n,2} s^{1,1}_{n,n} = 0, \tag{3.2.12}$$

where we have used $s^{0,1}_{n,n} = 0$ once more. We now recall that $\mathcal{H}^{0,p}_{n,n} = E_n s^{0,p}_{n,n}$ because $\hat{H}_0$ is Hermitian. Therefore, adding equations (3.2.9) and (3.2.12) reduces to

$$\tilde{\mathcal{H}}^{1,2}_{n,n} - \tilde{E}_{n,1} s^{1,2}_{n,n} - \tilde{E}_{n,2} s^{1,1}_{n,n} - \tilde{E}_{n,2} s^{0,2}_{n,n} - \tilde{E}_{n,4} = 0, \tag{3.2.13}$$

where we have used $s^{0,0}_{n,n} = 1$ and $s^{0,1}_{n,n} = 0$. Finally, applying the normalization condition $2\mathrm{Re} s^{0,2}_{n,n} + s^{1,1}_{n,n} = 0$, the first-order energy correction $\tilde{E}_{n,1} = Z_{nn}\mathcal{E}_0$ and solving for the fourth-order energy correction, we find

$$\tilde{E}_{n,4} = \langle \tilde{\psi}_{n,1} \,| z - Z_{nn} |\, \tilde{\psi}_{n,2} \rangle \mathcal{E}_0 - \frac{1}{2} \tilde{E}_{n,2} \langle \tilde{\psi}_{n,1} | \tilde{\psi}_{n,1} \rangle, \tag{3.2.14}$$

which only requires second-order information. We may now proceed and compute the energy corrections using equations (3.2.2), (3.2.8) and (3.2.14) for the Kratzer-like potential by applying the methods described in the previous section. The second-order energy correction for the first six states becomes

$$\begin{aligned}
\tilde{E}_{1,2} &= -\frac{1}{4}(1-\delta)^2(2\delta^4 - 11\delta^3 + 22\delta^2 - 19\delta + 6)\mathcal{E}_0^2 \\
\tilde{E}_{2,2} &= -\frac{1}{4}(2-\delta)^2(2\delta^4 - 25\delta^3 + 85\delta^2 - 122\delta + 66)\mathcal{E}_0^2, \\
\tilde{E}_{3,2} &= -\frac{1}{4}(3-\delta)^2(2\delta^4 - 39\delta^3 + 190\delta^2 - 393\delta + 306)\mathcal{E}_0^2, \\
\tilde{E}_{4,2} &= -\frac{1}{4}(4-\delta)^2(2\delta^4 - 53\delta^3 + 337\delta^2 - 916\delta + 936)\mathcal{E}_0^2, \\
\tilde{E}_{5,2} &= -\frac{1}{4}(5-\delta)^2(2\delta^4 - 67\delta^3 + 526\delta^2 - 1775\delta + 2250)\mathcal{E}_0^2, \\
\tilde{E}_{6,2} &= -\frac{1}{4}(6-\delta)^2(2\delta^4 - 81\delta^3 + 757\delta^2 - 3054\delta + 4626)\mathcal{E}_0^2.
\end{aligned} \tag{3.2.15}$$

We may now exploit the fact that the energy corrections are polynomials in the principle quantum number. It is clear from the expressions in equation (3.2.15) that the second-order energy for a general n takes the form

$$\tilde{E}_{n,2} = -\frac{1}{4}n^2 \left(a_n^{(4)}\delta^4 - a_n^{(3)}\delta^3 + a_n^{(2)}\delta^2 - a_n^{(1)}\delta + a_n^{(0)} \right) \mathcal{E}_0^2, \tag{3.2.16}$$

where $\{a_n^{(k)}\}$ are some coefficients. Because $\tilde{E}_{n,2}$ is a polynomial in n, then the sequences $\{a_n^{(k)}\}_{n\geq 1}$ for $0 \leq k \leq 4$ are polynomial sequences. This means that a general expression for each sequence can be determined using Newton-Gregory forward interpolation, see [Burden et al., 2015], which allows us to construct the unique polynomial generating the sequences $\{a_n^{(k)}\}$. To this end, we define the k'th forward-difference $\Delta_n^{(k)}$ of a sequence S is defined $\Delta_n^{(k)} \equiv \Delta_{n+1}^{(k-1)} - \Delta_n^{(k-1)}$ for $k \geq 1$ with $\Delta_n^{(0)} = s_n$. For some polynomial sequence $S = \{s_n\}_{n\geq 1}$ of degree d, the Newton-Gregory forward interpolation asserts that

$$s_n = \sum_{k=0}^{d} \Delta_1^{(k)} \binom{n-1}{k}. \tag{3.2.17}$$

The degree of the polynomials sequence is not known beforehand, so in this case we simply sum to the smallest d for which $\Delta_1^{(d)} = 0$. As an example, we apply the algorithm to the $a_n^{(3)}$ coefficient for the second-order energy. Here, we identify $\Delta_1^{(0)} = 11$, $\Delta_1^{(1)} = 14$ and $\Delta_1^{(2)} = 0$. This gives

$$\begin{aligned}
a_n^{(3)} &= \sum_{k=0}^{1} \Delta_1^{(k)} \binom{n-1}{k} \\
&= 14n - 3,
\end{aligned} \tag{3.2.18}$$

which correctly reproduces the listed coefficients. Repeating the procedure for the other coefficients gives $a_n^{(2)} = 21n^2 + 1$, $a_n^{(1)} = 14n^3 + 5n$ and $a_n^{(0)} = 7n^4/2 + 5n^2/2$. Clearly, $a_n^{(4)} = 2$. Substituting the coefficients back into equation (3.2.16) and reducing then yields

$$\begin{aligned}\tilde{E}_{n,2} &= -\frac{1}{4}n^2\left(2\delta^4 - (14n-3)\delta^3 + (21n^2+1)\delta^2 - (14n^3+5n)\delta + \frac{7}{2}n^4 + \frac{5}{2}n^2\right)\mathcal{E}_0^2 \\ &= -\frac{1}{8}n^2\left(7n^4 + 5n^2 - 3\delta^2(1-\delta)^2\right)\mathcal{E}_0^2, \qquad (3.2.19)\end{aligned}$$

which gives the general formula for the second-order energy. It reduces to $\tilde{E}_{n,2} = -(7n^6 + 5n^4)\mathcal{E}_0^2/8$ in the $\delta = 0$ case. We now repeat the procedure for the third-order and fourth-order energy corrections. We start by computing the first few corrections. The third-order and fourth-order corrections of the ground state are

$$\tilde{E}_{n,3} = \frac{1}{16}(1-\delta)^4\left(16\delta^6 - 140\delta^5 + 498\delta^4 - 922\delta^3 + 938\delta^2 - 498\delta + 108\right)\mathcal{E}_0^3, \qquad (3.2.20)$$

$$\begin{aligned}\tilde{E}_{n,4} = \frac{1}{64}(1-\delta)^6\big(&192\delta^8 - 2340\delta^7 + 12226\delta^6 - 35754\delta^5 + 64016\delta^4 - 71896\delta^3 \qquad (3.2.21)\\ &+ 49506\delta^2 - 19130\delta + 3180\big)\mathcal{E}_0^4.\end{aligned}$$

By computing higher-order terms, we find the general pattern

$$\tilde{E}_{n,3} = \frac{1}{16}n^4\left(a_n^{(6)}\delta^6 - a_n^{(5)}\delta^5 + a_n^{(4)}\delta^4 - a_n^{(3)}\delta^3 + a_n^{(2)}\delta^2 - a_n^{(1)}\delta + a_n^{(0)}\right)\mathcal{E}_0^3 \qquad (3.2.22)$$

$$\tilde{E}_{n,4} = -\frac{1}{64}n^6\left(a_n^{(8)}\delta^8 - a_n^{(7)}\delta^7 + a_n^{(6)}\delta^6 - a_n^{(5)}\delta^5 + a_n^{(4)}\delta^4 - a_n^{(3)}\delta^3 + a_n^{(2)}\delta^2 - a_n^{(1)}\delta + a_n^{(0)}\right)\mathcal{E}_0^4 \quad (3.2.23)$$

Applying equation (3.2.17) to each set of coefficients and substituting them back into equations (3.2.22) and (3.2.23), we eventually find

$$\tilde{E}_{n,3} = \frac{1}{16}n^4\left(33n^6 + 75n^4 - 7n^2\delta^2(1-\delta)^2 + 10\delta^3(1-\delta)^3\right)\mathcal{E}_0^3, \qquad (3.2.24)$$

$$\begin{aligned}\tilde{E}_{n,4} = -\frac{1}{64}n^6\Big(&465n^8 + 2275n^6 + 440n^4 - 99n^4\delta^2(1-\delta)^2 - 90n^2\delta^2(1-\delta)^2 \qquad (3.2.25)\\ &\times(\delta^2 - \delta + 2) - 84\delta^4(1-\delta)^4\Big)\mathcal{E}_0^4,\end{aligned}$$

which reduces to $\tilde{E}_{n,3} = (33n^{10} + 75n^8)\mathcal{E}_0^3/16$ and $\tilde{E}_{n,4} = -(465n^{14} + 2275n^{12} + 440n^{10})\mathcal{E}_0^4/64$ in the case of the Coulomb potential. The $\delta = 0$ case of these corrections agree with the findings of [Pedersen, 2024]. The procedure can be continued for higher-order corrections, although the presented method quickly becomes computationally expensive. Higher-order terms will not be needed for our analysis later on, so we move forward from here. We now aim to calculate the polarizability of the qubit electron system. When the qubit electron is perturbed by an external electric field, the wave function is distorted accordingly and a dipole moment $\mathbf{p}$ is induced. The strength of this response is quantified by the electric polarizability α, which is a measure for how strongly an atom or molecule reacts to an externally applied electric field. In general, the system's response to an external electric field can exhibit complex behaviour. For weaker field strengths, the response remains predominantly linear; however, as the field strength increases sufficiently, non-linear effects inevitably show and become important. By decomposing the system's full response into zeroth-, first-, second-order terms, and so on - similar to our approach with the perturbed eigenstates - we can associate a polarizability with each order. This enables us to characterize each component of the full response, such as the linear or quadratic terms, which may be of particular interest. To derive an expression for the polarizability, we start with the dipole moment $\mathbf{p} = -ez\hat{\mathbf{z}}$. When treating the dipole moment quantum mechanically, we instead use the expectation value of the coordinate z, which corresponds to the diagonal matrix elements Z_{nn}. In atomic units therefore, the dipole moment of the perturbed electron qubit system in state n reads

$$\tilde{\mathbf{p}}_n = -\langle\tilde{\psi}_n\,|z|\,\tilde{\psi}_n\rangle\hat{\mathbf{z}}. \tag{3.2.26}$$

As a means to extract the polarizability associated with each order of the perturbation, we now apply the same approach used in the start of this chapter to derive the corrections to the eigenstates due to the perturbation. In particular, we substitute the perturbative expansion of the eigenstates given in equation (3.1.3) into the dipole moment expression above, which in turn gives

$$\begin{aligned}\tilde{\mathbf{p}}_n &= -\langle\tilde{\psi}_{n,0} + \tilde{\psi}_{n,1} + \tilde{\psi}_{n,2} + \ldots\,|z|\,\tilde{\psi}_{n,0} + \tilde{\psi}_{n,1} + \tilde{\psi}_{n,2} + \ldots\rangle\hat{\mathbf{z}} \\ &= -\langle\tilde{\psi}_{n,0}\,|z|\,\tilde{\psi}_{n,0}\rangle\hat{\mathbf{z}} - \left(\langle\tilde{\psi}_{n,0}\,|z|\,\tilde{\psi}_{n,1}\rangle + \langle\tilde{\psi}_{n,1}\,|z|\,\tilde{\psi}_{n,0}\rangle\right)\hat{\mathbf{z}} - \ldots \\ &= -\tilde{\mathbf{p}}_{n,0} - \tilde{\mathbf{p}}_{n,1} - \tilde{\mathbf{p}}_{n,2} - \ldots,\end{aligned} \tag{3.2.27}$$

where we have gathered terms of equal order and defined the p'th correction to the dipole moment as

$$\tilde{\mathbf{p}}_{n,p} = -\sum_{j=0}^{p}\langle\tilde{\psi}_{n,j}\,|z|\,\tilde{\psi}_{n,p-j}\rangle\hat{\mathbf{z}}. \tag{3.2.28}$$

To extract the polarizability from the dipole moment, we must first consider the polarization $\tilde{\mathbf{P}}_n = \mathcal{N}\tilde{\mathbf{p}}_n$, where $\mathcal{N}$ is the number density. The polarization is traditionally expressed as a power series expansion in the applied electric field $\mathcal{E}$, which in this case and for regular units reads

$$\tilde{\mathbf{P}}_n = \sum_{j=0}^{\infty}\varepsilon_0\tilde{\chi}_{n,j}\mathcal{E}_0^j\hat{\mathbf{z}}, \tag{3.2.29}$$

where ε_0 is the vacuum permitivity and $\tilde{\chi}_{n,p}$ is the p'th order electric susceptibility. We now substitute the polarization $\tilde{\mathbf{P}}_n = \mathcal{N}\tilde{\mathbf{p}}_n$ along with the dipole moment expansion given in equation (3.2.27) into the left-hand side of equation (3.2.29). Dividing the resulting expression through by the number density $\mathcal{N}$ then yields

$$\begin{aligned}\sum_{j=0}^{\infty}\tilde{\mathbf{p}}_{n,j} &= \sum_{j=0}^{\infty}\varepsilon_0\mathcal{N}^{-1}\tilde{\chi}_{n,j}\mathcal{E}_0^j\hat{\mathbf{z}} \\ &= \sum_{j=0}^{\infty}\tilde{\alpha}_{n,0}\mathcal{E}_0^j\hat{\mathbf{z}},\end{aligned} \tag{3.2.30}$$

where we have defined the p'th order polarizability $\tilde{\alpha}_{n,p} = \varepsilon_0\mathcal{N}^{-1}\tilde{\chi}_{n,p}$, which is associated with state n. By realizing that $|\tilde{\mathbf{p}}_{n,p}| \propto \mathcal{E}_0^p$, we may group terms of equal powers in the electric field and hence extract the polarizability for every order. Returning to atomic units, we find

$$\tilde{\alpha}_{n,p}\mathcal{E}_0^p = -\sum_{j=0}^{p}\langle\tilde{\psi}_{n,j}\,|z|\,\tilde{\psi}_{n,p-j}\rangle. \tag{3.2.31}$$

The zeroth-order response $\tilde{\alpha}_{n,0}$ is called the permanent dipole moment and is relatively easily calculated using the matrix elements of the unperturbed eigenstates, Z_{nn}. For the quantum defect atom, the diagonal matrix elements are given by equation (2.2.21) and hence the permanent dipole moments becomes

$$\begin{aligned}\tilde{\alpha}_{n,0} &= -\langle\tilde{\psi}_{n,0}\,|z|\,\tilde{\psi}_{n,0}\rangle \\ &= -Z_{nn} \\ &= -\frac{1}{2}\left(3n^2 + \delta - \delta^2\right),\end{aligned} \tag{3.2.32}$$

which reduces to $\tilde{\alpha}_{n,0} = -3n^2/2$ for the Coulomb potential. We note that $\tilde{\alpha}_{n,0}\mathcal{E}_0 = -\tilde{E}_{n,1}$. In the ground state and first excited state of the Coulomb potential therefore, we have $\tilde{\alpha}_{1,0} = -3/2$ and $\tilde{\alpha}_{2,0} = -6$. The next order term $\tilde{\alpha}_{n,1}$ is the so-called electric polarizability and corresponds to the linear response due to the applied field. In atomic units, the polarizability is measured in $\overline{\mathrm{Ha}}/\overline{\mathcal{E}}_0^2$, which in the case of Helium and Neon yields 43.97 meVμm/V^2 and 0.186 meVμm/V^2. Using equation (3.2.31) for the $p = 1$ case, we find

$$\begin{aligned}\tilde{\alpha}_{n,1}\mathcal{E}_0 &= -\langle\tilde{\psi}_{n,0}\,|z|\,\tilde{\psi}_{n,1}\rangle - \langle\tilde{\psi}_{n,1}\,|z|\,\tilde{\psi}_{n,0}\rangle \\ &= -2\mathrm{Re}\langle\tilde{\psi}_{n,0}\,|z|\,\tilde{\psi}_{n,1}\rangle. \end{aligned} \tag{3.2.33}$$

Comparing with equation (3.2.2), we find $\tilde{\alpha}_{n,1}\mathcal{E}_0^2 = -2\tilde{E}_{n,2}$. It is then clear that, for a general n, we have

$$\tilde{\alpha}_{n,1} = \frac{1}{4}n^2\left(7n^4 + 5n^2 - 3\delta^2(1-\delta)^2\right). \tag{3.2.34}$$

This reduces to $\tilde{\alpha}_{n,1} = \left(7n^6 + 5n^4\right)/4$ in the Coulomb case. In addition, by employing the basis expansion method and substituting equation (3.1.24), we see that $\tilde{\alpha}_{n,1} = \mathcal{G}_{n,-2}$. For the ground state and first excited state of the Coulomb potential, this evaluates to $\mathcal{G}_{1,-2} = 2.99999990620281$ and $\mathcal{G}_{2,-2} = 131.999994502542$, which clearly agrees with equation (3.2.34). The final two corrections, which we will consider in this section, are the first and second hyperpolarizabilities $\tilde{\alpha}_{n,2}$ and $\tilde{\alpha}_{n,3}$. Starting with the first hyperpolarizability provided by the $p = 2$ case of equation (3.2.31), we may use the identity given in equation (3.2.7) to write

$$\begin{aligned}\tilde{\alpha}_{n,2}\mathcal{E}_0^2 &= -\langle\psi_n\,|z|\,\tilde{\psi}_{n,2}\rangle - \langle\tilde{\psi}_{n,1}\,|z|\,\tilde{\psi}_{n,1}\rangle - \langle\tilde{\psi}_{n,2}\,|z|\,\psi_n\rangle \\ &= -2\mathrm{Re}\langle\psi_n\,|z|\,\tilde{\psi}_{n,2}\rangle - \langle\tilde{\psi}_{n,1}\,|z|\,\tilde{\psi}_{n,1}\rangle \\ &= -3\langle\tilde{\psi}_{n,1}\,|z - Z_{nn}|\,\tilde{\psi}_{n,1}\rangle, \end{aligned} \tag{3.2.35}$$

from which we can compare with equation (3.2.8) and deduce that $\tilde{\alpha}_{n,2}\mathcal{E}_0^3 = -3\tilde{E}_{n,3}$. By using the $p = 3$ case of equation (3.2.31), the $p = 4$ case of equation (3.1.8) along with the normalization conditions, a similar calculation shows that $\tilde{\alpha}_{n,3}\mathcal{E}_0^4 = -4\tilde{E}_{n,4}$. It is then clear that

$$\tilde{\alpha}_{n,2} = -\frac{3}{16}n^4\left(33n^6 + 75n^4 - 7n^2\delta^2(1-\delta)^2 + 10\delta^3(1-\delta)^3\right), \tag{3.2.36}$$

$$\begin{aligned}\tilde{\alpha}_{n,3} = \frac{1}{16}n^6\Big(&465n^8 + 2275n^6 + 440n^4 - 99n^4\delta^2(1-\delta)^2 - 90n^2\delta^2(1-\delta)^2 \\ &\times(\delta^2 - \delta + 2) - 84\delta^4(1-\delta)^4\Big),\end{aligned} \tag{3.2.37}$$

which reduces to $\tilde{\alpha}_{n,2} = -(99n^{10} + 225n^8)/16$ and $\tilde{\alpha}_{n,3} = (465n^{14} + 2275n^{12} + 440n^{10})/16$ in the case of the Coulomb potential. For the ground state and first excited state of the Coulomb potential, we find $\tilde{\alpha}_{1,2} = -81/4$, $\tilde{\alpha}_{2,2} = 9936$, $\tilde{\alpha}_{1,3} = 795/4$ and $\tilde{\alpha}_{2,3} = 1086720$. Again, this may be repeated to higher order if desired. We have now generalized the energy corrections and related them to the polarizabilities, and so we can move forward and apply the results.

3.3 Hypergeometric Approximants

We have at this point determined the corrections to the energy and eigenstate using both the Basis Expansion Method and the Dalgarno-Lewis approach. In addition, we have generalized the energy correction results. We now aim to apply them by modelling how the energies of the system are Stark-shifted in response to the external field. We do this using a numerical approach, that will be presented in the next, and an approximate and analytical approach, that will be presented now. In section 3.1.2, we used the Dalgarno-Lewis approach to compute exact energy corrections, which we later generalized in section 3.2. In particular, we computed the first few energy corrections of the Coulomb ground state explicitly, as shown in equations (3.1.38) and (3.1.39). Substituting these corrections into the original perturbative expansion yields

$$\tilde{E}_1 = -\frac{1}{2} + \frac{3}{2}\mathcal{E}_0 - \frac{3}{2}\mathcal{E}_0^2 + \frac{27}{4}\mathcal{E}_0^3 - \frac{795}{16}\mathcal{E}_0^4 + \frac{3843}{8}\mathcal{E}_0^5 - 5583\mathcal{E}_0^6 + \dots . \tag{3.3.1}$$

The expression for the perturbed energy $\tilde{E}_n$ found in equation (3.3.1) is a power series in $\mathcal{E}_0$. Unfortunately, it is clear that the perturbative expansion has radius of convergence zero around its expansion point $\mathcal{E}_0 = 0$. As such, the expansion given in equation (3.3.1) cannot be used as a means of calculating the Stark shift. Despite this, we are still able to construct a sound model based on the calculated corrections. One way of doing this is to regularize the sum by means of a hypergeometric approximant, an approach that has been applied to a several other problems, see for instance [Pedersen, 2023], [Mera et al., 2015] and [Pedersen, 2024]. The idea is to assume that the perturbed energy can be approximated by a hypergeometric function ${}_pF_q$. Although the procedure can be generalized to hypergeometric functions with a larger number of parameters or even Meijer G-functions, see [Mera et al., 2018], we restrict ourselves to the hypergeometric ${}_2F_1$ approximant. If we write the perturbed energy as a function of the field strength, $\tilde{E}_n(\mathcal{E}_0)$, then the most general approximant, which is linear in argument and thus matches the original perturbation, is

$$\tilde{E}_n \approx c_1 {}_2F_1\left(c_2, c_3, c_4, c_5\mathcal{E}_0\right), \tag{3.3.2}$$

where c_1, c_2, c_3, c_4 and c_5 are a set of coefficients. A crucial benefit of the hypergeometric approximant given in equation (3.3.2) is that it acquires a non-zero imaginary part, which in turn allows the full complex energy eigenvalues $\tilde{E}_n - i\Gamma_n/2$, see [Landau and Lifshitz, 1965], to be determined, where Γ_n is the ionization rate associated with state n. The is a consequence of the branch point of ${}_2F_1(c_2, c_3, c_4, c_5\mathcal{E}_0)$ lying at $\mathcal{E}_0 = c_5^{-1}$, which when crossed from below along the real-axis assigns the perturbed energy an imaginary part corresponding to $-i\Gamma_n/2$. With the hypergeometric function given in equation (3.3.2) as a starting point, the remaining task is then to determine these coefficients as to accurately model the perturbed energy. One way of doing this is to choose the coefficients so that the Taylor series of the approximant matches the original perturbative expansion given in equation (3.1.3). To this end, we recall that $\tilde{E}_{n,p} \propto \mathcal{E}_0^p$. We have five coefficients and hence we can at most match up to fourth order so that $\tilde{E}_n = c_1 {}_2F_1(c_2, c_3, c_4, c_5\mathcal{E}_0) + O(\mathcal{E}_0^5)$. If we expand the perturbed energy given in equation (3.3.2) in a Taylor series around the point $\mathcal{E}_0 = 0$ and use the differentiation rule for the hypergeometric function given in equation (A.0.7), we find that

$$\begin{aligned} \tilde{E}_n &\approx \sum_{k=0}^{\infty} \frac{\mathrm{d}^k \tilde{E}_n}{\mathrm{d}\mathcal{E}_0^k} \frac{\mathcal{E}_0^k}{k!} \\ &= c_1 + c_1 \frac{c_2 c_3}{c_4} c_5 \mathcal{E}_0 + c_1 \frac{c_2(c_2+1) c_3(c_3+1)}{2c_4(c_4+1)} c_5^2 \mathcal{E}_0^2 + \dots . \end{aligned} \tag{3.3.3}$$

We clearly want the approximant to return the unperturbed energy once the strength goes to zero. Looking at equation (3.3.3), this is only possible if $c_1 = -1/2n^2$ is the unperturbed energy. We now match the next four terms by comparing with the original perturbative expansion. This gives

$$\begin{aligned} \tilde{E}_{n,1} &= -\frac{c_2 c_3}{2n^2 c_4} c_5 \mathcal{E}_0, \\ \tilde{E}_{n,2} &= -\frac{c_2(c_2+1) c_3(c_3+1)}{4n^2 c_4(c_4+1)} c_5^2 \mathcal{E}_0^2, \\ \tilde{E}_{n,3} &= -\frac{c_2(c_2+1)(c_2+2) c_3(c_3+1)(c_3+2)}{12n^2 c_4(c_4+1)(c_4+2)} c_5^3 \mathcal{E}_0^3, \\ \tilde{E}_{n,4} &= -\frac{c_2(c_2+1)(c_2+2)(c_2+3) c_3(c_3+1)(c_3+2)(c_3+3)}{48n^2 c_4(c_4+1)(c_4+2)(c_4+3)} c_5^4 \mathcal{E}_0^4. \end{aligned} \tag{3.3.4}$$

The conditions can be simplified by substituting one into another. In particular, by substituting the first equation into the second, second into the third and so on, we find

$$
\begin{aligned}
\frac{\tilde{E}_{n,1}}{E_n} &= \frac{\mathscr{c}_2\mathscr{c}_3}{\mathscr{c}_4}\mathscr{c}_5\mathcal{E}_0, \\
\frac{\tilde{E}_{n,2}}{\tilde{E}_{n,1}} &= \frac{(\mathscr{c}_2+1)(\mathscr{c}_3+1)}{2(\mathscr{c}_4+1)}\mathscr{c}_5\mathcal{E}_0, \\
\frac{\tilde{E}_{n,3}}{\tilde{E}_{n,2}} &= \frac{(\mathscr{c}_2+2)(\mathscr{c}_3+2)}{3(\mathscr{c}_4+2)}\mathscr{c}_5\mathcal{E}_0, \\
\frac{\tilde{E}_{n,4}}{\tilde{E}_{n,3}} &= \frac{(\mathscr{c}_2+3)(\mathscr{c}_3+3)}{4(\mathscr{c}_4+3)}\mathscr{c}_5\mathcal{E}_0.
\end{aligned}
\tag{3.3.5}
$$

All there is to do now is to substitute the energy corrections from the previous section and solve the system of equations for each coefficient, which is easily done in Python or Mathmatica. Taking the ground state of the Coulomb potential as an example, we see from equation (3.3.1) that $\tilde{E}_{1,1} = 3\mathcal{E}_0/2$, $\tilde{E}_{1,2} = -3\mathcal{E}_0^2/2$, $\tilde{E}_{1,3} = 27\mathcal{E}_0^3/4$ and $\tilde{E}_{1,4} = -795\mathcal{E}_0^4/16$. Substituting these corrections into equation (3.3.5) then yields

$$
\begin{aligned}
-3\mathcal{E}_0 &= \frac{\mathscr{c}_2\mathscr{c}_3}{\mathscr{c}_4}\mathscr{c}_5\mathcal{E}_0, \\
-\mathcal{E}_0 &= \frac{(\mathscr{c}_2+1)(\mathscr{c}_3+1)}{2(\mathscr{c}_4+1)}\mathscr{c}_5\mathcal{E}_0, \\
-\frac{9}{2}\mathcal{E}_0 &= \frac{(\mathscr{c}_2+2)(\mathscr{c}_3+2)}{3(\mathscr{c}_4+2)}\mathscr{c}_5\mathcal{E}_0, \\
-\frac{265}{36}\mathcal{E}_0 &= \frac{(\mathscr{c}_2+3)(\mathscr{c}_3+3)}{4(\mathscr{c}_4+3)}\mathscr{c}_5\mathcal{E}_0.
\end{aligned}
\tag{3.3.6}
$$

Solving the system of equations for each coefficient and using the ground state energy $E_1 = -1/2$ then shows that the corresponding hypergeometric approximant becomes

$$
\tilde{E}_1 \approx -\frac{1}{2}{}_2F_1\left(-\frac{1247+5\sqrt{3817}}{2534}, -\frac{1247-5\sqrt{3817}}{2534}, \frac{48}{29}, -\frac{1267}{58}\mathcal{E}_0\right). \tag{3.3.7}
$$

Testing shows that the approximant indeed does acquire an imaginary part once the critical field strength $\mathcal{E}_0 = -58/1267$ is passed. This is despite the fact that the coefficients are determined exclusively from the first five terms in the perturbative expansion, which are all real-valued. That said, this does highlight a minor issue with the chosen approximant in that it does not accurately reproduce the imaginary part for small field strengths. From a physical point of view, we expect tunnelling to occur whenever the electric field points in towards the substrate, meaning $\mathcal{E}_0 < 0$, but this is not reproduced by the chosen approximant. A slightly modified approximant can be chosen that alleviates this issue, see [Pedersen et al., 2016]. For the problem at hand however, this is not strictly necessary because the critical field strength is relatively small and decreases further with increasing n, although the issue is made worse with more strongly bound Kratzer potentials. We can repeat the procedure for the excited states, the approximants for the first four of which become

$$\tilde{E}_2 \approx -\frac{1}{8}{}_2F_1\left(\frac{-501539-5i\sqrt{76756895}}{946058}, \frac{-501539+5i\sqrt{76756895}}{946058}, \frac{11163}{8009}, -\frac{1892116}{8009}\mathcal{E}_0\right), \tag{3.3.8}$$

$$\tilde{E}_3 \approx -\frac{1}{18}{}_2F_1\left(-\frac{24240121+5i\sqrt{291559959335}}{45863962}, -\frac{24240121-5i\sqrt{291559959335}}{45863962}, \frac{40032}{36101}, -\frac{68795943}{72202}\mathcal{E}_0\right), \tag{3.3.9}$$

$$\tilde{E}_4 \approx -\frac{1}{32}{}_2F_1\left(-\frac{503957+5\sqrt{209041297}}{971454}, -\frac{503957-5\sqrt{209041297}}{971454}, \frac{32007}{35561}, -\frac{93259584}{35561}\mathcal{E}_0\right), \tag{3.3.10}$$

$$\tilde{E}_5 \approx -\frac{1}{50}{}_2F_1\left(-\frac{51689+\sqrt{143953681}}{101274}, -\frac{51689-\sqrt{143953681}}{101274}, \frac{1248}{1609}, -\frac{18988875}{3218}\mathcal{E}_0\right), \tag{3.3.11}$$

where we notice the rapidly increasing $\mathscr{c}_5$ and hence vanishing critical fields strength. This procedure of finding hypergeometric approximants is identical for the Kratzer-like potential, although the expressions quickly become very complicated, see for instance section E. The ground state is manageable however and will be presented here. Using the energy correction formulas from the previous section for the ground state of the Kratzer-like potential, we can show that perturbative expansion becomes

$$\begin{aligned}\tilde{E}_1 = &-\frac{1}{2(1-\delta)^2}+\frac{1}{2}(1-\delta)(3-2\delta)\mathcal{E}_0-\frac{1}{4}(1-\delta)^4(2-\delta)(3-2\delta)\mathcal{E}_0^2 \\ &+\frac{1}{8}(1-\delta)^7(2-\delta)(3-2\delta)(9-4\delta)\mathcal{E}_0^3-\frac{1}{32}(1-\delta)^{10}(2-\delta)(3-2\delta)(48\delta^2-225\delta+265)\mathcal{E}_0^4+\dots\end{aligned} \tag{3.3.12}$$

We now proceed in identical fashion to the previous example. If we define

$$\mathscr{c}_0 = \sqrt{53248\delta^6-592384\delta^5+2716560\delta^4-6615248\delta^3+9148284\delta^2-6985900\delta+2385625}, \tag{3.3.13}$$

it is then possible to show that the remaining coefficients can be written as

$$\begin{aligned}\mathscr{c}_2 &= \frac{-512\delta^3+3604\delta^2-8306\delta+6235-\mathscr{c}_0}{2(544\delta^3-3716\delta^2+8428\delta-6335)}, \\ \mathscr{c}_3 &= \frac{-512\delta^3+3604\delta^2-8306\delta+6235+\mathscr{c}_0}{2(544\delta^3-3716\delta^2+8428\delta-6335)}, \\ \mathscr{c}_4 &= \frac{12(2\delta^2-13\delta+20)}{40\delta^2-156\delta+145}, \\ \mathscr{c}_5 &= \frac{(1-\delta)^3(544\delta^3-3716\delta^2+8428\delta-6335)}{2(40\delta^2-156\delta+145)}.\end{aligned} \tag{3.3.14}$$

We note that the coefficients reduce to those given in equation (3.3.7) once $\delta = 0$. With $\delta = 0.1$ for instance, we find that $\mathscr{c}_5^{-1} \approx -0.064$, which is a roughly 29% increase from the Coulomb case. This concludes the presented analytical approach and we now move onto the numerical approach in the section.

3.4 Sturmian Basis Expansion

As a means of testing the accuracy of the hypergeomatric approximants, we compare them with a numerical approach. The numerical method we choose to solve the problem at hand is a complex-scaling basis expansion approach. The complex scaling method was originally extended to Stark-like problems by Herbst and Simon, see [Herbst and Simon, 1978], and, like the hypergeometric approximants, allows us to determine the full complex energy eigenvalues $\tilde{E}_n - i\Gamma_n/2$. The main idea behind the complex scaling method is that we analytically continue the perturbed problem into the complex plane. To show how this is accomplished in practice, we turn to the perturbed problem we are trying to solve, which for the Kratzer-like potential reads

$$\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{1}{z} - \frac{\delta(1-\delta)}{2z^2} + \mathcal{E}_0 z\right)\tilde{\psi}_n = \tilde{E}_n\tilde{\psi}_n. \tag{3.4.1}$$

The analytical continuation of this problem now involves applying the coordinate transformation $z \longrightarrow z\mathrm{e}^{i\theta}$, which rotates the problem into the complex plane by an angle θ, which we call the complex scaling angle. In full generality, the coordinates may also be scaled by some real factor. Applying the coordinate transformation to the perturbed problem then yields

$$\left(-\frac{1}{2}\mathrm{e}^{-2i\theta}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \mathrm{e}^{-i\theta}\frac{1}{z} - \mathrm{e}^{-2i\theta}\frac{\delta(1-\delta)}{2z^2} + \mathrm{e}^{i\theta}\mathcal{E}_0 z\right)\tilde{\psi}_n = \tilde{E}_n\tilde{\psi}_n. \tag{3.4.2}$$

The full Hamiltonian of the perturbed problem is no longer Hermitian and hence the energy eigenvalues become complex. The key result of the complex scaling method is that the newly acquired imaginary part exactly corresponds to the ionization rate term $-i\Gamma_n/2$. In this way, solving equation (3.4.2) numerically then provides both the energy and ionization rate. We choose a truncated basis expansion as the numerical approach. This involves expanding the perturbed eigenstates in terms of some appropriate basis, which we then truncate at some finite N. This in turn allows a generalized eigenvalue problem to be constructed and solved. The only remaining task is then to choose a fitting basis for this task. The basis formed by the unperturbed eigenstates are ill-suited for this purpose because part of their spectrum is continuous. This is exemplified by the oscillator strength calculations given in section 2.1.1, which showed that the discrete spectrum of the Coulomb potential only contributes roughly 33% to the TRK sum rule. Even the $\delta = -0.8$ case of the quantum defect atom only has a roughly 60% contribution from the discrete spectrum. We therefore opt for a different basis, which ideally is complete and discrete. A basis of this kind can be constructed by returning to the unperturbed problem, whose discrete spectrum is associated with the set of negative energy eigenvalues. The idea is to fix the energy to $E = -\kappa^2/2$, where the Sturmian parameter κ is a constant, and instead scale the Coulomb part of the potential by a new eigenvalue λ_n such that

$$\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{\lambda_n}{z} - \frac{\delta(1-\delta)}{2z^2}\right)\varphi_n = -\frac{\kappa^2}{2}\varphi_n, \tag{3.4.3}$$

where φ_n is the n'th eigenfunction of this modified Hamiltonian operator. The problem in equation (3.4.3) remains very reminiscent of a regular Schrödinger-type problem, save the eigenvalue-weighted potential. The purposes of this is to tune the potential in exactly such a way that keeps the energy constant for every eigenvalue. By choosing the energy negative, we ensure that every eigenfunction corresponds to a bound state, which in turn keeps the complete set entirely discrete. Eigenfunctions of this kind are called Sturmian basis functions and have been used in a variety of problems, see for instance [Susskind and Jensen, 1988]. The fact that they form a discrete and complete set makes them ideal for the proposed numerical approach. An additional benefit is that we can solve equation (3.4.3) relatively easily given that the unperturbed eigenstates are already known. To show this, we divide equation (3.4.3) through by λ_n^2 and make the change of variable $z' = \lambda_n z$ so that

$$\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z'^2} - \frac{1}{z'} - \frac{\delta(1-\delta)}{2z'^2}\right)\varphi_n = -\frac{\kappa^2}{2\lambda_n^2}\varphi_n. \tag{3.4.4}$$

Equation (3.4.4) exactly matches the eigenvalue problem of the quantum defect atom with energy eigenvalue $-\kappa^2/2\lambda_n^2$. We may now determine the eigenvalues λ_n by matching the energy eigenvalues with those of the quantum defect problem, $E_n = -1/2\varkappa^2$. This gives

$$-\frac{1}{2\varkappa^2} = -\frac{\kappa^2}{2\lambda_n^2} \quad \Rightarrow \quad \lambda_n = \kappa\varkappa. \tag{3.4.5}$$

It is then clear that the Sturmian basis functions may be expressed as $\varphi_n = \mathcal{N}'_n \psi_n(z')$, where ψ_n are the eigenstates of the quantum defect problem and $\mathcal{N}'_n$ is a new normalization factor. Using the eigenstates found in equation (2.2.13), we therefore conclude that

$$\begin{aligned}\varphi_n &= \frac{2^{1-\delta}(n)^{1/2}_{1-2\delta}}{n^{2-\delta}\Gamma(2-2\delta)}\mathcal{N}'_n z'^{1-\delta}\mathrm{e}^{-z'/n}{}_1F_1\left(1-n, 2-2\delta, \frac{2z'}{n}\right)\\ &= \frac{2^{1-\delta}(n)^{1/2}_{1-2\delta}}{n\Gamma(2-2\delta)}\mathcal{N}'_n(\kappa z)^{1-\delta}\mathrm{e}^{-\kappa z}{}_1F_1\left(1-n, 2-2\delta, 2\kappa z\right). \qquad (3.4.6)\end{aligned}$$

Before we determine $\mathcal{N}'_n$, we turn to deriving a particular weighted orthogonality relation, which it turns out that the set of Sturmian basis functions satisfies. To do so, we take two copies of the Sturmian eigenvalue problem, where the first is in state n and the second is in state m. Next, the first is multiplied through by $\varphi^\star_m$, whilst the second is conjugated and multiplied through by φ_n so that

$$\begin{aligned}-\frac{1}{2}\frac{\mathrm{d}^2\varphi_n}{\mathrm{d}z^2}\varphi^\star_m - \frac{\lambda_n}{z}\varphi_n\varphi^\star_m - \frac{\delta(1-\delta)}{2z^2}\varphi_n\varphi^\star_m &= -\frac{\kappa^2}{2}\varphi_n\varphi^\star_m,\\ -\frac{1}{2}\frac{\mathrm{d}^2\varphi^\star_m}{\mathrm{d}z^2}\varphi_n - \frac{\lambda_m}{z}\varphi^\star_m\varphi_n - \frac{\delta(1-\delta)}{2z^2}\varphi^\star_m\varphi_n &= -\frac{\kappa^2}{2}\varphi^\star_m\varphi_n. \qquad (3.4.7)\end{aligned}$$

Subtracting the latter from the former and integrating over $\mathbb{R}_{\geq 0}$ then yields

$$(\lambda_m - \lambda_n)\int_0^\infty \varphi_n\frac{1}{z}\varphi^\star_m\mathrm{d}z = \frac{1}{2}\int_0^\infty \frac{\mathrm{d}^2\varphi_n}{\mathrm{d}z^2}\varphi^\star_m\mathrm{d}z - \frac{1}{2}\int_0^\infty \frac{\mathrm{d}^2\varphi^\star_m}{\mathrm{d}z^2}\varphi_n\mathrm{d}z. \qquad (3.4.8)$$

We now apply integration by parts to the two integrals on the right-hand side as a means of lowering the order of differentiation by one. This gives

$$\begin{aligned}(\lambda_m - \lambda_n)\int_0^\infty \varphi_n\frac{1}{z}\varphi^\star_m\mathrm{d}z &= \frac{1}{2}\left(\left[\frac{\mathrm{d}\varphi_n}{\mathrm{d}z}\varphi^\star_m\right]_0^\infty - \int_0^\infty \frac{\mathrm{d}\varphi_n}{\mathrm{d}z}\frac{\mathrm{d}\varphi^\star_m}{\mathrm{d}z}\mathrm{d}z\right) - \frac{1}{2}\left(\left[\frac{\mathrm{d}\varphi^\star_m}{\mathrm{d}z}\varphi_n\right]_0^\infty - \int_0^\infty \frac{\mathrm{d}\varphi^\star_m}{\mathrm{d}z}\frac{\mathrm{d}\varphi_n}{\mathrm{d}z}\mathrm{d}z\right)\\ &= \left[\frac{\mathrm{d}\varphi_n}{\mathrm{d}z}\varphi^\star_m - \frac{\mathrm{d}\varphi^\star_m}{\mathrm{d}z}\varphi_n\right]_0^\infty - \int_0^\infty\left(\frac{\mathrm{d}\varphi^\star_m}{\mathrm{d}z}\frac{\mathrm{d}\varphi_n}{\mathrm{d}z} - \frac{\mathrm{d}\varphi_n}{\mathrm{d}z}\frac{\mathrm{d}\varphi^\star_m}{\mathrm{d}z}\right)\mathrm{d}z\\ &= \left[\frac{\mathrm{d}\varphi_n}{\mathrm{d}z}\varphi^\star_m - \frac{\mathrm{d}\varphi^\star_m}{\mathrm{d}z}\varphi_n\right]_0^\infty. \qquad (3.4.9)\end{aligned}$$

Like the quantum defect atom eigenstates, the Sturmian basis satisfies Dirichlet boundary conditions and is square-integrable on $\mathbb{R}_{\geq 0}$, which in turn ensures that both the lower and upper bound of the term on the right-hand side of equation (3.4.9) vanishes. This leaves the orthogonality relation

$$(\lambda_m - \lambda_n)\int_0^\infty \varphi^\star_m\frac{1}{z}\varphi_n\mathrm{d}z = 0, \qquad (3.4.10)$$

which in turn means that the Sturmian basis functions are orthogonal with weight $1/z$. It proves convenient to choose the normalization constant $\mathcal{N}'_n$ so that equation (3.4.10) becomes an orthonormality condition, meaning $\langle\varphi_m|z^{-1}|\varphi_n\rangle = \delta_{mn}$. The $m = n$ case of this condition becomes

$$\frac{(2\kappa)^{2-2\delta}(n)_{1-2\delta}}{n^2\Gamma^2(2-2\delta)}\mathcal{N}'^2_n\int_0^\infty z^{1-2\delta}\mathrm{e}^{-2\kappa z}\left[{}_1F_1\left(1-n, 2-2\delta, 2\kappa z\right)\mathrm{d}z\right]^2 = 1. \qquad (3.4.11)$$

This integral is another example of the W. Gordon integral given in equation (A.0.15). Substituting the resulting expression and solving for the normalization constant then yields

$$\begin{aligned}\mathcal{N}'_n &= \left\{\frac{(2\kappa)^{2-2\delta}(n)_{1-2\delta}}{n^2\Gamma^2(2-2\delta)}\frac{\Gamma(2-2\delta)(n-1)!}{(2\kappa)^{2-2\delta}(2-2\delta)_{n-1}}{}_3F_2(1-n, 0, 1, 2-2\delta, 1, 1)\right\}^{-1/2}\\ &= \left\{\frac{(2\kappa)^{2-2\delta}(n)_{1-2\delta}}{n^2\Gamma^2(2-2\delta)}\frac{\Gamma^2(2-2\delta)}{(2\kappa)^{2-2\delta}(n)_{1-2\delta}}\right\}^{-1/2}\\ &= n. \qquad (3.4.12)\end{aligned}$$

Finally, using the normalization constant given in equation (3.4.12), the full Sturmian basis functions may be written as

$$\begin{aligned}\varphi_n &= \frac{2^{1-\delta}(n)_{1-2\delta}^{1/2}}{\mathscr{n}\Gamma(2-2\delta)}\mathscr{n}(\kappa z)^{1-\delta}\mathrm{e}^{-\kappa z}{}_1F_1\left(1-n, 2-2\delta, 2\kappa z\right) \\ &= \frac{2^{1-\delta}(n)_{1-2\delta}^{1/2}}{\Gamma(2-2\delta)}(\kappa z)^{1-\delta}\mathrm{e}^{-\kappa z}\times\frac{\Gamma(n)}{(2-2\delta)_{n-1}}L_{n-1}^{(1-2\delta)}(2\kappa z) \\ &= (n)_{1-2\delta}^{-1/2}(2\kappa z)^{1-\delta}\mathrm{e}^{-\kappa z}L_{n-1}^{(1-2\delta)}(2\kappa z),\end{aligned} \tag{3.4.13}$$

where we for later convenience have used equation (A.0.6) to convert the hypergeometric function into an associated Laguerre polynomial. Now that we have determined the Sturmian basis functions, we turn to developing the chosen numerical method further by returning to equation (3.4.2). We proceed by writing the perturbed eigenstate in terms of an expansion of the Sturmian basis, meaning $\tilde{\psi}_n = \sum_{k=1}^{\infty} c_{n,k}\varphi_k$, where $c_{n,k}$ are some expansion coefficients. Substituting in the expansion then yields

$$\left(-\frac{1}{2}\mathrm{e}^{-2i\theta}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \mathrm{e}^{-i\theta}\frac{1}{z} - \mathrm{e}^{-2i\theta}\frac{\delta(1-\delta)}{2z^2} + \mathrm{e}^{i\theta}\mathcal{E}_0 z\right)\sum_{k=1}^{\infty} c_{n,k}\varphi_k = \tilde{E}_n\sum_{k=1}^{\infty} c_{n,k}\varphi_k. \tag{3.4.14}$$

Before proceeding, we return to the Sturmian eigenvalue problem given in equation (3.4.3), which can be rearranged as

$$-\frac{1}{2}\frac{\mathrm{d}^2\varphi_n}{\mathrm{d}z^2} = \frac{\lambda_n}{z}\varphi_n + \frac{\delta(1-\delta)}{2z^2}\varphi_n - \frac{\kappa^2}{2}\varphi_n. \tag{3.4.15}$$

We can now use equation (3.4.15) to eliminate the kinetic energy operator from the rotated problem in equation (3.4.14), which simply leaves potential and energy terms. This property of the Sturmian basis functions is crucial because it often allows the perturbed energy to be determined accurately with very small basis sets [Antonsen, 1999]. Substituting into the above expression with $\lambda_n = \kappa\mathscr{n}$ and reducing the resulting expression then leaves

$$\sum_{k=1}^{\infty}\left\{\kappa(k-\delta)\mathrm{e}^{-2i\theta}\frac{1}{z} + \mathrm{e}^{i\theta}\mathcal{E}_0 z - \frac{1}{2}\kappa^2\mathrm{e}^{-2i\theta} - \mathrm{e}^{-i\theta}\frac{1}{z}\right\}c_{n,k}\varphi_k = \tilde{E}_n\sum_{k=1}^{\infty} c_{n,k}\varphi_k. \tag{3.4.16}$$

We ultimately want to use equation (3.4.16) to construct a matrix equation. We accomplish this by first multiplying the equation through by $\varphi_m^{\star}$, integrating over $\mathbb{R}_{\geq 0}$ and truncating the summation to some finite N. Using the orthonormality condition given in equation (3.4.10), we find

$$\sum_{k=1}^{N} H_{k,m}c_{n,k} \approx \tilde{E}_n\sum_{k=1}^{N} K_{k,m}c_{n,k}, \tag{3.4.17}$$

where we have defined $K_{k,m} = \langle\varphi_m|\varphi_k\rangle$ and

$$H_{k,m} = \kappa(k-\delta)\mathrm{e}^{-2i\theta}\delta_{km} + \mathrm{e}^{i\theta}\mathcal{E}_0\left\langle\varphi_m\left|z\right|\varphi_k\right\rangle - \frac{1}{2}\kappa^2\mathrm{e}^{-2i\theta}\langle\varphi_m|\varphi_k\rangle - \mathrm{e}^{-i\theta}\delta_{km}. \tag{3.4.18}$$

Before finalizing, we need only determine the two integrals $\langle\varphi_m|\varphi_k\rangle$ and $\left\langle\varphi_m\left|z\right|\varphi_k\right\rangle$. Starting with $\langle\varphi_m|\varphi_k\rangle$, the initial calculation gives

$$\begin{aligned}\langle\varphi_m|\varphi_k\rangle &= \frac{1}{(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\int_0^{\infty}(2\kappa z)^{2-2\delta}\mathrm{e}^{-2\kappa z}L_{m-1}^{(1-2\delta)}(2\kappa z)L_{k-1}^{(1-2\delta)}(2\kappa z)\mathrm{d}z \\ &= \frac{1}{2\kappa(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\int_0^{\infty} z'^{2-2\delta}\mathrm{e}^{-z'}L_{m-1}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z',\end{aligned} \tag{3.4.19}$$

where we have applied the change of variable $z' = 2\kappa z$. We can evaluate the integral in equation (3.4.19) by first applying the functional identity given in equation (A.0.5), which for the $L_{m-1}^{(1-2\delta)}$ polynomial becomes

$$z' L_{m-1}^{(1-2\delta)}(z') = 2\mathscr{m} L_{m-1}^{(1-2\delta)}(z') - m L_{m}^{(1-2\delta)}(z') - (m-2\delta) L_{m-2}^{(1-2\delta)}(z'). \tag{3.4.20}$$

Substituting this into the above integral then yields

$$\begin{aligned}\langle\varphi_m|\varphi_k\rangle = \frac{1}{2\kappa(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\Bigg\{ & 2\mathscr{m}\int_0^\infty z'^{1-2\delta}\mathrm{e}^{-z'}L_{m-1}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z' \qquad (3.4.21)\\ & - m\int_0^\infty z'^{1-2\delta}\mathrm{e}^{-z'}L_{m}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z' - (m-2\delta)\int_0^\infty z'^{1-2\delta}\mathrm{e}^{-z'}L_{m-2}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z'.\end{aligned}$$

One may now apply the orthogonality relation of the Laguerre polynomials given in equation (A.0.4). This reduces equation (3.4.21) to

$$\begin{aligned}\langle\varphi_m|\varphi_k\rangle &= \frac{1}{2\kappa(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\left\{2\mathscr{m}(k)_{1-2\delta}\delta_{k,m} - m(k)_{1-2\delta}\delta_{k-1,m} - (m-2\delta)(k)_{1-2\delta}\delta_{k,m-1}\right\}\\ &= \frac{(k)_{1-2\delta}^{1/2}}{2\kappa(m)_{1-2\delta}^{1/2}}\left\{2\mathscr{m}\delta_{k,m} - m\delta_{k-1,m} - (m-2\delta)\delta_{k,m-1}\right\}, \qquad (3.4.22)\end{aligned}$$

which gives the first integral. A similar approach can be used to compute the second integral $\langle\varphi_m\,|z|\,\varphi_k\rangle$, which initially becomes

$$\begin{aligned}\langle\varphi_m\,|z|\,\varphi_k\rangle &= \frac{(2\kappa)^{2-2\delta}}{(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\int_0^\infty z^{3-2\delta}\mathrm{e}^{-2\kappa z}L_{m-1}^{(1-2\delta)}(2\kappa z)L_{k-1}^{(1-2\delta)}(2\kappa z)\mathrm{d}z\\ &= \frac{1}{4\kappa^2(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\int_0^\infty z'^{3-2\delta}\mathrm{e}^{-z'}L_{m-1}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z', \qquad (3.4.23)\end{aligned}$$

where we again have applied the change of coordinates $z' = 2\kappa z$. Substituting the functional identity in equation (3.4.20) into the above then yields

$$\begin{aligned}\langle\varphi_m\,|z|\,\varphi_k\rangle = \frac{1}{4\kappa^2(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\Bigg\{ & 2\mathscr{m}\int_0^\infty z'^{2-2\delta}\mathrm{e}^{-z'}L_{m-1}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z' \qquad (3.4.24)\\ & - m\int_0^\infty z'^{2-2\delta}\mathrm{e}^{-z'}L_{m}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z' - (m-2\delta)\int_0^\infty z'^{2-2\delta}\mathrm{e}^{-z'}L_{m-2}^{(1-2\delta)}(z')L_{k-1}^{(1-2\delta)}(z')\mathrm{d}z'.\end{aligned}$$

The three integrals in equation (3.4.24) are special cases of the first integral $\langle\varphi_m|\varphi_k\rangle$. Substituting in equation (3.4.19) then allows us to reduce equation (3.4.24) to

$$\begin{aligned}\langle\varphi_m\,|z|\,\varphi_k\rangle &= \frac{1}{4\kappa^2(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}}\Bigg\{4\kappa\mathscr{m}(m)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}\langle\varphi_m|\varphi_k\rangle\\ &\qquad - 2\kappa m(m+1)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}\langle\varphi_{m+1}|\varphi_k\rangle - 2\kappa(m-2\delta)(m-1)_{1-2\delta}^{1/2}(k)_{1-2\delta}^{1/2}\langle\varphi_{m-1}|\varphi_k\rangle\Bigg\}\\ &= \frac{1}{2\kappa(m)_{1-2\delta}^{1/2}}\Bigg\{2\mathscr{m}(m)_{1-2\delta}^{1/2}\langle\varphi_m|\varphi_k\rangle - m(m+1)_{1-2\delta}^{1/2}\langle\varphi_{m+1}|\varphi_k\rangle - (m-2\delta)(m-1)_{1-2\delta}^{1/2}\langle\varphi_{m-1}|\varphi_k\rangle\Bigg\}\\ &= \frac{2\mathscr{m}\langle\varphi_m|\varphi_k\rangle - \sqrt{m(m+1-2\delta)}\langle\varphi_{m+1}|\varphi_k\rangle - \sqrt{(m-2\delta)(m-1)}\langle\varphi_{m-1}|\varphi_k\rangle}{2\kappa}, \qquad (3.4.25)\end{aligned}$$

which provides the second integral in terms of the first integral $\langle\varphi_m|\varphi_k\rangle$ given in equation (3.4.22). This allows us to evaluate $H_{k,m}$ given in equation (3.4.18) for each k, m. We note that although φ_0 is not defined, equation (3.4.25) is still valid in the $m = 1$ case because the third term vanishes. We may now finalize the numerical approach by taking N copies of equation (3.4.17), where we let m run through every integer between one and N. The resulting N equations can then be expressed as the matrix equation

$$\mathbf{H}\mathbf{c}_n \approx \tilde{E}_n\mathbf{K}\mathbf{c}_n, \tag{3.4.26}$$

where $\mathbf{c}_n$ is a $N \times 1$ column vector containing the basis expansion coefficients and H, K are $N \times N$ matrices whose entries are given by the previously defined quantities $H_{k,m}$ and $K_{k,m}$. Equation (3.4.26) represents a generalized eigenvalue problem, which we can solve in for instance Python or Mathmatica. An important point to address is the choice of Sturmian parameter κ and complex scaling angle θ. Although the energy eigenvalues should, in principle, be independent of both, they are only approximately so because the basis expansion is truncated. The sign of the Sturmian parameter κ has no effect on the basis functions, so we can restrict $\kappa \in \mathbb{R}_{\geq 0}$. Testing in Python shows that the solutions of equation (3.4.26) are effectively independent of κ as long as it is not too close to zero, where the eigenvalue solver becomes unstable. A natural choice is $\kappa = 1$, for which the energy associated with Sturmian basis functions is the ground state energy of the Coulomb potential. The solutions of equation (3.4.26) show a stronger dependence on the complex scaling angle. To showcase this, we have plotted the energy eigenvalues provided by the outlined numerical approach for various complex scaling angles. This is shown in figure Figure 3.2 below, where we consider the ground state and first excited state for both the Coulomb potential and Kratzer-like potential.

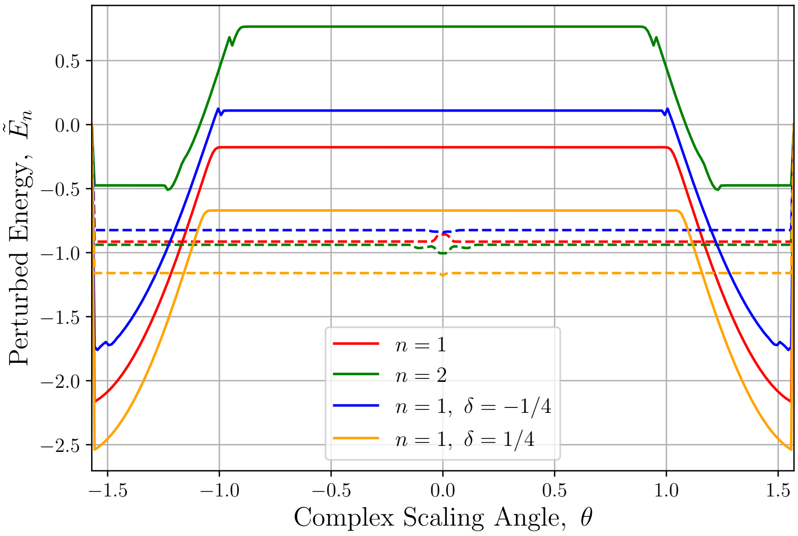

Figure 3.2: The perturbed energy $\tilde{E}_n$ is plotted as a function of the complex scaling angle θ in the interval $[-\pi/2, \pi/2]$ radians. The Sturmian basis expansion is truncated at $N = 150$ and the parameter $\kappa = 1$ is chosen. The ground state and first excited state of the Coulomb potential and the $\delta = \pm 1/4$ cases of the Kratzer potential ground state are shown. The solid line shows the field strength $\mathcal{E}_0 = 1/4$, whereas the dashed line shows the corresponding state with field strength $\mathcal{E}_0 = -1/4$. For positive field strength, the energy eigenvalues become unstable for too large θ. For negative field strength, the energy eigenvalues become unstable for θ near zero.

Figure Figure 3.2 shows the dependence of the perturbed energy $\tilde{E}_n$ on the complex scaling angle θ for the ground state and first excited state of both the Coulomb potential and the Kratzer-like potential. For applied fields pointing away from the substrate, we find that the energy eigenvalues remain stable within an interval centred around $\theta = 0$. Outside this window, the eigenvalues begin to deviate significantly. The width of this interval generally decreases with quantum number n and increases with basis length N. For negative field strengths, that induce tunnelling, we instead see that the energy eigenvalues become unstable for θ too close to zero, but otherwise remain stable up until $\theta = \pi/2$, where the model breaks down completely for all applied fields. Similarly to the positive field case, we also find that a smaller selection of complex scaling angles lead to convergence if the basis length is not chosen large enough. With the numerical approach now presented, we can move forward and discuss the results of the two provided methods.

3.5 Stark Shift & Ionization Rate

We have now presented both the analytical hypergeometric approximant approach and the numerical scheme provided by the complex-scaled Sturmian basis expansion method. We may then finally begin to apply these approaches and model the full complex energy eigenvalues $\tilde{E}_n - i\Gamma_n/2$. We start by making a very direct comparison between the two approaches as a means testing the validity of the hypergeometric approximant. This is shown in Figure 3.3 below, in which the perturbed energy $\tilde{E}_n$ is plotted as a function of the applied field strength for the first five states of the Coulomb potential, as well as two examples of the ground state for the Kratzer-like potential.

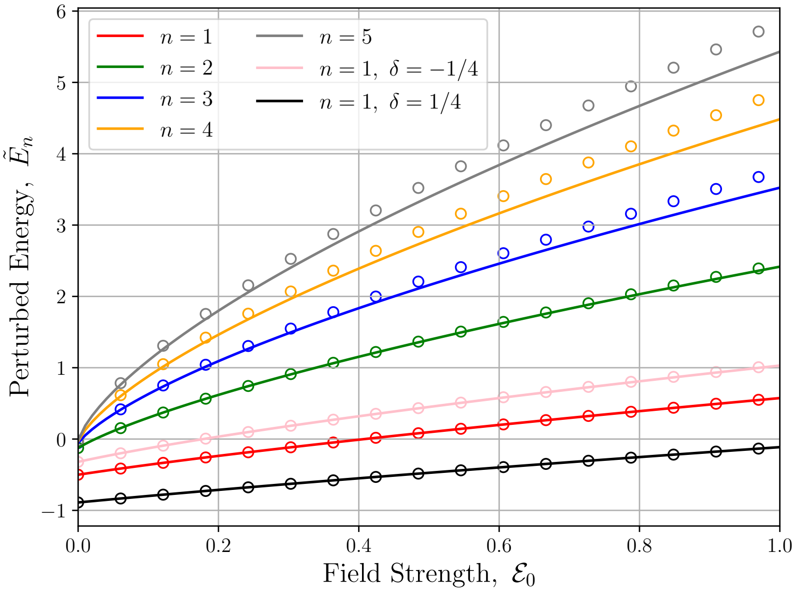

Figure 3.3: The perturbed energy is plotted as a function of the strength of the applied field $\mathcal{E}_0$ in the unit interval. The ground state and first four excited states of the Coulomb potential and the $\delta = \pm 1/4$ cases of the Kratzer potential ground state are shown with a complex scaling angle of $\theta = 0.2$. The curves showcase the corresponding hypergeometric approximant whilst the scatter points showcase the numerical results provided by the Sturmian basis expansion, which is truncated at $N = 200$. The parameter $\kappa = 1$ is chosen. The accuracy of the approximant generally decreases with larger perturbations and principle quantum number. The ground state approximants are highly accurate even at unit field strength, whereas the $n = 5$ state loses accuracy already at $\mathcal{E}_0 = 0.2$.

Figure 3.3 shows the dependence of the perturbed energy $\tilde{E}_n$ on the strength of the applied field $\mathcal{E}_0$. The solid lines and scatter points showcase the hypergeometric approximants and the numerical approach, respectively. The hypergeometric functions provide a solid approximation of the perturbed energy, the ground state and first excited state in particular. In general, the approximants lose accuracy with increasing field strength $\mathcal{E}_0$ and principle quantum number n. The approximant for the fourth excited state begins deviating from the numerical solution already at $\mathcal{E}_0 \approx 0.2$, whereas the ground state remains accurate even at unit field strength. This is a consequence of the approximants only containing fourth-order information. We can validate the hypergeometric approximant approach further by comparison with experimental observations, such as those provided in [Grimes et al., 1976] or [Lambert and Richards, 1981]. Here, a microwave absorption cell is used to measure the transition frequencies $f_{n \to m}$ of electrons lying on a liquid Helium substrate as a function of the potential difference ΔV across the cell. One of the cells used is a right-circular cylinder of radius $r_c = 1.1$ cm and height $h_c = 0.32$ cm. The electric field runs parallel with the cell and hence we may relate the potential difference across the cell with the electric field using $\Delta V = \mathcal{E}_0 h_c$. The transition frequency $f_{n\to m}$ refers to the frequency of the radiation required to excite an electron from state n to state m, meaning $hf_{n\to m} = \tilde{E}_m - \tilde{E}_n$, where h is Plancks constant. By calculating the transition frequencies using the hypergeometric approximants and plotting the results in terms of the potential difference, we may compare directly with the findings of [Grimes et al., 1976]. This is shown in Figure 3.4 below, where the 1 → 2 and 1 → 3 transitions are shown.

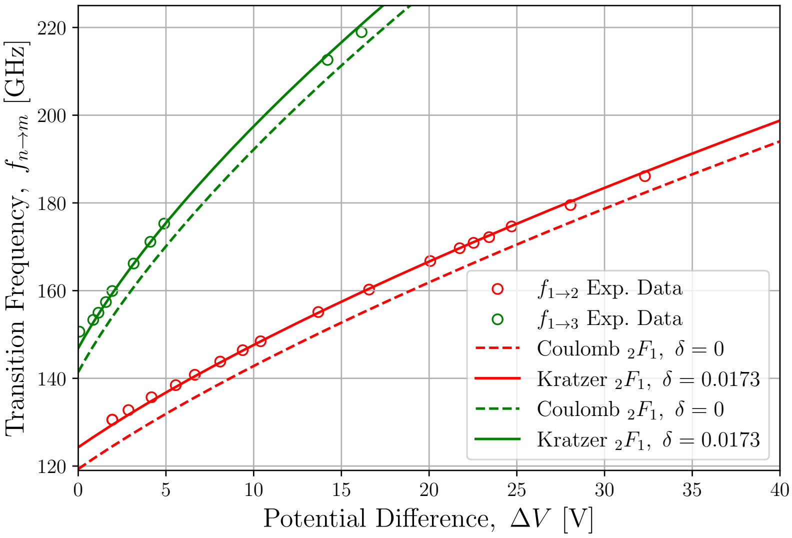

Figure 3.4: The transition frequencies $f_{n\to m}$ are plotted as a function of the potential difference across the right-circular cell. The scatter points show the measured transition frequencies of the 1 → 2 and 1 → 3 transitions provided in [Grimes et al., 1976]. The experimental data is compared with the corresponding hypergeometric approximants for both the Coulomb potential and the Kratzer-like potential. The quantum defect parameter is re-estimated using a simple residual sum-of-squares technique. Two data points pertaining to the 1 → 2 transition lying at $\Delta V \approx 60$ V are not included in the plot, but are considered in the re-estimation of the quantum defect parameter.

Figure 3.4 shows the transition frequency $f_{n\to m}$ as a function of the potential difference applied across the right-circular cell. The solid and dashed lines show the transition frequencies calculated from the hypergeometric approximants of the Kratzer-like potential and Coulomb potential, respectively, and the scatter points show the experimental observations provided in [Grimes et al., 1976]. The quantum defect parameter is re-estimated with a simple residual sum of squares technique, which yields $\delta = 0.0173$. This differs by approximately 36% from the initial $\delta = 0.0237$ estimate stemming from fitting the unperturbed 1 → 2 and 1 → 3 transitions. It clear that the bare Coulomb potential is an inadequate model in comparison with the Kratzer-like potential, which is required to reproduce the observations accurately. The Coulomb approximants are consistently off set by roughly 25-26 GHz and 28-30 GHz for the 1 → 2 transitions and the 1 → 3 transitions, respectively. With the re-estimated quantum defect parameter, we see that the hypergeometric approximant captures the experimental data well. A search for similar experimental data for the equivalent Neon substrate was made, but no such data could be found. The dependence of the perturbed energy on the quantum defect parameter can also be examined more closely using the hypergeometric approximants. This is shown in Figure 3.5 below, in which the perturbed energy is plotted as a function of quantum defect parameters for a variety of applied field strengths.

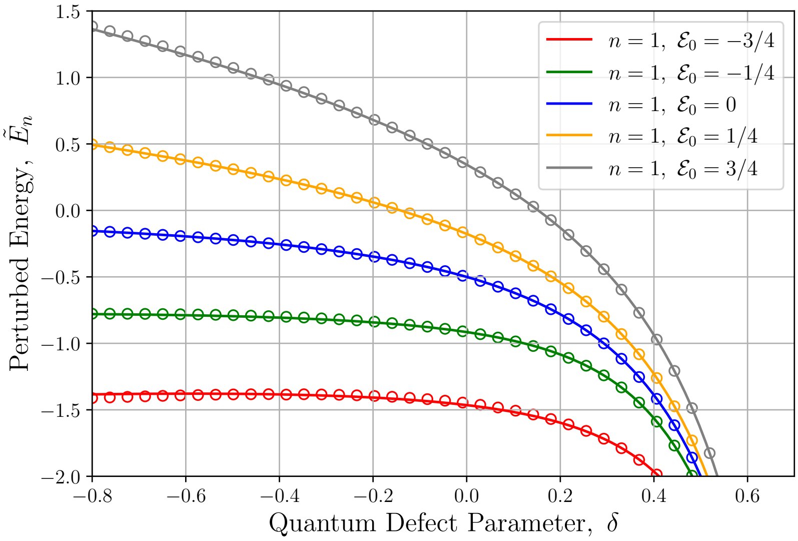

Figure 3.5: The perturbed energy is plotted as a function of the quantum defect parameter δ in the interval $[-0.8, 0.7]$. The ground state of the Kratzer-like potential is shown for various applied field strengths with a complex scaling angle of $\theta = 0.2$. The solid lines showcase the hypergeometric approximant whilst the scatter points showcase the numerical results provided by the Sturmian basis expansion, which is truncated at $N = 200$. The parameter $\kappa = 1$ is chosen.

Figure 3.5 shows the dependence of the perturbed energy $\tilde{E}_n$ on the quantum defect parameter δ. The solid lines show the hypergeometric approximants and the scatter points show the numerical results provided by the Sturmian basis expansion. As the quantum defect parameter increases, the electron is more strongly bound to the surface of the substrate and the energy decreases. Likewise, the rate at which the energy changes also increases with δ, which leads to more dramatic energy shifts for more positive quantum defect parameters. For sufficiently negative δ, we instead find that the perturbed energy eventually settles unless a sufficiently strong outward-facing field is present. Finally, we now, with particular focus on the ground state and first excited state, move forward and extend our analysis to include ionizing fields. This corresponds to $\mathcal{E}_0 < 0$ in this case because the electron is negatively charged and therefore moves in the direction opposite to the applied field. The inclusion of negative field strengths is showcased in Figure 3.6, where we consider the ground state and first excited of the Coulomb potential, as well as two instances of the ground state of the Kratzer-like potential.

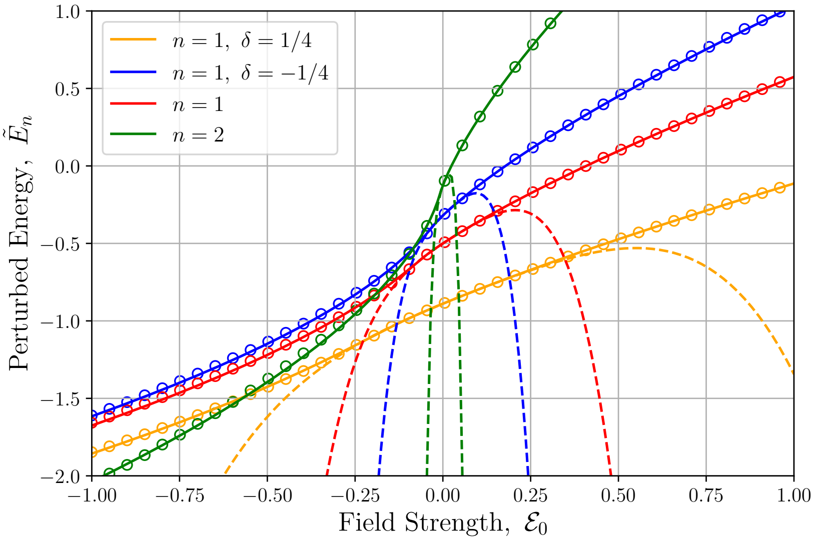

Figure 3.6: The perturbed energy is plotted as a function of the strength of the applied field $\mathcal{E}_0$ in the interval $[-1, 1]$. The ground state and first excited state of the Coulomb potential and the $\delta = \pm 1/4$ cases of the Kratzer potential ground state are shown with a complex scaling angle of $\theta = 0.2$. The solid lines showcase the corresponding hypergeometric approximant whilst the scatter points showcase the numerical results provided by the Sturmian basis expansion, which is truncated at $N = 200$. The parameter $\kappa = 1$ is chosen. The dashed lines show the original perturbative expansions derived from the Dalgarno-Lewis approach, which have also been included but are truncated at fourth-order similarly to the hypergeometric approximants.

Figure 3.6 shows the dependence of the perturbed energy $\tilde{E}_n$ on the strength of the applied field $\mathcal{E}_0$ in both directions. The solid lines show the hypergeometric approximants, the dashed lines show the truncated perturbative expansion and the scatter points showcase the numerical approach provided by the Sturmian basis expansion. The approximants retain good accuracy across the entire field strength spectrum. This is in stark contrast to the perturbative expansions, which diverge from the numerical solutions outside a relatively small interval centred at $\mathcal{E}_0 = 0$. The first excited state is a clear example in that its perturbative expansion diverges almost immediately. The tunnelling induced by negative field strengths is reflected in the energy eigenvalues acquiring a non-zero imaginary part. This is shown in Figure 3.7 below, where we repeat the previous example.

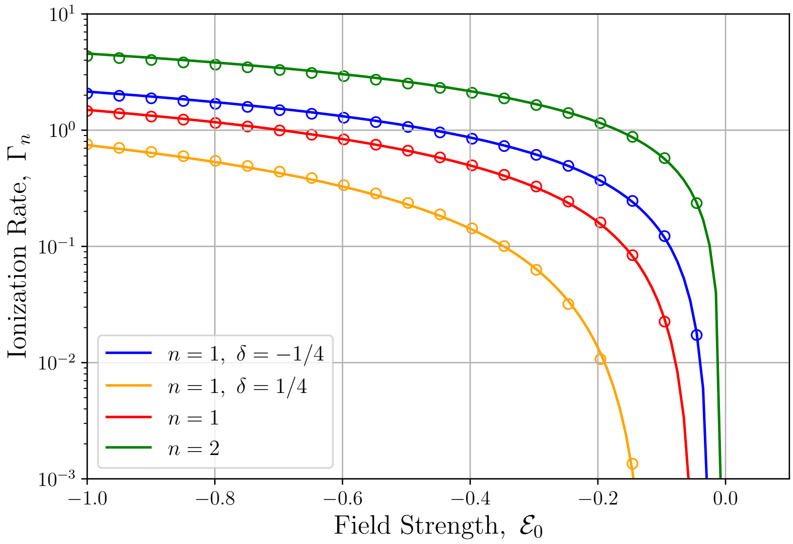

Figure 3.7: The ionization rate is plotted as a function of the strength of the applied field $\mathcal{E}_0$ in the interval $[-1, 0]$. The ground state and first excited state of the Coulomb potential and the $\delta = \pm 1/4$ cases of the Kratzer potential ground state are shown with a complex scaling angle of $\theta = 0.2$. The curves showcase the corresponding hypergeometric approximant whilst the scatter points showcase the numerical results provided by the Sturmian basis expansion, which is truncated at $N = 200$. The parameter $\kappa = 1$ is chosen.

Figure 3.7 shows the dependence of the ionization rate Γ_n on the strength of the applied field $\mathcal{E}_0$. The solid lines show the hypergeometric approximants and the scatter points showcase the numerical approach provided by the Sturmian basis expansion. Again, we find that the hypergeometric approximants accurately replicate the ionization rates provided by the numerical approach, which as expected goes to zero as the sign of the field begins to flip. Despite the fact that the critical field, at which the approximants develop an imaginary part, of the $\delta = 1/4$ case is an order of magnitude larger than the first excited Coulomb state, one finds that the approximants of the more extreme cases of the Kratzer-like potential still agree with the numerical approach. Although the critical field of the $\delta = 1/4$ case is an order of magnitude larger than that of the first excited Coulomb state, the approximants for the more extreme cases of the Kratzer-like potential still show good agreement with the numerical results at smaller field strengths. It is also important to note that the ionization rate of the excited state is significantly larger than the ground state, in particular at lower field strengths where they differ by almost one order of magnitude. This is a testament to the weak binding induced by the image potential effect. This is also made clear by the fact that Γ_2 is almost immediately comparable in magnitude with the corresponding unperturbed energy $E_2 = -1/8$. All in all, the hypergeometric approximants stands as a very solid approach to modeling the energy once the electrostatic perturbation is applied. This concludes the present chapter and we now move onto the next, in which we describe the electrodynamic perturbation.

Chapter 4

Electrodynamic Perturbation of Qubit

In this chapter, we further develop the results of chapters one and two by perturbing the electron system with a full harmonic field. By expanding the eigenstates in a perturbative series, we determine the eigenstate corrections using two methods. The eigenstate corrections are subsequently used to determine the dynamic response of the system due to the external field. Using these corrections, we determine the dynamic linear polarizability. The electro-optic response induced by simultaneous static and dynamic perturbation is then found as well.

4.1 Electrodynamic Perturbation of Qubit

We now aim to further develop the results of chapter one by introducing a time-dependent perturbation to the electron qubit system. In particular, we perturb the system with a harmonic electrodynamic field of the form $\boldsymbol{\mathcal{E}} = \mathcal{E}_0 \cos \omega t \hat{\mathbf{z}}$, where ω is the frequency of the external field pointing in the positive $\hat{\mathbf{z}}$ direction, as shown in Figure 4.1 below. The introduction of a dynamic electric field within the electron-qubit system provides a different but equally important kind of electric qubit control in comparison with the electrostatic perturbation.

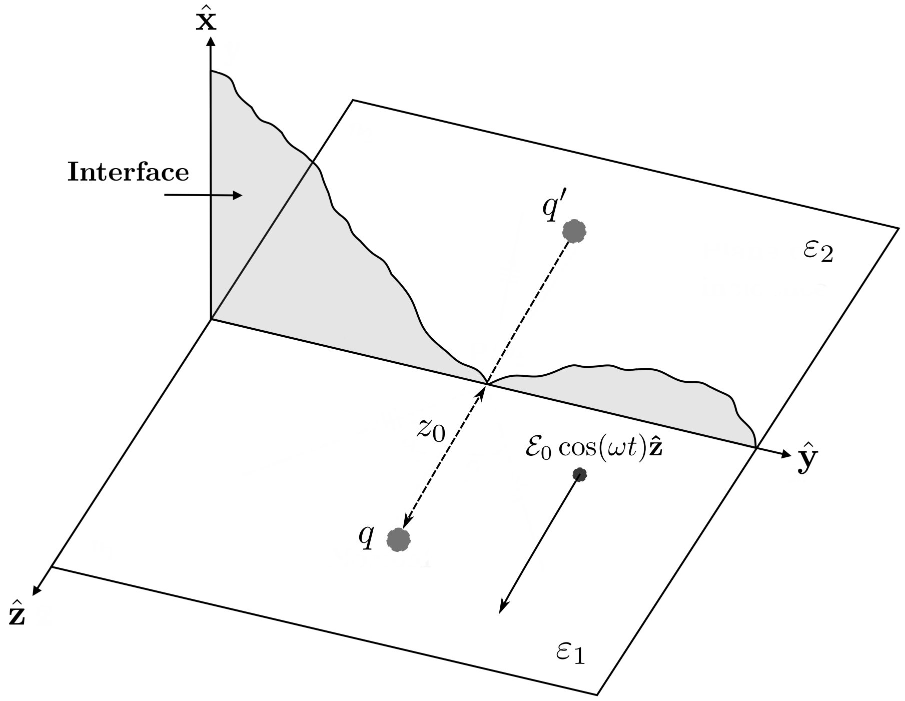

Figure 4.1: Electron-on-Helium qubit system perturbed by the electrodynamic field $\boldsymbol{\mathcal{E}} = \mathcal{E}_0 \cos \omega t \hat{\mathbf{z}}$, which is chosen so that it points away from the substrate. The figure is based on [Pedrotti et al., 2018].

The procedure for dealing with electrodynamic perturbations follows a method similar to the one presented in the previous chapter. As opposed to the electrostatic perturbation however, we are now dealing with a time-dependent perturbation. Our starting point for the dynamic perturbation is therefore the time-dependent Schrödinger equation. The contribution to the Hamiltonian due to the electric field is once again given by the induced dipole moment **p** and the energy associated with it U. Hence, the perturbation of the Hamiltonian in modified atomic units is $\mathcal{E}_0 \cos(\omega t) z$ and the full perturbed problem therefore reads

$$\left\{ \hat{H}_0 + \frac{1}{2}\hat{H}_1 \mathrm{e}^{i\omega t} + \frac{1}{2}\hat{H}_1 \mathrm{e}^{-i\omega t} \right\} \tilde{\Psi} = i\partial_t \tilde{\Psi}, \tag{4.1.1}$$

where we for later convenience have expressed the cosine function in terms of its complex frequency modes. We now proceed in a similar manner to the electrostatic perturbation and expand the eigenstate in a perturbative expansion of the form

$$\tilde{\Psi}_n = \sum_{i=0}^{\infty} \tilde{\Psi}_{n,i}, \tag{4.1.2}$$

where $\tilde{\Psi}_{n,i} = \tilde{\Psi}_{n,i}(z,t)$ is the i'th correction of the eigenstate that solves the time-dependent Schrödinger equation. The eigenstates $\tilde{\Psi}_{n,0} = \Psi_n$ are the solutions of the corresponding unperturbed problem $\hat{H}_0 \Psi_n = i\partial_t \Psi_n$, which is solved by assuming that $\Psi_n(z,t) = f(t)\psi_n(z)$ for some unknown $f(t)$. Using the corresponding time-independent Schrödinger equation $\hat{H}_0 \psi_n = E_n \psi_n$ to eliminate the Hamiltonian under this assumption, we then find that $\Psi_n = \psi_n \mathrm{e}^{-iE_n t}$. We now move forward by substituting the perturbative expansion above into the dynamic problem given in equation (4.1.2), which after rearranging terms yields

$$\left(\hat{H}_0 + \frac{1}{2}\hat{H}_1 \mathrm{e}^{i\omega t} + \frac{1}{2}\hat{H}_1 \mathrm{e}^{-i\omega t} - i\partial_t \right) \left(\tilde{\Psi}_{n,0} + \tilde{\Psi}_{n,1} + \tilde{\Psi}_{n,2} + \tilde{\Psi}_{n,3} + \dots \right) = 0. \tag{4.1.3}$$

Terms of equal perturbative order are now grouped individually as in the electrostatic problem. The zeroth-order grouping returns the unperturbed problem, which is solved just above. For every other perturbative order, we instead find groupings of the form

$$\left(i\partial_t - \hat{H}_0 \right) \tilde{\Psi}_{n,p} = \frac{1}{2}\hat{H}_1 \tilde{\Psi}_{n,p-1} \left(\mathrm{e}^{i\omega t} + \mathrm{e}^{-i\omega t} \right), \quad p > 0. \tag{4.1.4}$$

Equation (4.1.4) is a considerably more complicated problem to solve than the electrostatic problem given in equation (3.1.6). The main reason for this is of course the added time dependence, which consequently invalidates certain methods that do apply in the static case, such as the Dalgarno-Lewis approach. An approach to solving equation (4.1.4) above is to decompose it into each of its frequency modes. This reduces the dynamic problem at hand to a number of static problems. Specifically, in a manner very similar to grouping terms in powers of the electric field strength, we group terms with equal frequency modes, i.e. $\mathrm{e}^{-i\omega t}$, and require them to vanish. To accomplish this, we choose $\tilde{\Psi}_{n,p}$ so that its frequency modes match those given on the right-hand side of equation (4.1.4). We illustrate this with the $p = 1$ case, for which substituting $\tilde{\Psi}_{n,0} = \psi_n \mathrm{e}^{-iE_n t}$ and $\hat{H}_1 = \mathcal{E}_0 z$ yields

$$\left(i\partial_t - \hat{H}_0 \right) \tilde{\Psi}_{n,1} = \frac{1}{2}\mathcal{E}_0 z \psi_n \left(\mathrm{e}^{-i(E_n - \omega)t} + \mathrm{e}^{-i(E_n + \omega)t} \right). \tag{4.1.5}$$

Now that the two frequency modes $\mathrm{e}^{-i(E_n \pm \omega)t}$ are clearly identified, we can solve equation (4.1.5) by writing

$$\tilde{\Psi}_{n,1} = \frac{1}{2}\mathcal{E}_0 \psi_{n,1}^{(1)} \mathrm{e}^{-i(E_n + \omega)t} + \frac{1}{2}\mathcal{E}_0 \psi_{n,1}^{(-1)} \mathrm{e}^{-i(E_n - \omega)t}, \tag{4.1.6}$$

where we to each frequency mode have attached a corresponding eigenstate component $\psi_{n,1}^{(\pm 1)}$ whose superscript denotes the given ω-dependence. Substituting $\tilde{\Psi}_{n,1}$ back into equation (4.1.5), applying the temporal operator ∂_t and grouping terms of equal frequency mode then gives

$$\left(E_n \pm \omega - \hat{H}_0\right) \psi_{n,1}^{(\pm 1)} = z\psi_n. \tag{4.1.7}$$

We have now reduced the first-order dynamic problem into two static problems. Solving equation (4.1.7) in both cases then provides the first-order correction in accordance with equation (4.1.6). The second-order correction is now readily found by substituting $\tilde{\Psi}_{n,1}$ into the $p = 2$ case of equation (4.1.4) and proceeding in a similar fashion. The resulting correction will have a DC contribution and $\pm 2\omega$ contributions. For the scope of this project, we will not need to address the question of normalization as we did for the electrostatic perturbation. The remaining task is now to solve the eigenstate component equations so that the full correction may be found. As with the static problem, we now move forward and consider two different approaches.

4.1.1 Basis Expansion Method

We now turn to the first method for determining the electrodynamic eigenstate corrections, which once again is the basis expansion method. Like in the electrostatic example, we expand the unknown eigenstate corrections in a basis expansion as given in equation (3.1.12). In the electrodynamic case, we instead expand the eigenstate components $\tilde{\psi}_{n,p}^{(m)}$ from which the full corrections are composed. As with the static case, we start by writing

$$\begin{aligned} \tilde{\psi}_{n,p}^{(m)} &= \sum_{m'=1}^{\infty} c_{m',p}^{(m)} \psi_{m'} + \int_0^{\infty} c_{k,p}^{(m)} \psi_k \mathrm{d}k \\ &\equiv \sum\!\!\!\!\!\!\int_{j \in I} c_{j,p}^{(m)} \psi_j. \end{aligned} \tag{4.1.8}$$

Again, we now need to determine the coefficients $c_{j,p}^{(m)}$ by exploiting the orthonormality between the unperturbed eigenstates. Again, we restrict ourselves to the first order correction for this method. The first order equal index coefficient $c_{n,1}^{(m)}$ must, as in the electrostatic case, vanish. So for the $p = 1$ case for instance, we have $c_{n,1}^{(\pm 1)} = 0$. For the remaining set, we substitute the expansion into equation (4.1.7) so that

$$\sum\!\!\!\!\!\!\int_{j \in I} c_{j,p}^{(\pm 1)} \left(E_n \pm \omega - \hat{H}_0\right) \psi_j = z\psi_n. \tag{4.1.9}$$

We now let the Hamiltonian act on the unperturbed eigenstate and employ $\hat{H}_0 \psi_j = E_j \psi_j$. Multiplying the resulting equation through by $\psi_{j'}$ for some $j' \in I \backslash n$ and integrating over $\mathbb{R}_{\geq 0}$ then yields

$$\sum\!\!\!\!\!\!\int_{j \in I} c_{j,p}^{(\pm 1)} \left(E_{nj} \pm \omega\right) s_{j',j}^{0,0} = Z_{j'n}, \tag{4.1.10}$$

where we have interchanged sums and integrals. We know that $s_{j',j}^{0,0} = \delta_{j'j}$, and hence the sum-integral vanishes for every $j \neq j'$. This just leaves the coefficient $c_{j',p}^{(\pm 1)}$, which we can then express as

$$c_{j,p}^{(\pm 1)} = \frac{Z_{jn}}{E_{nj} \pm \omega}. \tag{4.1.11}$$

The first-order eigenstate corrections are then

$$\tilde{\psi}_{n,1}^{(\pm 1)} = \sum\!\!\!\!\!\!\int_{\substack{j \in I \\ j \neq n}} \frac{Z_{jn}}{E_{nj} \pm \omega} \psi_j, \tag{4.1.12}$$

which are very similar to those provided in the static case. Again, we may proceed in a similar manner and compute higher-order corrections, although this still requires the unbound-unbound matrix elements. We note that the static result is recovered in the $\omega = 0$ case. We now move onto the second for used to determine the dynamic corrections to the eigenstates.

4.1.2 Sturmian Basis Expansion

We now move to the second method of determining the electrodynamic eigenstate corrections, which we again accomplish by finding the eigenstate components $\tilde{\psi}_{n,p}^{(m)}$. Their governing differential equations, such as the first-order equations given in equation (4.1.7), are very similar to those given in the static case. The addition of the ω terms, however, make the equations much more difficult to solve and unfortunately invalidates the Dalgarno-Lewis approach as presented in the static case. To see why this is, we consider equation (3.1.28). The new ω terms will introduce a non-differentiated field factor term, which in turn no longer allows the left-hand side to be factored using an integrating factor. Constructing the eigenstates directly as a polynomial sum breaks down as well because the sums no longer truncate. Instead, we will for our purposes take inspiration from [Pedersen, 2024] and expand the eigenstate components in a Sturmian basis expansion. While this does not provide closed-form corrections, it does have a number of advantages in comparison with the basis expansion method, as we will see. To showcase the procedure, we again consider the first-order correction equation, which for the $+\omega$ solution reads

$$\left(E_n + \omega - \hat{H}_0\right) \psi_{n,1}^{(1)} = z\psi_n. \tag{4.1.13}$$

We start by exploiting the fact that the Sturmian basis functions make a complete, discrete set. We may then expand the eigenstate component in an expansion of the form

$$\psi_{n,1}^{(1)} = \sum_{m=1}^{\infty} C_{nm}^{(1)} \varphi_m. \tag{4.1.14}$$

Substituting the expansion along with the unperturbed Hamiltonian into equation (4.1.13) then yields

$$\sum_{m=1}^{\infty} C_{nm}^{(1)} \left(E_n + \omega + \frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} + \frac{1}{z} + \frac{\delta(1-\delta)}{2z^2}\right) \varphi_m = z\psi_n. \tag{4.1.15}$$

We now exploit the governing differential equation for the Sturmian basis functions as given in equation (3.4.15) for $\lambda_m = m\kappa$, which allows us to eliminate the kinetic energy operator and the non-Coulombic potential. This in turn gives

$$\sum_{m=1}^{\infty} C_{nm}^{(1)} \left(E_n + \omega + \frac{1}{2}\kappa^2 + \frac{1}{z} - \frac{m\kappa}{z}\right) \varphi_m = z\psi_n. \tag{4.1.16}$$

We now choose the Sturmian parameter such that $-\kappa^2/2 = E_n + \omega$, which in turn reduces the above equation to

$$\sum_{m=1}^{\infty} C_{nm}^{(1)} \left(1 - m\kappa\right) \frac{1}{z} \varphi_m = z\psi_n. \tag{4.1.17}$$

We can now determine the coefficients $C_{nm}^{(1)}$ by exploiting the orthogonality relation given in equation (3.4.10). Multiplying equation (4.1.17) through by φ_k for some $k \in \mathbb{Z}_{>0}$ and integrating over $\mathbb{R}_{\geq 0}$ then yields

$$\sum_{m=1}^{\infty} C_{nm}^{(1)} \left(1 - m\kappa\right) \delta_{km} = \left\langle \varphi_k \left| z \right| \psi_n \right\rangle. \tag{4.1.18}$$

The terms in the above sum now vanish for every $m \neq k$. This leaves only one coefficient, which we in turn may express as

$$C_{nm}^{(1)} = \frac{\left\langle \varphi_m \left| z \right| \psi_n \right\rangle}{1 - m\kappa}. \tag{4.1.19}$$

The first-order correction is then

$$\psi_{n,1}^{(1)} = \sum_{m=1}^{\infty} \frac{\langle \varphi_m \left| z \right| \psi_n \rangle}{1 - \mathscr{m}\kappa} \varphi_m. \tag{4.1.20}$$

The other eigenstate component is then found by letting $\omega \to -\omega$. Because the Sturmian functions have the same structure as the unperturbed eigenstates, we may calculate the matrix elements $\langle \varphi_m \left| z \right| \psi_n \rangle$ in terms of $\mathrm{J}_{2-2\delta}^{2,0}$ W. Gordon integrals. In particular, we find

$$\begin{aligned} \langle \varphi_m \left| z \right| \psi_n \rangle &= \int_0^\infty \mathcal{N}_n \mathcal{M}_m z^{3-2\delta} \mathrm{e}^{-(\mathscr{n}^{-1}+\kappa)z} {}_1F_1\left(1-n, 2-2\delta, 2\mathscr{n}^{-1} z\right) {}_1F_1\left(1-m, 2-2\delta, 2\kappa z\right) \mathrm{d}z \\ &= \mathcal{N}_n \mathcal{M}_m \mathrm{J}_{2-2\delta}^{2,0} \left(1-n, 1-m, 2\mathscr{n}^{-1}, 2\kappa\right), \end{aligned} \tag{4.1.21}$$

where $\mathcal{M}_m = (2\kappa)^{1-\delta} (m)_{1-2\delta}^{1/2} / \Gamma(2-2\delta)$ as given in equation (3.4.13). Using the procedure described in sections 2.1.1 and 2.2.1 to compute the $\mathrm{J}_{2-2\delta}^{2,0}$ integral, we eventually find that

$$\begin{aligned} \langle \varphi_m \left| z \right| \psi_n \rangle = & \frac{2(-1)^n \mathscr{n}^{4-2\delta} \mathcal{N}_n \mathcal{M}_m \Gamma(2-2\delta)}{(1-\kappa\mathscr{n})^{4-2\delta}} \left(\frac{1-\kappa\mathscr{n}}{1+\kappa\mathscr{n}}\right)^{\mathscr{m}+\mathscr{n}} \left\{(m-1)\right. \\ & \times {}_2F_1\left(1-n, 2-m, 2-2\delta, \frac{-4\kappa\mathscr{n}}{(1-\kappa\mathscr{n})^2}\right) - \frac{\mathscr{m} + 2\mathscr{n}^2 + (2\mathscr{m}+1)\mathscr{m}\mathscr{n}^2\kappa^2 - 2(2\mathscr{m}+1)\mathscr{n}^2\kappa}{(1+\kappa\mathscr{n})^2} \\ & \times {}_2F_1\left(1-n, 1-m, 2-2\delta, \frac{-4\kappa\mathscr{n}}{(1-\kappa\mathscr{n})^2}\right). \end{aligned} \tag{4.1.22}$$

For a general n, the matrix elements are quite complicated. In the case of the ground state and first few excited states however, these reduce significantly. The first order correction comes appropriately normalized in the presented form. This concludes the second approach and allows us to now move forward and apply the presented results.

4.2 Dynamic Linear Polarizability

In this section, we aim to apply the results of the previous sections and calculate the more general dynamic linear polarizability, for which the static polarizabilities found in section 3.2 are special cases. The starting point is once again the dipole moment $\tilde{\mathbf{p}}_n$, which we now write as

$$\tilde{\mathbf{p}}_n = -\langle \tilde{\Psi}_n \left| z \right| \tilde{\Psi}_n \rangle \hat{\mathbf{z}}. \tag{4.2.1}$$

We now expand $\tilde{\Psi}_n$ in a perturbative series and extract the dynamic polarizabilities in a similar fashion to the static case. The first-order contributions are then

$$\begin{aligned} \tilde{\mathbf{p}}_{n,1} &= -\langle \tilde{\Psi}_{n,0} \left| z \right| \tilde{\Psi}_{n,1} \rangle \hat{\mathbf{z}} - \langle \tilde{\Psi}_{n,1} \left| z \right| \tilde{\Psi}_{n,0} \rangle \hat{\mathbf{z}} \\ &= -2\mathrm{Re} \langle \tilde{\Psi}_{n,0} \left| z \right| \tilde{\Psi}_{n,1} \rangle \hat{\mathbf{z}} \\ &= -2\mathrm{Re} \langle \psi_n \mathrm{e}^{-iE_n t} \left| z \right| \frac{1}{2} \mathcal{E}_0 \psi_{n,1}^{(1)} \mathrm{e}^{-i(E_n+\omega)t} + \frac{1}{2} \mathcal{E}_0 \psi_{n,1}^{(-1)} \mathrm{e}^{-i(E_n-\omega)t} \rangle \hat{\mathbf{z}} \\ &= -\langle \psi_n \left| z \right| \psi_{n,1}^{(1)} + \psi_{n,1}^{(-1)} \rangle \mathcal{E}_0 \cos\omega t \hat{\mathbf{z}}. \end{aligned} \tag{4.2.2}$$

The linear dynamic polarizability is then conveniently expressed as

$$\tilde{\alpha}_{n,1}(\omega) = -\langle \psi_n \left| z \right| \psi_{n,1}^{(1)} \rangle + (\omega \to -\omega). \tag{4.2.3}$$

We may now evaluate equation (4.2.3) using either of the two expansions given in equations (4.1.12) and (4.1.20). Starting with the basis expansion eigenstate, we find

$$\begin{aligned}\tilde{\alpha}_{n,1}(\omega) &= -\langle\psi_n|z|\sum\!\!\!\!\!\!\!\int_{\substack{j\in I\\ j\neq n}} \frac{Z_{jn}}{E_{nj}+\omega}\psi_j\rangle + (\omega\to-\omega)\\ &= -\sum\!\!\!\!\!\!\!\int_{\substack{j\in I\\ j\neq n}} \frac{Z_{jn}Z_{nj}}{E_{nj}+\omega} + (\omega\to-\omega)\\ &= -\sum\!\!\!\!\!\!\!\int_{\substack{j\in I\\ j\neq n}} \frac{g_{jn}}{2E_{nj}\left(E_{nj}+\omega\right)} + (\omega\to-\omega)\\ &= -\sum\!\!\!\!\!\!\!\int_{\substack{j\in I\\ j\neq n}} \frac{g_{jn}}{E_{nj}^2-\omega^2}. \end{aligned} \tag{4.2.4}$$

As required, we see that equation (4.2.4) reduces to $\tilde{\alpha}_{n,1}(0) = \mathcal{G}_{n,-2}$ in the static limit. As with the static case, the sum-integral in equation (4.2.4) is impractical to evaluate in closed form even in the ground state. Fortunately, we can, for small n, evaluate the linear dynamic polarizability in closed form by instead using the Sturmian basis expansion, which when substituted into equation (4.2.3) yields

$$\begin{aligned}\tilde{\alpha}_{n,1}(\omega) &= -\langle\psi_n|z|\sum_{m=1}^{\infty}\frac{\langle\varphi_m|z|\psi_n\rangle}{1-m\kappa}\varphi_m\rangle + (\omega\to-\omega)\\ &= \sum_{m=1}^{\infty}\frac{|\langle\varphi_m|z|\psi_n\rangle|^2}{m\kappa-1} + (\omega\to-\omega)\,. \end{aligned} \tag{4.2.5}$$

We may now use equation (4.1.22) to compute the matrix elements $\langle\varphi_m|z|\psi_n\rangle$ for a given n. Starting with the Coulomb ground state, we eventually find

$$\frac{|\langle\varphi_m|z|\psi_1\rangle|^2}{m\kappa-1} = \frac{64m\kappa^2\left[(1+2m^2)\kappa^2-6m\kappa+3\right]^2}{(1-\kappa^2)^6(1-m\kappa)}\left(\frac{1-\kappa}{1+\kappa}\right)^{2m} \tag{4.2.6}$$

At this point, we need only sum the contributions of equation (4.2.6) over all m to acquire the polarizability. This may be accomplished by using series identities of the form

$$\sum_{m=1}^{\infty}\frac{m}{1-m\kappa}\left(\frac{1-\kappa}{1+\kappa}\right)^{2m} = \frac{1-\kappa}{(1+\kappa)^2}\,{}_2F_1\left(2,\frac{\kappa-1}{\kappa},\frac{2\kappa-1}{\kappa},\left(\frac{1-\kappa}{1+\kappa}\right)^2\right). \tag{4.2.7}$$

Increasing the power of m in the numerator on the left-hand side of equation (4.2.7) by one simply places an additional factor $(2)_k/(1)_k$ in the hypergeometric function on the right-hand side. The ground state contributions given in equation (4.2.6) will then amount to several of these series when summing over m. The resulting expression in full may then be reduced significantly in for instance Mathmatica. Carrying out the resulting calculations for the ground state then shows that

$$\begin{aligned}\tilde{\alpha}_{1,1}(\omega) &= \sum_{m=1}^{\infty}\frac{64m\kappa^2\left[(1+2m^2)\kappa^2-6m\kappa+3\right]^2}{(1-\kappa^2)^6(1-m\kappa)}\left(\frac{1-\kappa}{1+\kappa}\right)^{2m} + (\omega\to-\omega)\\ &= \frac{16-2\kappa(1-\kappa)^2(3\kappa^2+9\kappa+8)}{(1-\kappa)^3(1+\kappa)^4} - \frac{64\kappa}{(1-\kappa)^3(1+\kappa)^6}\\ &\quad\times{}_2F_1\left(1,\frac{\kappa-1}{\kappa},\frac{2\kappa-1}{\kappa},\left(\frac{1-\kappa}{1+\kappa}\right)^2\right) + (\omega\to-\omega)\,, \end{aligned} \tag{4.2.8}$$

where we for the ground state have $\kappa^2 = 1-2\omega$ and find that $\tilde{\alpha}_{1,1}(0) = 3$ indeed is satisfied. This approach may in principle be extended to the excited states, although the resulting expressions quickly become complicated. This is exemplified by the $n = 2$ case, for which one finds that

$$\frac{|\langle\varphi_m|z|\psi_2\rangle|^2}{m\kappa-1} = \left[16(2m^2+1)\kappa^4 - 16(2m^2+7)m\kappa^3 + 8(11m^2+7)\kappa^2 - 60m\kappa + 9\right]^2 \times \frac{2048m\kappa^2}{(1-4\kappa^2)^8(m\kappa-1)}\left(\frac{1-2\kappa}{1+2\kappa}\right)^{2m}. \tag{4.2.9}$$

Summing the contributions over all m then eventually yields

$$\tilde{\alpha}_{2,1}(\omega) = \sum_{j=1}^{7} \frac{6144\kappa^2 a_j}{(j\kappa-1)(1-4\kappa^2)^8}\left(\frac{1-2\kappa}{1+2\kappa}\right)^{2m} {}_2F_1\left(1+j, \frac{j\kappa-1}{\kappa}, \frac{(1+j)\kappa-1}{\kappa}, \left(\frac{1-2\kappa}{1+2\kappa}\right)^2\right) + (\omega \to -\omega), \tag{4.2.10}$$

where the coefficients a_j are given by

$$\begin{aligned}
a_1 &= 3(3-2\kappa)^2(1-2\kappa)^6,\\
a_2 &= 48\kappa(2\kappa-5)(1-2\kappa)^4(8\kappa^2-18\kappa+3),\\
a_3 &= 128\kappa^2(1-2\kappa)^2(104\kappa^4-808\kappa^3+1694\kappa^2-720\kappa+81),\\
a_4 &= 1024\kappa^3(80\kappa^5-1104\kappa^4+3800\kappa^3-3072\kappa^2+905\kappa-87),\\
a_5 &= 2560\kappa^4(16\kappa^4-480\kappa^3+2440\kappa^2-1320\kappa+181),\\
a_6 &= 122880\kappa^5(42\kappa-4\kappa^2-11),\\
a_7 &= 1720320\kappa^6,
\end{aligned} \tag{4.2.11}$$

for which $\kappa^2 = 1/4 - 2\omega$. Again, this correctly reproduces the static limit $\tilde{\alpha}_{2,1}(0) = 132$. To showcase how the polarizability depends on the frequency of the externally applied field, we plot the polarizability of the Coulomb potential for the ground state and first excited state. To account for absorption, we add a phenomenological line broadening by introducing a complex frequency $\omega \to \omega + i\Gamma$, which in turn regularizes the dynamic polarizability at resonance. This adds an imaginary part to the polarizability, which characterizes the absorption. The ground state is shown in Figure 4.2 below.

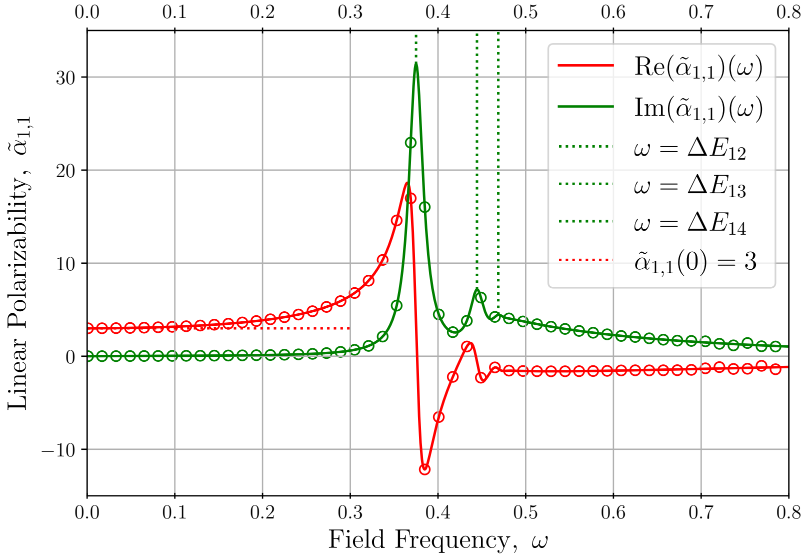

Figure 4.2: The dynamic linear polarizability for the ground state of the Coulomb potential is plotted as a function of the frequency of the applied field. The real part is plotted in red, whilst the imaginary part is plotted in green. The solid line shows the analytical expression provided by Sturmian expansion, whilst the circles show the numerical expression given by the expansion of the unperturbed states. The phenomenological line broadening $\Gamma = 0.01$ is chosen. The two are in good agreement. The red horizontal dotted line showcases the static limit $\tilde{\alpha}_{1,1}(0) = 3$. The resonance frequencies $\Delta E_{12} = 3/8$, $\Delta E_{13} = 4/9$ and $\Delta E_{14} = 15/32$ are marked with vertical dotted lines, at which the absorption peaks accordingly.

Figure 4.2 above shows the dynamic linear polarizability of the Coulomb ground state plotted as a function of the frequency of the applied field as calculated both using the analytical solution given in equation (4.2.8) and the numerical solution provided by equation (4.2.4). The two are in good agreement and the static limit is accurately reproduced. The resonance frequencies corresponding to $\Delta E_{12} = 3/8$, $\Delta E_{13} = 4/9$ and $\Delta E_{14} = 15/32$ can be seen as peaks in the imaginary part. The remaining resonances are not visible with the chosen line broadening $\Gamma = 0.01$. The remaining bound-bound transitions are smeared out up until $\omega = 1/2$, from where the bound-unbound transitions are situated. An important technical point to note is that κ acquires an imaginary part once the frequency of the external field becomes larger than the binding energy. If not for the introduction of the line broadening, we would find a purely imaginary κ and immediate divergence of all the transition contributions beyond $\omega = 1/2$ due to the $\mathrm{e}^{-2\kappa z}$ term in the integrand. The real part of κ, although positive, still decreases with increasing ω and hence the continuum contributions are unbounded as $\omega \longrightarrow \infty$. This issue is only handled once the closed-form expression in terms of hypergeometric functions is acquired, meaning that simply truncating the sum in equation (4.2.5) at some large, finite N will not work. We can do the same for the first excited state, whose dynamic polarizability is plotted in Figure 4.3 below.

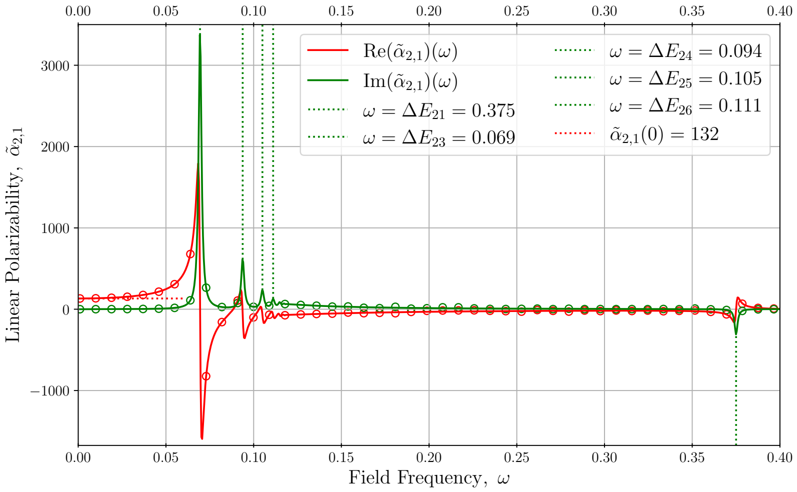

Figure 4.3: The dynamic linear polarizability for the first excited state of the Coulomb potential is plotted as a function of the frequency of the applied field. The real part is plotted in red, whilst the imaginary part is plotted in green. The solid line shows the analytical expression provided by Sturmian expansion, whilst the circles show the numerical expression given by the expansion of the unperturbed states. The phenomenological line broadening $\Gamma = 0.001$ is chosen. The two are in good agreement. The red horizontal dotted line showcases the static limit $\tilde{\alpha}_{2,1}(0) = 132$. The resonance frequencies at $\Delta E_{21} = 3/8$, $\Delta E_{23} = 5/72$, $\Delta E_{24} = 3/32$, $\Delta E_{25} = 21/200$ and $\Delta E_{26} = 1/9$ are marked with vertical dotted lines, at which the absorption peaks accordingly.

Figure 4.3 above again shows the dynamic linear polarizability but now for the first excited state. It is calculated analytically and numerically, both of which are in agreement. The smaller chosen line broadening $\Gamma = 0.001$ illuminates a larger number of resonance frequencies as peaks in the imaginary part. The contribution stemming from the resonance at $\omega = \Delta E_{21}$ reappears with opposing sign. The shown procedure of finding closed form expressions for the linear dynamic polarizability may also be generalized to the Kratzer potential. Starting with the ground state and using equation (4.1.22), one finds that

$$\begin{aligned}\frac{|\langle\varphi_m|z|\psi_1\rangle|^2}{m\kappa-1} &= \frac{2^{6-4\delta}(1-\delta)^{6-2\delta}\kappa^{2-2\delta}}{(1-(1-\delta)\kappa)^6\,(1+(1-\delta)\kappa)^{6-4\delta}\,\Gamma(2-2\delta)} \\ &\times \frac{\left[(1-\delta)(2m^2+1-\delta)\kappa^2+(3-2\delta)(1-2m\kappa)\right]^2 (m)_{1-2\delta}}{1-m\kappa}\left(\frac{1-(1-\delta)\kappa}{1+(1-\delta)\kappa}\right)^{2m},\end{aligned} \tag{4.2.12}$$

which in the $\delta = 0$ case returns equation (4.2.6). Then, by summing in accordance with equation (4.2.5), one finds that

$$\begin{aligned}\tilde{\alpha}_{1,1}(\omega) &= \sum_{j=1}^{2}\frac{2^{4-4\delta}(1-\delta)^{6-2\delta}\kappa^{2-2\delta}a_j}{(1-(1-\delta)\kappa)^4(1+(1-\delta)\kappa)^{7-4\delta}} \\ &\times {}_2F_1\left(j+1-2\delta,\frac{(1-\delta)\kappa-1}{\kappa},\frac{(2-\delta)\kappa-1}{\kappa},\left(\frac{1-(1-\delta)\kappa}{1+(1-\delta)\kappa}\right)^2\right)+(\omega\to-\omega),\end{aligned} \tag{4.2.13}$$

where the coefficients a_j are

$$\begin{aligned}a_1 &= (1+(1-\delta)\kappa)^3(2(1-\delta)(2-\delta)(3-2\delta)\kappa-(1-\delta)^2(2-\delta)(3-2\delta)\kappa^2+7\delta-4-2\delta^2), \\ a_2 &= 2(1-\delta)((1-\delta)(2-\delta)^2\kappa^2+\delta)((1-\delta)^2(3-2\delta)\kappa^2+2\delta-7)\kappa.\end{aligned} \tag{4.2.14}$$

The dynamic polarizability given in equation (4.2.13) correctly reproduces those given in equation (3.2.34) in the static limit. This can also be done for the first excited state, which yields a formula similar to equation

(4.2.10). It is shown in section F. The dependence of the dynamic linear polarizability on the quantum defect parameter δ is shown in Figure 4.4 below.

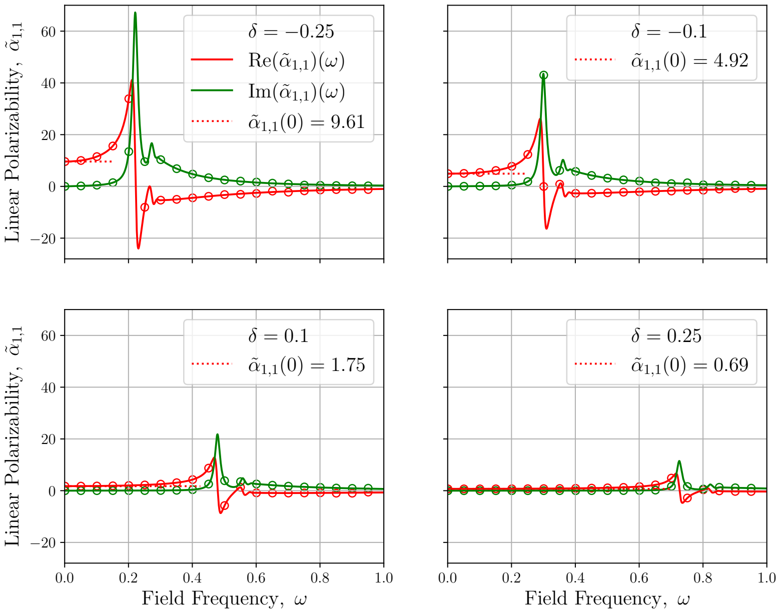

Figure 4.4: The dynamic linear polarizability for the ground state of the Kratzer potential is plotted as a function of the frequency of the applied field for several quantum defect parameters δ. The real part is plotted in red, whilst the imaginary part is plotted in green. The solid line shows the analytical expression provided by Sturmian expansion, whilst the circles show the numerical expression given by the expansion of the unperturbed states. The phenomenological line broadening $\Gamma = 0.01$ is chosen. The static polarizabilities are accurately reproduced in accordance with equation (3.2.34).

Figure 4.4 above shows the dynamic linear polarizability of the Kratzer ground state plotted as a function of the frequency of the applied field for a variety of quantum defect parameters δ. The analytical and numerical approach remain in agreement and the resonance frequencies are shifted appropriately in accordance with the quantum defect energies. The polarizabilities given by equation (3.2.34) are also reproduced in the static limit as required. As the quantum defect parameter δ increases, we find that the magnitude of the response decreases. This also extends to the first excited state, which is shown in Figure 4.5 below.

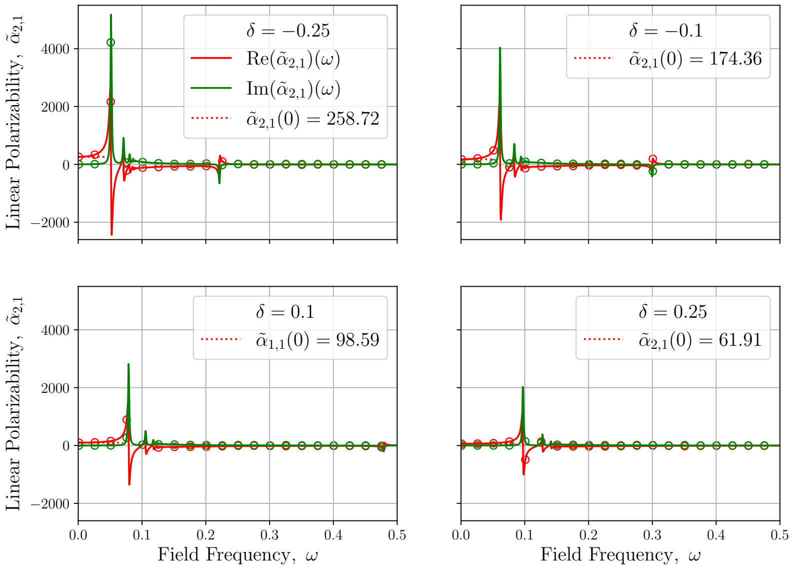

Figure 4.5: The dynamic linear polarizability for the first excited state of the Kratzer potential is plotted as a function of the frequency of the applied field for several quantum defect parameters δ. The real part is plotted in red, whilst the imaginary part is plotted in green. The solid line shows the analytical expression provided by Sturmian expansion, whilst the circles show the numerical expression given by the expansion of the unperturbed states. The phenomenological line broadening $\Gamma = 0.001$ is chosen. The static polarizabilities are accurately reproduced in accordance with equation (3.2.34).

Figure 4.5 above shows the dynamic linear polarizability of the first Kratzer excited state plotted as a function of the frequency of the applied field for a variety of quantum defect parameters δ. As with the ground state, the static polarizabilities are reproduced correctly and both the analytical and numerical approaches are in agreement. In addition, the resonance frequencies may be shifted by altering the quantum defect parameter, although in comparison with the ground state, one finds that the window in which these shifts occur is significantly smaller. This concludes this section on the dynamic linear polarizability and allows us to move forward and investigate other responses induced by the external field.

4.3 Electro-Optic Response

We now aim to expand the degree to which the electron qubits may be controlled by perturbing the platform with both an electrostatic and electrodynamic field. This is known as the electro-optic response of the system in the presence of two perturbations. The first order response induced by this is called the Pockels effect [Boyd, 2008]. Describing it requires us to expand the framework introduced in section 4.1. To do this, we start by introducing the static field $\mathcal{E}_{\mathrm{DC}}\hat{\mathbf{z}}$ and the dynamic field $\mathcal{E}_{\mathrm{AC}}\cos(\omega t)\hat{\mathbf{z}}$ as shown in Figure 4.6 below.

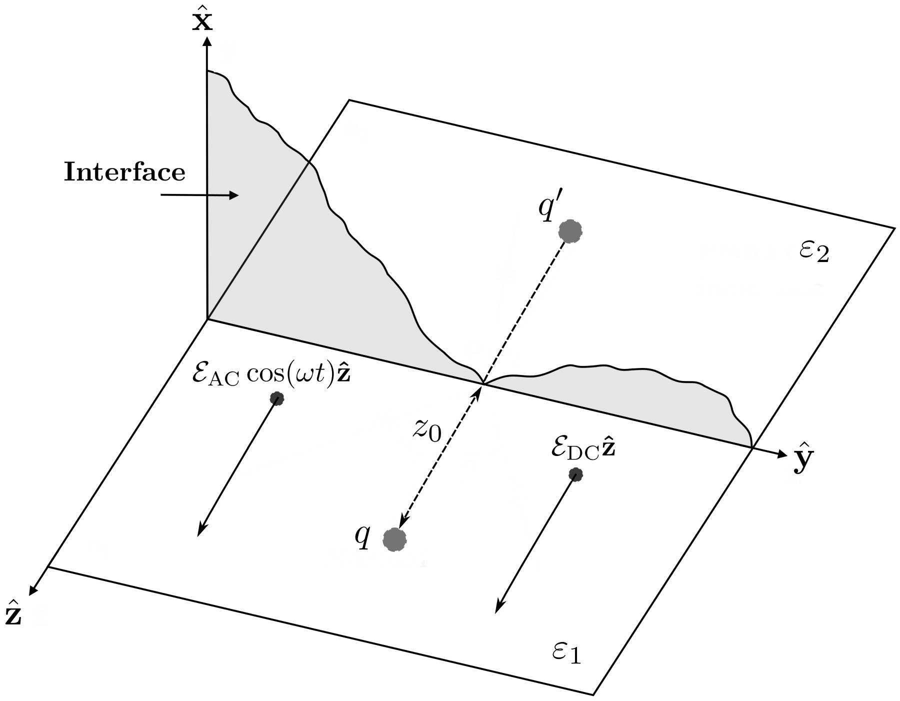

Figure 4.6: Electron-on-Helium qubit system perturbed by the electrostatic field $\boldsymbol{\mathcal{E}} = \mathcal{E}_0\hat{\mathbf{z}}$, which is chosen so that it points away from the substrate. The figure is based on [Pedrotti et al., 2018].

We now proceed in a similar manner to the electrodynamic perturbation. In the dipole approximation, the corresponding contributions to the Hamiltonian are $\mathcal{E}_{\mathrm{DC}}z$ and the dynamic field $\mathcal{E}_{\mathrm{AC}}\cos(\omega t)z$. In the presence of both the static and dynamic field therefore, we may write the full perturbed problem as

$$\left\{\hat{H}_0 + \frac{1}{2}\mathcal{E}_{\mathrm{AC}}z\mathrm{e}^{i\omega t} + \frac{1}{2}\mathcal{E}_{\mathrm{AC}}z\mathrm{e}^{-i\omega t} + \mathcal{E}_{\mathrm{DC}}z\right\}\tilde{\Psi}_n = i\partial_t\tilde{\Psi}_n. \tag{4.3.1}$$

We now write the eigenstate in terms of a perturbative expansion once more. The expansion will in this case not only include corrections in the static and dynamic fields individually, but also mixed corrections in both. To account for this, we expand the eigenstate as

$$\begin{aligned}\tilde{\Psi}_n &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty}\tilde{\Psi}_{n,ij} \\ &= \tilde{\Psi}_{n,00} + \tilde{\Psi}_{n,10} + \tilde{\Psi}_{n,01} + \tilde{\Psi}_{n,11} + \dots,\end{aligned} \tag{4.3.2}$$

where i denotes the perturbative order in the electrostatic field and j denotes the perturbative order in the electrodynamic field. The corrections $\tilde{\Psi}_{n,10}$, $\tilde{\Psi}_{n,01}$ and $\tilde{\Psi}_{n,11}$ then account for the linear response of the static field, the linear response of the dynamic field and the linear response of both fields, respectively. Substituting the expansion into equation (4.3.1) and rearranging then yields

$$\left\{\hat{H}_0 - i\partial_t + \frac{1}{2}\mathcal{E}_{\mathrm{AC}}z\mathrm{e}^{i\omega t} + \frac{1}{2}\mathcal{E}_{\mathrm{AC}}z\mathrm{e}^{-i\omega t} + \mathcal{E}_{\mathrm{DC}}z\right\}\left\{\tilde{\Psi}_{n,00} + \tilde{\Psi}_{n,10} + \tilde{\Psi}_{n,01} + \tilde{\Psi}_{n,11} + \dots\right\} = 0. \tag{4.3.3}$$

We now group terms of equal order and use the strategy described in section 4.1 to deduce the form of each solution. The zeroth order terms once again return the unperturbed problem, which is solved by $\tilde{\Psi}_{n,00} = \psi_n\mathrm{e}^{-iE_nt}$. Next, by grouping terms linear in the dynamic perturbation we find that

$$\left(\hat{H}_0 - i\partial_t\right)\tilde{\Psi}_{n,01} + \frac{1}{2}\mathcal{E}_{\mathrm{AC}}z\tilde{\Psi}_{n,00}\mathrm{e}^{i\omega t} + \frac{1}{2}\mathcal{E}_{\mathrm{AC}}z\tilde{\Psi}_{n,00}\mathrm{e}^{-i\omega t} = 0, \tag{4.3.4}$$

which upon substitution of $\tilde{\Psi}_{n,00} = \psi_n \mathrm{e}^{-iE_n t}$ is identical to the first order dynamic problem given in equation (4.1.5). Using equation (4.1.6), it is then clear a solution of the above takes the form

$$\tilde{\Psi}_{n,01} = \frac{1}{2}\mathcal{E}_{\mathrm{AC}}\psi^{(1)}_{n,01}\mathrm{e}^{-i(E_n+\omega)t} + \frac{1}{2}\mathcal{E}_{\mathrm{AC}}\psi^{(-1)}_{n,01}\mathrm{e}^{-i(E_n-\omega)t}, \tag{4.3.5}$$

where the corrections $\psi^{(1)}_{n,01}, \psi^{(-1)}_{n,01}$ again satisfy

$$\left(E_n \pm \omega - \hat{H}_0\right)\psi^{(\pm 1)}_{n,01} = z\psi_n. \tag{4.3.6}$$

The corrections solving equation (4.3.6) are then found using the same techniques described in sections 4.1.1 and 4.1.2. We now move on to grouping the terms linear only in the static field, which yields

$$\left(\hat{H}_0 - i\partial_t\right)\tilde{\Psi}_{n,10} + \mathcal{E}_{\mathrm{DC}} z \tilde{\Psi}_{n,00} = 0. \tag{4.3.7}$$

By substituting the unperturbed eigenstates $\tilde{\Psi}_{n,00} = \psi_n \mathrm{e}^{-iE_n t}$ into equation (4.3.7), we then quickly find that the solution has the form

$$\tilde{\Psi}_{n,10} = \mathcal{E}_{\mathrm{DC}}\tilde{\psi}^{(0)}_{n,10}\mathrm{e}^{-iE_n t}, \tag{4.3.8}$$

from which the correction $\tilde{\psi}^{(0)}_{n,10}$ can be shown to satisfy

$$\left(E_n - \hat{H}_0\right)\tilde{\psi}^{(0)}_{n,10} = z\psi_n. \tag{4.3.9}$$

Equation (4.3.9) is clearly the static limit of the first order dynamic problem given in equation (4.3.6). As such, we know that $\tilde{\psi}^{(0)}_{n,10}$ is the $\omega = 0$ case of the first order dynamic correction $\psi^{(1)}_{n,01}$. Alternatively, it may be found in closed form using the Dalgarno-Lewis approach. In fact, it is in effect the first-order correction of the purely static problem in the $\tilde{E}_{n,1} = 0$ case. The final correction we will need is acquired by grouping terms linear in both fields, which gives

$$\left(\hat{H}_0 - i\partial_t\right)\tilde{\Psi}_{n,11} + \mathcal{E}_{\mathrm{DC}} z \tilde{\Psi}_{n,01} + \frac{1}{2}\mathcal{E}_{\mathrm{AC}} z \tilde{\Psi}_{n,10}\mathrm{e}^{i\omega t} + \frac{1}{2}\mathcal{E}_{\mathrm{AC}} z \tilde{\Psi}_{n,10}\mathrm{e}^{-i\omega t} = 0. \tag{4.3.10}$$

We now substitute the corrections given in equations (4.3.5) and (4.3.8) into the problem above. Reducing the resulting expression then shows that a solution must be

$$\tilde{\Psi}_{n,11} = \frac{1}{2}\mathcal{E}_{\mathrm{DC}}\mathcal{E}_{\mathrm{AC}}\psi^{(1)}_{n,11}\mathrm{e}^{-i(E_n+\omega)t} + \frac{1}{2}\mathcal{E}_{\mathrm{DC}}\mathcal{E}_{\mathrm{AC}}\psi^{(-1)}_{n,11}\mathrm{e}^{-i(E_n-\omega)t}, \tag{4.3.11}$$

where the corrections $\psi^{(1)}_{n,11}, \psi^{(-1)}_{n,11}$ satisfy

$$\left(E_n \pm \omega - \hat{H}_0\right)\psi^{(\pm 1)}_{n,11} = z\tilde{\psi}^{(0)}_{n,10} + z\psi^{(\pm 1)}_{n,01}. \tag{4.3.12}$$

The solutions of equation (4.3.12) could in principle be found using the techniques described in sections 4.1.1 and 4.1.2, but will not be needed for the purposes of the project. With the first order corrections now acquired, we may move on to calculating the resulting response due to the presence of both fields by expanding the dipole moment. The inclusion of a static field introduces additional contributions to the polarizability. Substituting the original expansion in equation (4.3.2) into the dipole moment then yields

$$\tilde{\mathbf{p}}_n = -\langle\tilde{\Psi}_{n,00} + \tilde{\Psi}_{n,10} + \tilde{\Psi}_{n,01} + \tilde{\Psi}_{n,11} + \ldots\left|z\right|\tilde{\Psi}_{n,00} + \tilde{\Psi}_{n,10} + \tilde{\Psi}_{n,01} + \tilde{\Psi}_{n,11} + \ldots\rangle\hat{\mathbf{z}}. \tag{4.3.13}$$

As before, we now group terms of equal perturbative order. The correction to the dipole moment of particular interest in this section is $\tilde{\mathbf{p}}_{n,11}$, meaning the correction linear in both perturbations. This grouping becomes

$$\begin{aligned}\tilde{\mathbf{p}}_{n,11} &= -\left[\langle\tilde{\Psi}_{n,00}\left|z\right|\tilde{\Psi}_{n,11}\rangle + \langle\tilde{\Psi}_{n,11}\left|z\right|\tilde{\Psi}_{n,00}\rangle + \langle\tilde{\Psi}_{n,01}\left|z\right|\tilde{\Psi}_{n,10}\rangle + \langle\tilde{\Psi}_{n,10}\left|z\right|\tilde{\Psi}_{n,01}\rangle\right]\hat{\mathbf{z}} \\ &= -2\mathrm{Re}\left[\langle\tilde{\Psi}_{n,00}\left|z\right|\tilde{\Psi}_{n,11}\rangle + \langle\tilde{\Psi}_{n,10}\left|z\right|\tilde{\Psi}_{n,01}\rangle\right]\hat{\mathbf{z}} \\ &= -\left[\langle\psi_n\left|z\right|\tilde{\psi}^{(1)}_{n,11}\rangle + \langle\tilde{\psi}^{(0)}_{n,10}\left|z\right|\tilde{\psi}^{(1)}_{n,01}\rangle\right]\mathcal{E}_{\mathrm{DC}}\mathcal{E}_{\mathrm{AC}}\cos(\omega t)\hat{\mathbf{z}} + (\omega \to -\omega)\,,\end{aligned} \tag{4.3.14}$$

where we in the last step have substituted the unperturbed eigenstate and corrections given in equations (4.3.5), (4.3.8) and (4.3.11) and reduced. The corresponding polarizability induced with this response is then conveniently expressed as

$$\tilde{\alpha}_{n,11}(\omega) = \langle\psi_n\,|z|\,\tilde{\psi}^{(1)}_{n,11}\rangle + \langle\tilde{\psi}^{(0)}_{n,10}\,|z|\,\tilde{\psi}^{(1)}_{n,01}\rangle + (\omega \to -\omega)\,. \tag{4.3.15}$$

This polarizability is called the first order electro-optic or Pockels polarizability and is effectively the first order correction of the dynamic polarizability $\tilde{\alpha}_{n,1}$ in response to the static field [Pedersen, 2024]. The electro-optic effect is fundamentally a second order phenomenon [Boyd, 2008], so the static limit must be the first hyperpolarizability given in equation (3.2.36). The expression given in equation (4.3.15) can be brought to a more convenient form by employing a technique similar to the one used to write the third order static energy correction in terms of first order information. We start by multiplying equation (4.3.6) through by $\tilde{\psi}^{(1)}_{n,11}$, equation (4.3.12) through by $\psi^{(1)}_{n,01}$ and integrating both over $\mathbb{R}_{\geq 0}$ so that

$$\langle\tilde{\psi}^{(1)}_{n,11}|E_n + \omega - \hat{H}_0|\psi^{(1)}_{n,01}\rangle = \langle\tilde{\psi}^{(1)}_{n,11}\,|z|\,\psi_n\rangle, \tag{4.3.16}$$

$$\langle\psi^{(1)}_{n,01}|E_n + \omega - \hat{H}_0|\tilde{\psi}^{(1)}_{n,11}\rangle = \langle\psi^{(1)}_{n,01}\,|z|\,\tilde{\psi}^{(0)}_{n,10}\rangle + \langle\psi^{(1)}_{n,01}\,|z|\,\psi^{(1)}_{n,01}\rangle. \tag{4.3.17}$$

By conjugating equation (4.3.16), we may equate it to (4.3.17). This in turn shows that

$$\langle\tilde{\psi}^{(1)}_{n,11}\,|z|\,\psi_n\rangle = \langle\psi^{(1)}_{n,01}\,|z|\,\tilde{\psi}^{(0)}_{n,10}\rangle + \langle\psi^{(1)}_{n,01}\,|z|\,\psi^{(1)}_{n,01}\rangle. \tag{4.3.18}$$

We may then substitute equation (4.3.18) into the electro-optic polarizability given in equation (4.3.15), which then yields

$$\tilde{\alpha}_{n,11}(\omega) = 2\mathrm{Re}\langle\tilde{\psi}^{(0)}_{n,10}\,|z|\,\tilde{\psi}^{(1)}_{n,01}\rangle + \langle\psi^{(1)}_{n,01}\,|z|\,\psi^{(1)}_{n,01}\rangle + (\omega \to -\omega)\,. \tag{4.3.19}$$

By relating $\tilde{\psi}^{(0)}_{n,10}$ to the purely static first order correction $\tilde{\psi}_{n,1}$ then finally allows us to write

$$\tilde{\alpha}_{n,11}(\omega) = 2\mathrm{Re}\langle\tilde{\psi}_{n,1}\,|z - Z_{nn}|\,\tilde{\psi}^{(1)}_{n,1}\rangle + \langle\psi^{(1)}_{n,1}\,|z - Z_{nn}|\,\psi^{(1)}_{n,1}\rangle + (\omega \to -\omega)\,, \tag{4.3.20}$$

where we have substituted the purely dynamic correction $\psi^{(1)}_{n,1}$ given in section 4.1. Equation (4.3.20) provides a form very closely reminiscent of the first static hyperpolarizability as given in equation (3.2.35). Even though the above expression only requires first order information, it is still a second order process and hence is unusable for the basis expansion method because it requires the unbound-unbound matrix elements. Interestingly, it is usable using the Sturmian expansion method and can in principle even be brought to closed form [Pedersen, 2024]. To explain how, we consider the first integral given on the right-hand side of equation (4.3.20). For the ground state, the first order purely static eigenstate correction is given in equation (3.1.50) and upon substitution yields

$$\begin{aligned}\langle\tilde{\psi}_{1,1}\,|z - Z_{11}|\,\tilde{\psi}^{(1)}_{1,1}\rangle = {} & \frac{1}{4}\,(1-\delta)^3\,(2-\delta)\,(3-2\delta)^2\left(\langle\psi_1\,|z|\,\tilde{\psi}^{(1)}_{1,1}\rangle - Z_{11}\langle\psi_1|\tilde{\psi}^{(1)}_{1,1}\rangle\right) \\ & \frac{1}{2}\,(1-\delta)\left(Z_{11}\langle\psi_1\,\left|z^2\right|\tilde{\psi}^{(1)}_{1,1}\rangle - \langle\psi_1\,\left|z^3\right|\tilde{\psi}^{(1)}_{1,1}\rangle\right).\end{aligned} \tag{4.3.21}$$

The integrals given on the right-hand side of equation (4.3.21) can be calculated in the same manner as those required for the linear polarizability in section 4.2 despite requiring higher order multipole moment components, meaning $\mathrm{J}^{3,0}_{2-2\delta}$ and $\mathrm{J}^{4,0}_{2-2\delta}$ integrals. This is accomplished by repeated use of the recurrence relations generated by equation (A.0.12) for the W. Gordon integrals. By reducing the $\mathrm{J}^{3,0}_{2-2\delta}$ and $\mathrm{J}^{4,0}_{2-2\delta}$ integrals down to $\mathrm{J}^{2,0}_{2-2\delta}$, $\mathrm{J}^{1,0}_{2-2\delta}$ and $\mathrm{J}^{0,0}_{2-2\delta}$ integrals then allows us to use the formulas derived in earlier sections to compute the terms in equation (4.3.21). Next, we consider the second integral given in equation

(4.3.20). Taking the Sturmian expansion representation of the purely dynamic first order correction given in equation (4.1.20) and substituting it directly into the second integral yields

$$\langle \psi_{1,1}^{(1)} |z - Z_{11}| \psi_{1,1}^{(1)} \rangle = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \frac{\langle \varphi_n |z| \psi_1 \rangle \langle \varphi_n |z - Z_{11}| \varphi_m \rangle \langle \varphi_m |z| \psi_1 \rangle}{(1 - n\kappa)(1 - m\kappa)}. \tag{4.3.22}$$

The double series given in equation (4.3.22) reduces to a single series upon application of the integrals derived in section 3.4, namely equation (3.4.22) and (3.4.25). The orthonormality between the Laguerre polynomials ensure that the inner m-sum can at most contain five non-zero terms. This ultimately allows the expression to be reduced to a single, albeit complicated, series. Several attempts of finding a closed form expression for $\tilde{\alpha}_{n,11}(\omega)$ was attempted but could not be found because the computational tasks involved in reducing the expressions produced by equations (4.3.21) and (4.3.22) were too expensive. This was even the case in the simplest $\delta = 0$ case and hence the result reported in [Pedersen, 2024] could not be reproduced. This is unfortunate in part because the closed-form expression involving hypergeometric functions is required to capture proper continuum contributions, so an accurate polarizability spectrum cannot be found simply by truncating the series' given in equations (4.3.21) and (4.3.22). For the sake completion, we include the expression provided in [Pedersen, 2024] here, which is given by

$$\begin{aligned} \tilde{\alpha}_{1,11}(\omega) = {} & \frac{9\left(31 + 2\kappa^2 - \kappa^4\right)}{4(1-\kappa^2)^4} + \frac{36\left(\kappa^4 - 8\kappa^2 - 1\right)}{(1-\kappa)^5\,(1+\kappa)^6} \\ & \times \left[{}_2F_1\left(1, 1, \frac{2\kappa - 1}{\kappa}, -\frac{(1-\kappa)^2}{4\kappa}\right) - 2 \right] + (\omega \to -\omega), \end{aligned} \tag{4.3.23}$$

where we again have $\kappa^2 = 1 - 2\omega$. The expression very closely resembles the linear counterpart $\tilde{\alpha}_{1,1}(\omega)$ as given in equation (4.2.8). In order to plot the response, we again introduce the phenomenological line broadening Γ through the complex frequency $\omega \to \omega + i\Gamma$. The resulting plot is shown in Figure 4.7 below, where we have also include the dynamic linear polarizability $-\tilde{\alpha}_{1,1}(\omega)$ for comparison.

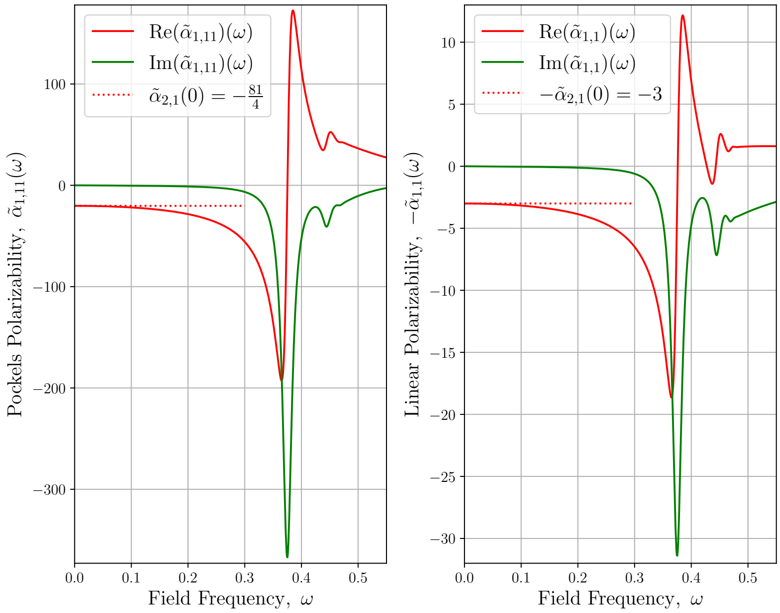

Figure 4.7: The dynamic electro-optic or Pockels polarizability $\tilde{\alpha}_{1,11}(\omega)$ for the ground state of the Coulomb potential is plotted as a function of the frequency of the applied field. The negative dynamic polarizability $-\tilde{\alpha}_{1,1}(\omega)$ is plotted alongside for comparison. The real part is plotted in red, whilst the imaginary part is plotted in green. The solid lines show the analytical expression obtained using the Sturmian expansion and the horizontal dotted lines show the exact static limits provided by equations (3.2.34) and (3.2.36). The phenomenological line broadening $\Gamma = 0.01$ is chosen. The magnitude of the electro-optic response $\tilde{\alpha}_{1,11}(\omega)$ is roughly 10 times greater than the linear response $-\tilde{\alpha}_{1,1}(\omega)$. However, the $1 \longrightarrow 3$ and $1 \longrightarrow 4$ resonances are comparatively weaker for the electro-optic response.

As seen on Figure 4.7, we find that the static limit $\tilde{\alpha}_{1,11}(0) = \tilde{\alpha}_{1,2} = -81/4$ is accurately reproduced in accordance with equation (3.2.36). The magnitude of the electro-optic response is roughly ten times larger than the linear response $\tilde{\alpha}_{1,1}(\omega)$. In comparison with the linear response, we find that the $1 \longrightarrow 3$ and $1 \longrightarrow 4$ resonances of the electro-optic response are visible but are comparatively weaker in magnitude with respect to the main $1 \longrightarrow 2$ resonance. The fundamental resonance structure remains the same. Similarly with the linear polarizability, if the strength of the electrostatic perturbation is sufficiently strong, we will find that higher-order corrections are required to accurately describe the response. The reason for this is that the stronger electrostatic field may shift the resonances and hence dramatically increase the magnitude of the response. This property will also be important for more advanced qubit control because the static field can tune the frequencies at which transitions induced by the dynamic field can occur. This is relatively easily accomplished on the electron-on-Helium qubit platform because the Stark shifts are significant even for lower field strengths. This concludes the present chapter on the electrodynamic perturbation of the electron-on-Helium qubit system.

Chapter 5

Conclusion

We have throughout this thesis investigated the out-of-plane electrostatic and electrodynamic properties of electrons bound to cryogenic substrates by image-charge induced potentials using a number of quantum mechanical perturbation methods. The initial analysis of chapter two involved establishing the system of interest and deriving an expression for the image-charge induced force $\mathbf{F}$ and subsequent Coulomb potential V binding the electron to the cryogenic substrate using the method of images. Next, we moved to the quantum mechanical picture and determined the energy eigenvalues E_n and corresponding eigenstates ψ by constructing and then solving the time-independent Schrödinger equation with the derived image potential for both the bound states ψ_n and the unbound states ψ_k. Using the Coulomb eigenstates, we then computed the diagonal bound-bound Z_{nn}, off-diagonal bound-bound Z_{nm} and bound-unbound dipole matrix elements Z_{nk} along with the corresponding oscillator strengths g_{nm}, g_{nk} and oscillator strength moments $\mathcal{G}_{n,p}$. The calculations were subsequently verified by confirming that the Thomas-Reiche-Kuhn sum rule $\sumint_{j\in I} g_{nj} = 1$ holds. Moving forward, we then phenomenologically introduced the Kratzer potential V_δ as a generalization of the Coulomb potential V by means of the quantum defect parameter δ, which in turn allowed the energy eigenvalues E_n to be fitted to experimental data. The correspondingly modified eigenstates ψ of the Kratzer potential V_δ were then determined, from which the various dipole matrix elements Z_{nn}, Z_{nm} and Z_{nk} were also generalized and the Thomas-Reiche-Kuhn sum rule $\sumint_{j\in I} g_{nj} = 1$ reconfirmed.

With the mechanics of the electron system established, we then moved forward to chapter three, wherein the electrostatic perturbation for the system $\hat{H}_0$ was presented and examined. We started by writing the perturbed eigenstates $\tilde{\psi}_n$ and energy eigenvalues $\tilde{E}_n$ in terms of perturbative expansions in the electric field strength $\mathcal{E}_0$, from which the framework of time-independent perturbation theory could then be introduced. This included expressions for each energy correction $\tilde{E}_{n,p}$, the governing differential equations for each eigenstate correction $\tilde{\psi}_{n,p}$ and their respective normalization criteria. We subsequently described two different procedures for determining these corrections, namely the basis expansion method and the Dalgarno-Lewis approach, the latter of which provided exact, closed-form results. With the corrections readily calculable to high perturbative order for both the Coulomb potential V and Kratzer potential V_δ, we applied the presented techniques and derived a number of lower-order energy correction expressions $\tilde{E}_{n,1}$, $\tilde{E}_{n,2}$, $\tilde{E}_{n,3}$, $\tilde{E}_{n,4}$ for arbitrary principle quantum number n. This in turn also allowed the first few static polarizabilities $\tilde{\alpha}_{n,1}$ and hyperpolarizabilities $\tilde{\alpha}_{n,2}$, $\tilde{\alpha}_{n,3}$ to be determined. With the energy corrections $\tilde{E}_{n,p}$ calculated, it became clear that the perturbative expansion of the energy diverged. As a means of regularizing it, we introduced the hypergeometric approximant $\mathscr{c}_1 {}_2F_1(\mathscr{c}_2, \mathscr{c}_3, \mathscr{c}_4, \mathscr{c}_5\mathcal{E}_0)$ that would also allow us to extract the ionization rates Γ_n. The accuracy of the chosen approximant was then compared with a numerical approach involving a complex-scaled Sturmian basis function expansion $\sum_m c_m\varphi_m$. The resulting calculations then showed good agreement between both approaches for the Stark-shifted energy $\tilde{E}_n$ and the ionization rates Γ_n. A re-estimate of the quantum defect parameter δ was also made by fitting experimental data and the dependence of the Stark-shifted energy $\tilde{E}_n$ on the quantum defect parameter δ was also tested.

In the fourth and final chapter, we instead introduced an electrodynamic perturbation in the form of a harmonic electric field $\mathcal{E}_0 \cos \omega t$. The time-varying perturbation in turn required a new perturbative framework based on the time-dependent Schrödinger equation, which was subsequently introduced by writing the full eigenstates $\tilde{\Psi}_n$ in a perturbative expansion. Like with the electrostatic perturbation, we introduced two methods for determining the eigenstate corrections $\tilde{\Psi}_{n,p}$, which in the electrodynamic case was the basis expansion method and the Sturmian expansion approach. We then employed these two methods to evaluate several electrodynamic response properties of the system under an external field. The first of these was the linear dynamic polarizability $\tilde{\alpha}_{n,1}(\omega)$, which we determined in closed form for the ground state and first excited for both the Coulomb potential V and Kratzer potential V_δ using the Sturmian expansion approach. The analytical results were then checked against the numerical approach provided by the basis expansion method, which showed good agreement and accurately reproduced the polarizability $\tilde{\alpha}_{n,1}$ from the previous chapter in the static limit. Finally, we also introduced a static electric field in addition to the dynamic perturbation, which in turn yielded electro-optic responses. The lowest order polarizability of these responses, the Pockels polarizability $\tilde{\beta}_{n,1}(\omega)$, was subsequently also shown in closed form for the Coulomb ground state and produced the correct static limit.

The analysis of the electrostatic and electrodynamic properties of the proposed qubit platform reveal several promising characteristics. The weakly bound electron states induced by the image-charge effect exhibit great sensitivity to electric field perturbations, which in turn facilitates a high degree of external control; a property that would be important for real-world applications. In the case of electrostatic perturbations, the sensitivity of the electron states is exemplified by the fact that the ionization rates and binding energies are comparable at remarkably low field strengths, and in addition, that the ionization rates increase significantly with rising principle quantum number. The ease at which ionization is induced then allows it to be exploited for practical applications such as state-selective qubit read-out schemes. The application of dynamic perturbations enables for additional qubit manipulation such as induced state transitions, which are characterized by the dynamic, frequency-dependent polarizabilities. Introducing both static and dynamic perturbations simultaneously provides more advanced qubit manipulation because the static field can shift the frequencies at which transitions are induced by the dynamic field. Tuning the static fields is relatively easily accomplished because the qubits' sensitivity to Stark shifts. This speaks to the degree in which the electrons can be electrically controlled and their viability as a potential qubit platform. The presented analysis may be extended by investigating the dynamic non-linear response in more detail. These contributions will inevitably become more important once the strength of the electric field is sufficiently strong. This includes the not only higher order corrections to the purely dynamic perturbation, where higher harmonic generation comes into play, but also higher order corrections to the electro-optic effect. The next correction in that case would be the Kerr electro-optic effect. In the present framework, this would require dealing with the problems involving the unbound-unbound matrix elements. The analysis may also be generalized further to concern problems outside out-of-plane bound states, such as qubit-qubit interaction and the in-plane dynamics caused by the embedded electrodes, which would shine further light on capabilities of the electron-on-Helium platform.

Bibliography

[Abramowitz and Stegun, 1964] Abramowitz, M. and Stegun, I. A. (1964). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, ninth dover printing, tenth gpo printing edition.

[Antonsen, 1999] Antonsen, F. (1999). Sturmian basis functions for the harmonic oscillator. *Physical Review A*, 60(2):812–826.

[Benioff, 1980] Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of Statistical Physics*, 22:563–591.

[Bernardini et al., 2023] Bernardini, F., Chakraborty, A., and Ordóñez, C. (2023). Quantum computing with trapped ions: a beginner's guide.

[Bethe and Salpeter, 1957] Bethe, H. A. and Salpeter, E. E. (1957). *Quantum Mechanics of One- and Two-Electron Atoms*. Springer-Verlag Berlin Heidelberg.

[Boyd, 2008] Boyd, R. W. (2008). *Nonlinear Optics, Third Edition*. Academic Press, Inc., USA, 3rd edition.

[Bruschi et al., 1966] Bruschi, L., Maraviglia, B., and Moss, F. E. (1966). Measurement of a barrier for the extraction of excess electrons from liquid helium. *Phys. Rev. Lett.*, 17:682–684.

[Burden et al., 2015] Burden, R., Faires, J., and Burden, A. (2015). *Numerical Analysis*. Cengage Learning.

[Chen et al., 2022] Chen, Q., Martin, I., Jiang, L., and Jin, D. (2022). Electron spin coherence on a solid neon surface.

[Chuang et al., 1998] Chuang, I. L., Gershenfeld, N., and Kubinec, M. (1998). Experimental implementation of fast quantum searching. *Phys. Rev. Lett.*, 80:3408–3411.

[Cole and Cohen, 1969] Cole, M. W. and Cohen, M. H. (1969). Image-potential-induced surface bands in insulators. *Phys. Rev. Lett.*, 23:1238–1241.

[Conway, 2015] Conway, J. T. (2015). A lagrangian method for deriving new indefinite integrals of special functions.

[Conway, 2021] Conway, J. T. (2021). Indefinite integrals for some orthogonal polynomials obtained using integrating factors. *Integral Transforms and Special Functions*, 32(1):1–13.

[Dalgarno and Lewis, 1955] Dalgarno, A. and Lewis, J. T. (1955). The Exact Calculation of Long-Range Forces between Atoms by Perturbation Theory. *Proceedings of the Royal Society of London Series A*, 233(1192):70–74.

[Davies, 2002] Davies, B. (2002). *Integral Transforms and Their Applications*. Texts in Applied Mathematics. Springer New York.

[de Wolf, 2023] de Wolf, R. (2023). Quantum computing: Lecture notes.

[Deutsch, 1985] Deutsch, D. (1985). Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London Series A*, 400(1818):97–117.

[Deutsch and Jozsa, 1992] Deutsch, D. and Jozsa, R. (1992). Rapid Solution of Problems by Quantum Computation. *Proceedings of the Royal Society of London Series A*, 439(1907):553–558.

[Dykman and Platzman, 2000] Dykman, M. and Platzman, P. (2000). Quantum computing using electrons floating on liquid helium. *Fortschritte der Physik*, 48(9–11):1095–1108.

[Dykman et al., 2003] Dykman, M. I., Platzman, P. M., and Seddighrad, P. (2003). Qubits with electrons on liquid helium. *Physical Review B*, 67(15).

[Feynman, 1982] Feynman, R. P. (1982). Simulating physics with computers. 21(6):467–488.

[Flügge, 1999] Flügge, S. (1999). *Practical Quantum Mechanics*. Springer.

[Gill et al., 2025] Gill, S. S., Cetinkaya, O., Marrone, S., Claudino, D., Haunschild, D., Schlote, L., Wu, H., Ottaviani, C., Liu, X., Machupalli, S. P., Kaur, K., Arora, P., Liu, J., Farouk, A., Song, H. H., Uhlig, S., and Ramamohanarao, K. (2025). Quantum computing: Vision and challenges.

[Gordon, 1929] Gordon, W. (1929). Zur berechnung der matrizen beim wasserstoffatom. *Annalen der Physik*, 394:1031 – 1056.

[Grimes et al., 1976] Grimes, C. C., Brown, T. R., Burns, M. L., and Zipfel, C. L. (1976). Spectroscopy of electrons in image-potential-induced surface states outside liquid helium. *Phys. Rev. B*, 13:140–147.

[Grotch, 1981] Grotch, H. (1981). Comment on radiative corrections to the energy levels of "murium": An electron bound by its image charge to a wall. *Phys. Rev. A*, 24:1120–1122.

[Harvey, 2022] Harvey, S. P. (2022). Quantum dots/spin qubits.

[Herbst and Simon, 1978] Herbst, I. W. and Simon, B. (1978). Stark effect revisited. *Phys. Rev. Lett.*, 41:67–69.

[Huang et al., 2020] Huang, H.-L., Wu, D., Fan, D., and Zhu, X. (2020). Superconducting quantum computing: a review. *Science China Information Sciences*, 63(8).

[IBM Quantum, 2023] IBM Quantum (2023). Ibm quantum roadmap: 2025 and beyond. `https://www.ibm.com/quantum/blog/ibm-quantum-roadmap-2025`. Accessed: 2025-05-18.

[Jennings et al., 2024] Jennings, A., Zhou, X., Grytsenko, I., and Kawakami, E. (2024). Quantum computing using floating electrons on cryogenic substrates: Potential and challenges. *Applied Physics Letters*, 124(12):120501.

[Ji et al., 2024] Ji, J.-B., Ueda, K., Han, M., and Wörner, H. J. (2024). Analytical expression for continuum-continuum transition amplitude of hydrogen-like atoms with angular-momentum dependence.

[Kawakami et al., 2023] Kawakami, E., Chen, J., Benito, M., and Konstantinov, D. (2023). Blueprint for quantum computing using electrons on helium. *Physical Review Applied*, 20(5).

[Kibble and Berkshire, 2004] Kibble, T. and Berkshire, F. (2004). *Classical Mechanics (5th Edition)*. World Scientific Publishing Company.

[Lambert and Richards, 1981] Lambert, D. K. and Richards, P. L. (1981). Far-infrared and capacitance measurements of electrons on liquid helium. *Phys. Rev. B*, 23:3282–3290.

[Landau and Lifshitz, 1965] Landau, L. D. and Lifshitz, E. M. (1965). *Quantum Mechanics: Non-relativistic Theory*. Pergamon Press.

[Landau and Lifshitz, 1984] Landau, L. D. and Lifshitz, E. M. (1984). *Electrodynamics of Continuous Media*. Pergamon, New York.

[Lide, 2005] Lide, D. R., editor (2005). *CRC Handbook of Chemistry and Physics, Internet Version 2005*. CRC Press, Boca Raton, FL. `http://www.hbcpnetbase.com`.

[Lloyd, 1996] Lloyd, S. (1996). Universal quantum simulators. *Science*, 273(5278):1073–1078.

[Lyon, 2006] Lyon, S. A. (2006). Spin-based quantum computing using electrons on liquid helium.

[Mera et al., 2015] Mera, H., Pedersen, T. G., and Nikolić, B. K. (2015). Nonperturbative quantum physics from low-order perturbation theory. *Phys. Rev. Lett.*, 115:143001.

[Mera et al., 2018] Mera, H., Pedersen, T. G., and Nikolić, B. K. (2018). Fast summation of divergent series and resurgent transseries from meijer- <mml:math xmlns:mml="http://www.w3.org/1998/math/mathml" display="inline"><mml:mi>g</mml:mi></mml:math> approximants. *Physical Review D*, 97(10).

[Nieto, 2000] Nieto, M. M. (2000). Electrons above a helium surface and the one-dimensional rydberg atom. *Phys. Rev. A*, 61:034901.

[Pedersen, 2022] Pedersen, T. G. (2022). *Electrical, optical and magnetic properties of nanostructures*. N/A.

[Pedersen, 2023] Pedersen, T. G. (2023). Coulomb-zeeman-stark problem in two dimensions. *Phys. Rev. A*, 107:022804.

[Pedersen, 2024] Pedersen, T. G. (2024). Unpublished. *N/A*.

[Pedersen et al., 2016] Pedersen, T. G., Mera, H., and Nikolić, B. K. (2016). Stark effect in low-dimensional hydrogen. *Phys. Rev. A*, 93:013409.

[Pedrotti et al., 2018] Pedrotti, F. L., Pedrotti, L. M., and Pedrotti, L. S. (2018). *Introduction to optics*. Cambridge University Press, Cambridge, third edition. / frank l. pedrotti, leno m. pedrotti, leno s. pedrotti. edition.

[Romero and Milburn, 2024] Romero, J. and Milburn, G. (2024). Photonic quantum computing.

[Saad, 2014] Saad, N. (2014). On w. gordon's integral (1929) and related identities.

[Sakurai and Napolitano, 2017] Sakurai, J. and Napolitano, J. (2017). *Modern Quantum Mechanics*. Cambridge University Press.

[Seaton, 1958] Seaton, M. J. (1958). The quantum defect method. *Monthly Notices of the Royal Astronomical Society*, 118(5):504–518.

[Shakeshaft and Spruch, 1980] Shakeshaft, R. and Spruch, L. (1980). Radiative corrections to the energy levels of "murium," an electron bound by its image charge to a wall. *Phys. Rev. A*, 22:811–817.

[Shankar, 1994] Shankar, R. (1994). *Principles of Quantum Mechanics*. Springer, 2nd edition.

[Shor, 1997] Shor, P. W. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509.

[Simon, 1997] Simon, D. (1997). On the power of quantum computation. *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science*, 26.

[Susskind and Jensen, 1988] Susskind, S. M. and Jensen, R. V. (1988). Numerical calculations of the ionization of one-dimensional hydrogen atoms using hydrogenic and sturmian basis functions. *Phys. Rev. A*, 38:711–728.

[Vandersypen et al., 2001] Vandersypen, L. M. K., Steffen, M., Breyta, G., Yannoni, C. S., Sherwood, M. H., and Chuang, I. L. (2001). Experimental realization of shor's quantum factoring algorithm using nuclear magnetic resonance. *Nature*, 414(6866):883–887.

[Zhahir et al., 2023] Zhahir, A., Mohd, S., M Shuhud, M. I., Idrus, B., Zainuddin, H., Mohamad Jan, N., and Wahiddin, M. R. (2023). Quantum computing and its application. *International Journal of Advanced Research in Technology and Innovation*.

[Zhou et al., 2022] Zhou, X., Koolstra, G., Zhang, X., Yang, G., Han, X., Dizdar, B., Li, X., Divan, R., Guo, W., Murch, K. W., Schuster, D. I., and Jin, D. (2022). Single electrons on solid neon as a solid-state qubit platform. *Nature*, 605(7908):46–50.

Appendix A

Hypergeometric Functions & Identities

An important class of functions that see numerous applications in quantum mechanics are the hypergeometric functions. The general case of the hypergeometric function, which we denote ${}_pF_q$, is defined in terms of an analytical continuation of the power series

$$
\begin{aligned}
{}_pF_q(a_1, a_2, \dots, a_p, b_1, b_2, \dots, b_q, x) &= \sum_{k=0}^{\infty} \frac{\prod_{j=1}^{p}(a_j)_k}{\prod_{l=1}^{q}(b_l)_k} \frac{x^k}{k!} \\
&= \sum_{k=0}^{\infty} \frac{(a_1)_k(a_2)_k \dots (a_p)_k}{(b_1)_k(b_2)_k \dots (b_q)_k} \frac{x^k}{k!},
\end{aligned} \tag{A.0.1}
$$

where $p, q \in \mathbb{Z}_{>0}$ and $(x)_n = \Gamma(x+n)/\Gamma(x)$ is the Pochhammer symbol defined in terms of the gamma function $\Gamma(x)$. Two particularly important cases for the purposes of this project are the Kummer confluent hypergeometric function ${}_1F_1(a, b, x)$ and the Gauss hypergeometric function ${}_2F_1(a, b, c, x)$. An important property of the Pochhammer symbol $(-m)_n$ is that it terminates for $0 < m < n$ and $m \in \mathbb{Z}_{>0}$. This truncates the hypergeometric functions into polynomials of finite order. An important class of polynomials that the hypergeometric functions can generate, are the associated Laguerre polynomials $L_b^{(a)}(x)$, which are solutions to the differential equation given by

$$
x\frac{\mathrm{d}^2y}{\mathrm{d}x^2} + (a+1-x)\frac{\mathrm{d}y}{\mathrm{d}x} + by = 0, \tag{A.0.2}
$$

where $a, b \in \mathbb{C}$ are two constants. The function $y = L_b^{(a)}(x)$ that solves equation (A.0.2) above may be expressed as

$$
L_b^{(a)}(x) = \sum_{k=0}^{b} \binom{a+b}{b-k} \frac{(-x)^k}{k!}, \tag{A.0.3}
$$

where $\binom{a}{b} = \Gamma(a+1)/\Gamma(a-b+1)\Gamma(b+1)$ is the generalized binomial coefficient expressed in terms of gamma functions. The associated Laguerre polynomials form an orthogonal set using the weight $x^a \mathrm{e}^{-x}$, meaning

$$
\int_0^{\infty} x^a \mathrm{e}^{-x} L_m^{(a)}(x) L_n^{(a)}(x) \mathrm{d}x = (n+1)_a \delta_{mn}, \tag{A.0.4}
$$

where δ_{mn} is the Kronecker delta. The Laguerre polynomials satisfy a number of functional identities that allow us to shift the parameters a, b. One such identity reads

$$
xL_n^{(a)}(x) = (2n+a+1)L_n^{(a)}(x) - (n+1)L_{n+1}^{(a)}(x) - (n+a)L_{n-1}^{(a)}(x). \tag{A.0.5}
$$

The Laguerre polynomials can up to a constant be represented as a special case of the Kummer confluent hypergeometric function ${}_1F_1(-b, 1+a, x)$. In particular, we find that

$$
L_b^{(a)}(x) = \frac{(1+a)_b}{\Gamma(1+b)} {}_1F_1(-b; 1+a, x). \tag{A.0.6}
$$

The associated Laguerre polynomials and/or the Kummer confluent hypergeometric functions frequently appear in calculations involving the radial part of the eigenstates within quantum mechanics. Examples include calculating dipole matrix elements and normalization constants for eigenstates, which will be done numerous times during this project. For convenience therefore, the remaining part of this section will described various differentiation and integral identities involving these functions. The first of these identities is the derivative of the hypergeometric function, which can be shown to be

$$\frac{\mathrm{d}}{\mathrm{d}x}{}_pF_q(a_1,\dots,a_p,b_1,\dots,b_q,x) = \frac{a_1\dots a_p}{b_1\dots b_q}{}_pF_q(a_1+1,\dots,a_p+1,b_1+1,\dots,b_q+1,x). \tag{A.0.7}$$

A similar formula for integration can be found from the above equation. We now move onto a number of integral identities, which will be used extensively. All the integrals of this type relevant to this project fall under the so-called W. Gordon integral, which we may express as

$$\mathrm{J}_\varrho^{\sigma,p}(\alpha_1,\alpha_2;\gamma,\beta_1,\beta_2) = \int_0^\infty x^{\varrho+\sigma-1}\mathrm{e}^{-\gamma x}{}_1F_1(\alpha_1;\varrho;\beta_1 x){}_1F_1(\alpha_2;\varrho-p;\beta_2 x)\mathrm{d}x. \tag{A.0.8}$$

This integral was originally studied for similar applications in 1929 by W. Gordon, see [Gordon, 1929]. A summary of the results needed for this thesis are described in the appendix of [Landau and Lifshitz, 1965], which we include here. For the purposes of thesis, we only consider a few cases of the general integral. The first of these is

$$\begin{aligned}\mathrm{J}_\varrho^{\sigma,p}\left(\alpha_1,\alpha_2;\frac{\beta_1+\beta_2}{2},\beta_1,\beta_2\right) &= \int_0^\infty x^{\varrho+\sigma-1}\mathrm{e}^{-\frac{\beta_1+\beta_2}{2}x}{}_1F_1(\alpha_1;\varrho;\beta_1 x){}_1F_1(\alpha_2;\varrho-p;\beta_2 x)\mathrm{d}x\\ &\equiv \mathrm{J}_\varrho^{\sigma,p}\left(\alpha_1,\alpha_2,\beta_1,\beta_2\right).\end{aligned} \tag{A.0.9}$$

A general closed-form expression for the integral given in equation (A.0.9) above does exist, but is typically too complex to be used conveniently. Instead, we use a set of recursive relations that reduce the general integral into a number of simpler integrals of the $\sigma = p = 0$ case, which may be evaluated using

$$\begin{aligned}\mathrm{J}_\varrho^{0,0}(\alpha_1,\alpha_2,\beta_1,\beta_2) &= \int_0^\infty x^{\varrho-1}\mathrm{e}^{-\frac{\beta_1+\beta_2}{2}x}{}_1F_1(\alpha_1;\varrho;\beta_1 x){}_1F_1(\alpha_2;\varrho;\beta_2 x)\mathrm{d}x\\ &= \frac{2^\varrho\Gamma(\varrho)(\beta_1+\beta_2)^{\alpha_1+\alpha_2-\varrho}}{(\beta_2-\beta_1)^{\alpha_1}(\beta_1-\beta_2)^{\alpha_2}}{}_2F_1\left(\alpha_1,\alpha_2,\varrho,\frac{-4\beta_1\beta_2}{(\beta_2-\beta_1)^2}\right).\end{aligned} \tag{A.0.10}$$

The above identity can be derived by substituting the contour integral representation of the second ${}_1F_1$ series, interchanging order of integration and applying an appropriate substitution. To reduce the general integral given in equation (A.0.8), we first lower the p-index using

$$\mathrm{J}_\varrho^{\sigma,p}(\alpha_1,\alpha_2,\beta_1,\beta_2) = \frac{\varrho-1}{\beta_1}\left(\mathrm{J}_{\varrho-1}^{\sigma,p-1}(\alpha_1,\alpha_2,\beta_1,\beta_2) - \mathrm{J}_{\varrho-1}^{\sigma,p-1}(\alpha_1-1,\alpha_2,\beta_1,\beta_2)\right). \tag{A.0.11}$$

Once $p = 0$, we can then reduce the σ-index using

$$\begin{aligned}\mathrm{J}_\varrho^{\sigma+1,0}(\alpha_1,\alpha_2,\beta_1,\beta_2) = \ &\frac{4}{\beta_1^2-\beta_2^2}\Bigg\{\left(\frac{1}{2}\varrho(\beta_1-\beta_2)-\beta_1\alpha_1+\beta_2(\alpha_2-\sigma)\right)\mathrm{J}_\varrho^{\sigma,0}(\alpha_1,\alpha_2,\beta_1,\beta_2)\\ &+\ \sigma\left(\varrho-1+\sigma-2\alpha_2\right)\mathrm{J}_\varrho^{\sigma-1,0}(\alpha_1,\alpha_2,\beta_1,\beta_2)+2\alpha_2\sigma\mathrm{J}_\varrho^{\sigma-1,0}(\alpha_1,\alpha_2+1,\beta_1,\beta_2)\Bigg\}.\end{aligned} \tag{A.0.12}$$

These recurrence relations can be derived by considering the governing differential equation of the radial eigenfunctions of the Hydrogen atom and manipulating it using various functional relations of the ${}_1F_1$ function. Two particularly important cases of equation (A.0.12) are the $\sigma = 0$ case given by

$$\mathrm{J}_\varrho^{1,0}(\alpha_1,\alpha_2,\beta_1,\beta_2) = \frac{4\mathcal{A}}{\beta_1^2-\beta_2^2}\mathrm{J}_\varrho^{0,0}(\alpha_1,\alpha_2,\beta_1,\beta_2) \tag{A.0.13}$$

and the $\sigma = 1$ case given by

$$\begin{aligned} \mathrm{J}_{\varrho}^{2,0}(\alpha_1, \alpha_2, \beta_1, \beta_2) = \; & \frac{4}{\beta_1^2 - \beta_2^2} \Big\{ (\mathcal{A} - \beta_2)\, \mathrm{J}_{\varrho}^{1,0}(\alpha_1, \alpha_2, \beta_1, \beta_2) + (\varrho - 2\alpha_2)\, \mathrm{J}_{\varrho}^{0,0}(\alpha_1, \alpha_2, \beta_1, \beta_2) \\ & + \; 2\alpha_2 \mathrm{J}_{\varrho}^{0,0}(\alpha_1, \alpha_2 + 1, \beta_1, \beta_2) \Big\}, \end{aligned} \tag{A.0.14}$$

where we for convenience have defined the parameter $\mathcal{A} = \frac{1}{2}\varrho(\beta_1 - \beta_2) - \beta_1\alpha_1 + \beta_2\alpha_2$. There exists a number of useful special cases of the general integral given in equation (A.0.8). One particular example that will be used in this thesis is reported in [Saad, 2014], where $p = 0$, $\alpha_1 = \alpha_2 = -n$ for $n \in \mathbb{Z}_{>0}$ and $\beta_1 = \beta_2 = \gamma$ such that

$$\begin{aligned} \mathrm{J}_{\varrho}^{\sigma,0}\,(-n, -n; \beta, \beta, \beta) &= \int_0^\infty x^{\varrho+\sigma-1} \mathrm{e}^{-\beta x}\, [{}_1F_1(-n; \varrho; \beta x)]^2\, \mathrm{d}x \\ &= \frac{\Gamma(\varrho + \sigma) n!}{\beta^{\varrho+\sigma} (\varrho)_n} {}_3F_2(-n, -\sigma, 1 + \sigma, \varrho, 1, 1). \end{aligned} \tag{A.0.15}$$

Despite being a ${}_3F_2$ series, we can evaluate it in closed form because it truncates like the ${}_2F_1$ series above. In addition, because it has unit argument, it typically simplifies greatly. Another special case, which is provided in [Landau and Lifshitz, 1965], is the $\alpha_2 = 0$ case and reads

$$\begin{aligned} \mathrm{J}_{\varrho}^{\sigma,p}(\alpha_1, 0; \gamma, \beta_1, \beta_2) &= \int_0^\infty x^{\varrho+\sigma-1} \mathrm{e}^{-\gamma x} {}_1F_1(\alpha_1; \varrho; \beta_1 x) \mathrm{d}x \\ &= \frac{\Gamma(\varrho + \sigma)}{\gamma^{\varrho+\sigma}} {}_2F_1 \left(\alpha_1, \varrho + \sigma, \varrho, \frac{\beta_1}{\gamma} \right). \end{aligned} \tag{A.0.16}$$

Typical applications of equation (A.0.16) used in this project will allow substitutions that reduce the ${}_2F_1$ function significantly similarly to the ${}_3F_2$ function in equation (A.0.15).

Appendix B

Coulomb Wave Functions & Identities

The Coulomb wave functions are a pair of functions that arise when studying the unbound or continuum eigenstates of a particle subject to a Coulomb-like potential. Under the appropriate substitutions, the corresponding Schrödinger equation can be written as

$$\frac{\mathrm{d}^2 y}{\mathrm{d}x^2} + \left\{1 - \frac{2\eta}{x} - \frac{\ell\,(\ell+1)}{x^2}\right\} y = 0, \tag{B.0.1}$$

where $\ell, \eta \in \mathbb{R}$ are real parameters. The parameter ℓ corresponds to the angular moment quantum number and $\eta = 1/a_0 k$, where a_0 and k are the Bohr radius and wavenumber, respectively. In this context, the independent variable is $x = kr$, where r is the radial coordinate. The Schrödinger equation written above is a second-order linear differential equation to which there exists two linearly independent solutions. The first of these solutions is denoted $y = F_{\ell,\eta}$ and is regular at the origin. Specifically, it goes as $F_{\ell,\eta} \sim x^{\ell+1}$ for x near zero. The second solution is denoted $y = G_{\ell,\eta}$ and is irregular at the origin, meaning $G_{\ell,\eta} \sim x^{-\ell}$ for x near zero. For the purposes of this thesis, we are only interested in the regular solution $F_{\ell,\eta}$, which may be expressed as

$$F_{\ell,\eta} = N_{\ell,\eta} x^{\ell+1} \mathrm{e}^{\pm ix} {}_1F_1\left(1 + \ell \pm i\eta, 2 + 2\ell, \mp 2ix\right), \tag{B.0.2}$$

where $N_{\ell,\eta}$ is a normalization parameter. Bound eigenstates are square-integrable functions on $\mathbb{R}_+$ and hence can be normalized according to the condition $\langle \psi_n | \psi_m \rangle = \delta_{nm}$, where δ_{nm} is the Kronecker delta. It is clear from equation (B.0.2) that $N_{\ell,\eta}$ cannot be chosen such that $F_{\ell,\eta}$ satisfies this condition. Instead, the unbound eigenstates are normalized on the "k-scale" or δ-normalized according to $\langle \psi_{k'} | \psi_k \rangle = \delta(k' - k)$, where $\delta(k' - k)$ is the Dirac delta function. In this section, we outline a procedure presented in [Susskind and Jensen, 1988] that allows the determination of the normalization parameter for the regular Coulomb potential given in equation (2.1.27), where unbound states satisfy

$$\left(-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}z^2} - \frac{1}{z}\right)\psi_k = \frac{1}{2}k^2\psi_k, \quad \text{for } \psi_k = \mathcal{N}_k z \mathrm{e}^{-ikz} {}_1F_1\left(1 + \frac{i}{k}, 2, 2ikz\right), \tag{B.0.3}$$

which corresponds to $\ell = 0$ and $\eta = -1/k$. In practice, the normalization condition may be enforced by scaling the asymptotic behaviour of the eigenstates appropriately for large x. To do so, we start by taking two copies of the differential equation given in equation (B.0.3). The first copy is for wavenumber k and the second is for wavenumber k'. We now multiply the first copy through by $\psi^\star_{k'}$. In addition, we conjugate the second copy and multiply through by ψ_k so that

$$-\frac{1}{2}\frac{\mathrm{d}^2\psi_k}{\mathrm{d}z^2}\psi^\star_{k'} - \frac{1}{z}\psi_k\psi^\star_{k'} = \frac{1}{2}k^2\psi_k\psi^\star_{k'}, \tag{B.0.4}$$

$$-\frac{1}{2}\psi_k\frac{\mathrm{d}^2\psi^\star_{k'}}{\mathrm{d}z^2} - \frac{1}{z}\psi_k\psi^\star_{k'} = \frac{1}{2}k'^2\psi_k\psi^\star_{k'}. \tag{B.0.5}$$

We now subtract equation (B.0.5) from equation (B.0.4). The resulting expression is then integrated over $[0, L]$ for some real $L > 0$, which yields

$$(k'^2 - k^2)\int_0^L \psi_k\psi^\star_{k'}\mathrm{d}z = \int_0^L \psi_k\frac{\mathrm{d}^2\psi^\star_{k'}}{\mathrm{d}z^2}\mathrm{d}z - \int_0^L \frac{\mathrm{d}^2\psi_k}{\mathrm{d}z^2}\psi^\star_{k'}\mathrm{d}z. \tag{B.0.6}$$

We now apply integration by parts to both integrals on the right-hand side in such a way that the second order derivatives are reduced to first order. Reducing then shows

$$
\begin{aligned}
(k'^2 - k^2) \int_0^L \psi_k \psi_{k'}^\star \mathrm{d}z &= \left[\psi_k \frac{\mathrm{d}\psi_{k'}^\star}{\mathrm{d}z} - \psi_{k'}^\star \frac{\mathrm{d}\psi_k}{\mathrm{d}z}\right]_0^L + \int_0^L \left(\frac{\mathrm{d}\psi_{k'}^\star}{\mathrm{d}z}\frac{\mathrm{d}\psi_k}{\mathrm{d}z} - \frac{\mathrm{d}\psi_k}{\mathrm{d}z}\frac{\mathrm{d}\psi_{k'}^\star}{\mathrm{d}z}\right)\mathrm{d}z \\
&= \left[\psi_k \frac{\mathrm{d}\psi_{k'}^\star}{\mathrm{d}z} - \psi_{k'}^\star \frac{\mathrm{d}\psi_k}{\mathrm{d}z}\right]_{z=L},
\end{aligned}
\tag{B.0.7}
$$

where the lower bound on the first term on the right-hand side goes to zero because of the Dirichlet boundary conditions. We eventually want to let $L \longrightarrow \infty$ so that the integral on the left-hand side becomes the inner product $\langle \psi_{k'} | \psi_k \rangle$. The right-hand side will then only depend on the behaviour of ψ_k for large z. The trick now is to substitute the asymptotic representation of ψ_k for large z into the right-hand side and then let $L \longrightarrow \infty$. The asymptotic representation of the Coulomb wave functions are reported in the literature, see for instance [Abramowitz and Stegun, 1964]. For this particular eigenstate, we find

$$
\psi_k \sim \frac{-\sqrt{2\pi}\mathcal{N}_k \mathrm{e}^{-\pi/2k}}{k\left|\Gamma(1 - ik^{-1})\right|}\sin\theta_k, \quad \text{for } \theta_k = kz + k^{-1}\ln(2kz) + \arg\Gamma(1 - ik^{-1}).
\tag{B.0.8}
$$

We now briefly provide an outline of how this representation is derived, which follows an approach given in an appendix in [Susskind and Jensen, 1988]. We start by returning to equation (B.0.3) and solving the problem again in the momentum-space representation. Accordingly, the eigenstates $\psi_k(p)$ simply are the Fourier transform of the position space eigenstates $\psi_k(x)$, meaning $\psi_k(p) = \mathcal{F}\left\{\psi_k(x)\right\}(p)$. They can be shown to be

$$
\begin{aligned}
\psi_k(p) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \psi_k(z)\mathrm{e}^{-ipz}\mathrm{d}z \\
&= \frac{\mathcal{N}_k}{p^2 - k^2}\left(\frac{p + k}{p - k}\right)^{i/k}.
\end{aligned}
\tag{B.0.9}
$$

The trick now is to apply Watson's lemma, see [Davies, 2002], which allows us to extract an asymptotic representation of $\psi_k(x)$ from its Laplace transform $\psi_k(s) = \mathcal{L}\left\{\psi_k(x)\right\}(s)$. The Laplace transform is obtained directly from Fourier transform by using the relation $\mathcal{L}\left\{\psi_k(x)\right\}(s) = \sqrt{2\pi}\mathcal{F}\left\{\psi_k(x)\right\}(-is)$, which applies in this case because of the Dirichlet boundary conditions. Hence

$$
\begin{aligned}
\psi_k(s) &= \sqrt{2\pi}\mathcal{F}\left\{\psi_k(x)\right\}(-is) \\
&= \frac{-\sqrt{2\pi}\mathcal{N}_k}{s^2 + k^2}\left(\frac{s + ik}{s - ik}\right)^{i/k}.
\end{aligned}
\tag{B.0.10}
$$

We now expand $\psi_k(s)$ near its branch points $s = \pm ik$ and apply Watson's Lemma, which can be shown to eventually yield equation (B.0.8). We now substitute the asymptotic representation back into equation (B.0.7) and taking $L \longrightarrow \infty$, which yields

$$
\int_0^\infty \psi_k \psi_{k'}^\star \mathrm{d}z = \frac{(k'^2 - k^2)^{-1}\mathcal{N}_{k'}^\star \mathcal{N}_k \mathrm{e}^{-\frac{\pi(k+k')}{2kk'}}}{kk'\left|\Gamma(1 - ik^{-1})\right|\left|\Gamma(1 - ik'^{-1})\right|} \lim_{z\to\infty}\left[\theta_{k'}'^\star \sin\theta_k \cos\theta_{k'}^\star - \theta_k' \sin\theta_{k'}^\star \cos\theta_k\right].
\tag{B.0.11}
$$

We now reduce the argument of the limit on the right-hand side. This eventually brings it to a form that allows us to apply the identity

$$
\lim_{x\to\infty} \frac{\sin\left(f(x)\Delta k\right)}{\Delta k} = \pi\delta(\Delta k),
\tag{B.0.12}
$$

where $\Delta k = k' - k$ and f is an arbitrary function that satisfies $\lim_{x\to\infty} f(x) = \infty$. Doing so then yields

$$
\int_0^\infty \psi_k \psi_{k'}^\star \mathrm{d}z = \frac{\pi\mathcal{N}_{k'}^\star \mathcal{N}_k \mathrm{e}^{-\pi/2k'}\mathrm{e}^{-\pi/2k}}{2kk'\left|\Gamma(1 - ik^{-1})\right|\left|\Gamma(1 - ik'^{-1})\right|}\delta(k' - k).
\tag{B.0.13}
$$

It is then clear that $\langle \psi_{k'} | \psi_k \rangle = \delta(k' - k)$ is satisfied if we choose $\mathcal{N}_k$ so that the fraction on the right-hand side of equation (B.0.13) equals unity. This in turn shows that

$$\mathcal{N}_k = \sqrt{\frac{2}{\pi}} k \mathrm{e}^{\pi/2k} \left| \Gamma(1 - ik^{-1}) \right|. \tag{B.0.14}$$

We can simplify the expression by exploiting $|z| = \sqrt{zz^\star}$ and $\Gamma(1+z) = z\Gamma(z)$, which in turn yields

$$\begin{aligned} \mathcal{N}_k &= \sqrt{\frac{2}{\pi}} k \mathrm{e}^{\pi/2k} \left| \Gamma(1 - ik^{-1}) \right| \\ &= \sqrt{\frac{2}{\pi}} k \mathrm{e}^{\pi/2k} \sqrt{\Gamma(1 - ik^{-1})\Gamma(1 + ik^{-1})} \\ &= \sqrt{\frac{2}{\pi}} k \mathrm{e}^{\pi/2k} \sqrt{\Gamma(1 - ik^{-1}) ik^{-1} \Gamma(ik^{-1})} \\ &= \sqrt{\frac{2}{\pi}} k \mathrm{e}^{\pi/2k} \sqrt{\frac{i\pi}{k \sin(i\pi/k)}}, \end{aligned} \tag{B.0.15}$$

where we in the last line have used Euler's reflection formula $\Gamma(z)\Gamma(1-z) = \pi/\sin(\pi z)$. We may reduce further by noting that $\sin(iz) = i \sinh z$ so that

$$\begin{aligned} \mathcal{N}_k &= \sqrt{\frac{2}{\pi}} k \mathrm{e}^{\pi/2k} \sqrt{\frac{\pi}{k \sinh(\pi/k)}} \\ &= \frac{2k^{1/2}}{\sqrt{1 - \mathrm{e}^{-2\pi/k}}}, \end{aligned} \tag{B.0.16}$$

which gives the final normalization constant. A similar calculation for the Kratzer potential given in equation (2.2.14) shows that

$$\mathcal{N}_k = \frac{(2k)^{1-\delta} \left| \Gamma(1 - \delta - ik^{-1}) \right|}{\sqrt{2\pi} \mathrm{e}^{-\pi/2k} \Gamma(2 - 2\delta)}, \tag{B.0.17}$$

which reduces to equation (B.0.16) in the $\delta = 0$ case as it should.

Appendix C

Electrostatic Correction Code

The following code generates the electrostatic eigenstate corrections under the electrostatic perturbation using the Dalgarno-Lewis approach. It is valid for the Coulomb corrections. The code is written for Python.

Listing C.1: Perturbed Eigenfunctions Calculator 1

```python
import numpy as np
import matplotlib.pyplot as plt
import sympy as smp
import mpmath as mp
from sympy.abc import z

'LaTeX'
plt.rcParams['text.usetex'] = True # Enable LaTeX rendering
plt.rcParams['font.family'] = 'serif' # Use a serif font
plt.rcParams['font.serif'] = ['Computer Modern'] # LaTeX's default serif font
plt.rcParams['text.latex.preamble'] = r'\usepackage{amsmath}' # Optional: for more LaTeX symbols

'Parameters'
p = 6 #Perturbation Order
n = 1 #Quantum Number
delta = 0 #Quantum Defect Parameter
scriptn = n - delta #Effective Quantum Number

'Symbols'
z = smp.symbols('z')
E = smp.symbols('E_0', real=True)

'Functions'
def Gamma(z):
    return mp.gamma(z)
def Poch(a, b):
    return Gamma(a+b)/Gamma(a)
def LaguerreL(n, alpha, z):
    return smp.assoc_laguerre(n, alpha, z)
def Hyp1F1(a, b, z):
    return smp.hyper([a], [b], z)
def Hyp2F1(a, b, c, z):
    return smp.hyper([a, b], [b], z)
def SInf(H, f):
    z = smp.symbols('z')
    integrand = smp.nsimplify(H * f)
    S = smp.integrate(integrand, (z, 0, smp.oo), conds='none')
    return smp.refine(S)
def SInd(H, f):
    z = smp.symbols('z')
    integrand = smp.nsimplify(H * f)
    S = smp.integrate(integrand, z, conds='none')
    return smp.refine(S)

'Unperturbed Eigenstates'
def psi(n, delta, z):
    scriptn = n - delta
    Factor1 = 2**(1-delta) / (np.sqrt(Poch(n, 1-2*delta)) * scriptn**(2-delta))
    psi = Factor1 * z**(1-delta) * smp.exp(-z/scriptn) * LaguerreL(n-1, 1 - 2*delta, 2*z/scriptn)
    return psi

'Storage'
EnergyStorage = [None]*(p+1)
FieldFactorStorage = [None]*(p+1)
C2Storage = [None]*(p+1)
EigenstateStorage = [None]*(p+1)

'Perturbative Operator'
def H(p):
    H = E*z * FieldFactorStorage[p-1] - sum(EnergyStorage[j]*FieldFactorStorage[p-j] for j in range(1, p+1))
    return H

'Zeroth-Order Calculation'
#Zeroth-Order Energy
E0 = -1/(2*(n-delta)**2)
EnergyStorage[0] = E0
#Zeroth-Order Integration Constant
C0 = 0
C2Storage[0] = C0
#Zeroth-Order Field Factor
f0 = 1
FieldFactorStorage[0] = f0
#Zeroth-Order Eigenstate Correction
psi0 = psi(n, delta, z)
EigenstateStorage[0] = psi0

'First-Order Calculation'
#First-Order Energy
E1 = smp.nsimplify(SInf(E*z, psi0**2))
EnergyStorage[1] = E1
#First-Order Integration Constant
C1 = smp.nsimplify(SInd(-H(1), psi0**2))
C1 = smp.nsimplify(SInd(C1, psi0**(-2)))
C1 = smp.nsimplify(SInf(C1, psi0**(2)))
C2Storage[1] = C1
#First-Order Field Factor
f1 = smp.nsimplify(SInd(H(1), psi0**2))
f1 = smp.nsimplify(SInd(f1, psi0**(-2)))
f1 = smp.nsimplify(2*f1 + 2*C1)
FieldFactorStorage[1] = f1
#First-Order Eigenstate Correction
EigenstateStorage[1] = FieldFactorStorage[1]*psi0

'Higher-Order Calculations'
if p>1:
    for i in range(2, p+1):
        'Energy Correction'
        Ei = smp.nsimplify(SInf(psi0*E*z, EigenstateStorage[i-1]))
        Ei = Ei - sum(EnergyStorage[j]*SInf(psi0, EigenstateStorage[i-j]) for j in range(1, i))
        EnergyStorage[i] = smp.nsimplify(Ei)
        'Integration Constant'
        Ci = smp.nsimplify(SInd(-H(i), psi0**2))
        Ci = smp.nsimplify(SInd(Ci, psi0**(-2)))
        Ci = smp.nsimplify(SInf(Ci, psi0**(2)))
        Ci = Ci - 1/4 * sum(SInf(EigenstateStorage[j], EigenstateStorage[i-j]) for j in range(1, i))
        C2Storage[i] = smp.nsimplify(Ci)
        'Field Factor'
        fi = smp.nsimplify(SInd(H(i), psi0**(2)))
        fi = smp.nsimplify(SInd(fi, psi0**(-2)))
        FieldFactorStorage[i] = smp.nsimplify(2*fi + 2*C2Storage[i])
        'Eigenstate Correction'
        EigenstateStorage[i] = smp.nsimplify(FieldFactorStorage[i] * psi0)
```

The following code generates the electrostatic eigenstate corrections under the electrostatic perturbation by directly constructing them. It provides the first and second order eigenstate corrections for arbitrary n

and can also calculate up to the fourth order energy correction. The code is written for Mathmatica.

Listing C.2: Perturbed Eigenfunctions Calculator 2

```
ClearAll["Global‘*"];
BinomialCoeff[a_, b_] := If[b < 0, 0, Binomial[a, b]];

$Assumptions = E0 > 0;
n = 2;
(*delta = 0;*)

Mn[n_, \[Delta]_] := 2^(1 - \[Delta])*(n - \[Delta])^(\[Delta] - 2) / Sqrt[Pochhammer[n, 1 - 2*\[Delta]]];
Znn[n_, \[Delta]_] := 3/2*n^(2) - \[Delta]*(3*n - 1/2 - \[Delta]);
psin [n_, \[Delta]_] := Mn[n, \[Delta]]*z^(1 - \[Delta])*Exp[-z/(n - \[Delta])]*LaguerreL[n - 1, 1 - 2*\[Delta],
      2*z/(n - \[Delta])];

coeff1[m_, n_, \[Delta]_] := (m - n + 1)/((m + 2 - 2*\[Delta])*(m + 1));
coeff2[m_, n_, \[Delta]_] := (-1)^(m)*(n - \[Delta])^(2)*m/(4*(m + 2 - 2*\[Delta])*(m + 1)*Factorial[m]);
coeff3[m_, n_, \[Delta]_] := ((n - \[Delta])*(m - 1)*(m - 2*\[Delta])/(n - m + 1) + 2*Znn[n,
      \[Delta]])*BinomialCoeff[n - 2*\[Delta], n - m];
coeff[m_, n_, \[Delta]_] := coeff1[m, n, \[Delta]]*a[m] + coeff2[m, n, \[Delta]]*coeff3[m, n, \[Delta]];
coeffn[n_, \[Delta]_] := (-1)^(n)*(n - \[Delta])^(3)/(8*Gamma[n]);

CoeffTable = Table[coeff[i, n, \[Delta]], {i, 0, n - 1}];
CoeffTable = Prepend[CoeffTable, a[0]];
CoeffTable = Append[CoeffTable, coeffn[n, \[Delta]]];
For[i = 2, i <= n + 1, i++, CoeffTable[[i]] = CoeffTable[[i]] /. a[i - 2] -> CoeffTable[[i - 1]]];

Pn1 = Sum[CoeffTable[[k + 1]]*(2*z/(n - \[Delta]))^(k), {k, 0, n + 1}];
Psin1 = Mn[n, \[Delta]]*E0*z^(1 - \[Delta])*Exp[-z/(n - \[Delta])]*Pn1;

NormalizationCrit = Integrate[psin [n, \[Delta]]*Psin1, {z, 0, Infinity}, Assumptions -> \[Delta] < 0] == 0;
a0 = Solve[NormalizationCrit, a[0]][[1]][[1, 2]];
For[i = 1, i <= n + 1, i++, CoeffTable[[i]] = CoeffTable[[i]] /. a[0] -> a0];

Pn1 = Sum[CoeffTable[[k + 1]]*(2*z/(n - \[Delta]))^(k), {k, 0, n + 1}];
Psin1 = Collect[Mn[n, \[Delta]]*E0*z^(1 - \[Delta])*Exp[-z/(n - \[Delta])]*Pn1, z];

CoeffTable = Prepend[CoeffTable, 0];

Ynn = Integrate[psin [n, \[Delta]]*Mn[n, \[Delta]]*z^(2 - \[Delta])*Exp[-z/(n - \[Delta])]*Pn1, {z, 0, Infinity},
      Assumptions -> \[Delta] < 0];
coeffb1[m_, n_, \[Delta]_] := (m + 1 - n)/((m + 1)*(m + 2 - 2*\[Delta]));
coeffb2[m_, n_, \[Delta]_] := (n - \[Delta])^(2)*((n - \[Delta])*CoeffTable[[m - 2 + 3]] - 2*Znn[n,
      \[Delta]]*CoeffTable[[m - 1 + 3]])/(4*(m + 1)*(m + 2 - 2*\[Delta]));
coeffb3[m_, n_, \[Delta]_] := (-1)^(m)*(n - \[Delta])^(2)*m*Ynn/(2*(m + 1)*(m + 2 -
      2*\[Delta])*Factorial[m])*BinomialCoeff[n - 2*\[Delta], n - m];
coeffb[m_, n_, \[Delta]_] := coeffb1[m, n, \[Delta]]*b[m] + coeffb2[m, n, \[Delta]] + coeffb3[m, n, \[Delta]];
coeffbn[n_, \[Delta]_] := -1/16*(n - \[Delta])^(3)*coeffn[n, \[Delta]];

CoeffbTable = Table[coeffb[i, n, \[Delta]], {i, 0, n + 1}];
CoeffbTable = Prepend[CoeffbTable, b[0]];
CoeffbTable = Append[CoeffbTable, coeffbn[n, \[Delta]]];
For[i = 2, i <= n + 3, i++, CoeffbTable[[i]] = CoeffbTable[[i]] /. b[i - 2] -> CoeffbTable[[i - 1]]];

Pn2 = Sum[CoeffbTable[[k + 1]]*(2*z/(n - \[Delta]))^(k), {k, 0, n + 3}];
Psin2 = FullSimplify[Mn[n, \[Delta]]*E0^(2)*z^(1 - \[Delta])*Exp[-z/(n - \[Delta])]*Pn2];

NormalizationCrit21 = 2*Integrate[psin[n, \[Delta]]*Psin2, {z, 0, Infinity}, Assumptions -> \[Delta] < 0];
NormalizationCrit22 = Integrate[Psin1*Psin1, {z, 0, Infinity}, Assumptions -> \[Delta] < 0];
NormalizationCrit2 = NormalizationCrit21 + NormalizationCrit22 == 0;
b0 = Solve[NormalizationCrit2, b[0]][[1]][[1, 2]];
For[i = 1, i <= n + 3, i++, CoeffbTable[[i]] = CoeffbTable[[i]] /. b[0] -> b0];

Pn2 = Sum[CoeffbTable[[k + 1]]*(2*z/(n - \[Delta]))^(k), {k, 0, n + 3}];
Psin2 = FullSimplify[Mn[n, \[Delta]]*E0^(2)*z^(1 - \[Delta])*Exp[-z/(n - \[Delta])]*Pn2];

```

```
En1 = E0*Integrate[psin[n, \[Delta]]*z*psin[n, \[Delta]], {z, 0, Infinity}, Assumptions -> \[Delta] <= 0];
En2 = E0*Integrate[psin[n, \[Delta]]*z*Psin1, {z, 0, Infinity}, Assumptions -> \[Delta] <= 0];
En3 = E0*Integrate[Psin1*(z - Znn[n, \[Delta]])*Psin1, {z, 0, Infinity}, Assumptions -> \[Delta] <= 0];
En4 = Integrate[E0*Psin1*(z - Znn[n, \[Delta]])*Psin2 + En2*psin[n, \[Delta]]*Psin2, {z, 0, Infinity}, Assumptions -> \[Delta] <= 0];
```

Appendix D

Eigenstate & Matrix Element Plotter Code

The following code generates plots of the unperturbed eigenstates in both the bound and unbound case. It is used to make figures Figure 2.2, Figure 2.3, Figure 2.6 and Figure 2.7. In addition, the code is used to plot the matrix elements Z_{nm} and Z_{km}. It is used to make figures Figure 2.4 and Figure 2.5. The code is written for Python.

Listing D.1: Plotter Code

```
import numpy as np
import mpmath as mp
import matplotlib.pyplot as plt

'Latex Font'
plt.rcParams['text.usetex'] = True # Enable LaTeX rendering
plt.rcParams['font.family'] = 'serif' # Use a serif font
plt.rcParams['font.serif'] = ['Computer Modern'] # LaTeX's default serif font
plt.rcParams['text.latex.preamble'] = r'\usepackage{amsmath}' # Optional: for more LaTeX symbols

'Functions'
def Poch(a, b):
    return mp.gamma(a+b)/mp.gamma(a)
def hypergeom_1F1(a, b, z):
    return mp.hyp1f1(a, b, z)
def hypergeom_2F1(a, b, c, z):
    return mp.hyp2f1(a, b, c, z)

'Constants'
a_He = 7.609 #Nanometer
a_Ne = 1.939 #Nanometer
E_He = 1.316 #meV
E_Ne = 20.26 #meV
delta = 0.0237 #Quantum Defect Parameter
n = 3 #Principle Quantum Number

'Electron Energies'
def en(n):
    en = - 1 / (2*n**(2))
    return en
def ek(k):
    ek = k**(2) / 2
    return ek

'Quantum Defect Energies'
def En(n, delta):
    scriptn = n - delta
    en = - 1 / (2*scriptn**(2))
    return en
def Ek(k):
    Ek = k**(2) / 2
    return Ek

'Electron Eigenstates'
def psin(n, x):
    psi = 2 * n**(-3/2) * x * mp.exp(-x/n) * hypergeom_1F1(1-n,2,2*x/n)
    return psi
def psik(k, x):
    Factor1 = 2*mp.sqrt(k) / mp.sqrt(1 - mp.exp(-2*mp.pi/k))
    psi = Factor1 * x * mp.exp(-1j*k*x) * hypergeom_1F1(1+1j/k, 2, 2j*k*x)
    return psi

'Quantum Defect Eigenstates'
def Psin(n, delta, x):
    scriptn = n - delta
    Factor = 2**(1-delta) * mp.sqrt(Poch(n, 1-2*delta)) / (scriptn**(2-delta) * mp.gamma(2-2*delta))
    psi = Factor * x**(1-delta) * mp.exp(-x/scriptn) * hypergeom_1F1(1-n, 2-2*delta, 2*x/scriptn)
    return psi
def Psik(k, delta, x):
    Factor = (2*k)**(1-delta) * mp.fabs(mp.gamma(1-delta-1j/k)) / (mp.sqrt(2*mp.pi) * mp.exp(-mp.pi/(2*k)) * mp.gamma(2-2*delta))
    psi = Factor * x**(1-delta) * mp.exp(-1j*k*x) * hypergeom_1F1(1-delta+1j/k, 2-2*delta, 2j*k*x)
    return psi

'Electron Matrix Elements'
def znm(n, m):
    if n == m:
        znm = 3 * n**(2) / 2
    else:
        Factor1 = 8 * (-1)**(n) * (n*m)**(5/2) / (m-n)**(4) * ((m-n)/(m+n))**(m+n)
        Factor2 = m * (n-m) / (m+n) * mp.hyp2f1(1-n, 1-m, 2, -4*m*n/(m-n)**(2))
        Factor3 = (m-1) * mp.hyp2f1(1-n, 2-m, 2, -4*m*n/(m-n)**(2))
        znm = Factor1 * (Factor2 + Factor3)
    return znm
def zkm(k, m):
    Factor1 = 8 * (-1)**(m) * m**(5/2) * k**(1/2) / ((1 + 1j*k*m)**(4) * mp.sqrt(1 - mp.exp(-2*mp.pi/k)))
    Factor2 = ((1-1j*k*m)/(1+1j*k*m))**(-1j/k-m)
    Factor3 = m * (1+1j*k*m) / (1 - 1j*k*m) * mp.hyp2f1(1-1j/k, 1-m, 2, 4j*k*m/(1+1j*k*m)**(2))
    Factor4 = (m-1) * mp.hyp2f1(1-1j/k, 2-m, 2, 4j*k*m/(1+1j*k*m)**(2))
    zkm = Factor1 * Factor2 * (Factor3 + Factor4)
    return zkm

'Quantum Defect Matrix Elements'
def Znm(n, m, delta):
    if n == m:
        Znm = 3 * n**(2) / 2 - delta * (3 * n - 1/2 - delta)
    else:
        scriptn = n - delta
        scriptm = m - delta
        Factor1 = (-1)**(n) * 2**(3-2*delta) * (scriptn*scriptm)**(2-delta) * mp.sqrt(Poch(n, 1-2*delta) * Poch(m, 1-2*delta))
        Factor2 = (scriptm + scriptn)**(-2*delta) * (scriptm - scriptn)**(4) * mp.gamma(2-2*delta)
        Factor3 = ((scriptm - scriptn)/(scriptm + scriptn))**(n + m)
        Factor4 = scriptm * (scriptn - scriptm) / (scriptn + scriptm) * hypergeom_2F1(1-n, 1-m, 2-2*delta, -4*scriptn*scriptm / (scriptn - scriptm)**(2))
        Factor5 = (m-1) * hypergeom_2F1(1-n, 2-m, 2-2*delta, -4*scriptn*scriptm / (scriptn - scriptm)**(2))
        Znm = Factor1 / Factor2 * Factor3 * (Factor4 + Factor5)
    return Znm
def Zkm(k, m, delta):
    scriptm = m - delta
    N = (2*k)**(1-delta) * mp.fabs(mp.gamma(1-delta-1j/k)) / (mp.sqrt(2*mp.pi) * mp.exp(-mp.pi/(2*k)) * mp.gamma(2-2*delta))
    Factor1 = (-1)**(m) * (2*scriptm)**(2-delta) * N * mp.sqrt(Poch(m, 1-2*delta))
    Factor2 = (1 + 1j*k*scriptm)**(4-2*delta)
    Factor3 = ((1-1j*k*scriptm) / (1 + 1j*k*scriptm))**(-1j/k - scriptm)
    Factor4 = scriptm * (1 + 1j*k*scriptm) / (1 - 1j*k*scriptm) * hypergeom_2F1(1 - delta - 1j/k, 1 - m, 2 - 2*delta, 4j*k*scriptm / (1 + 1j*k*scriptm)**(2))
    Factor5 = (m-1) * hypergeom_2F1(1 - delta - 1j/k, 2 - m, 2 - 2*delta, 4j*k*scriptm / (1 + 1j*k*scriptm)**(2))
    Zkm = Factor1 / Factor2 * Factor3 * (Factor4 + Factor5)
    return Zkm

'Oscillator Strengths'
```

```
def gmn(m, n, delta):
    gmn = 2 * (En(n, delta) - En(m, delta)) * mp.fabs(Znm(n, m, delta))**(2)
    return gmn
def gmk(m, k, delta):
    gmk = 2 * (Ek(k) - En(m, delta)) * mp.fabs(Zkm(k, m, delta))**(2)
    return gmk

'Oscillator Moments'
def G(m, p, delta):
    Gbound = sum(gmn(m, n, delta)*(En(n, delta) - En(m, delta))**(p) for n in range(1, 5000) if n != m)
    Gunbound = mp.quad(lambda k: gmk(m, k, delta)*(Ek(k) - En(m, delta))**(p), [1e-6, mp.inf])
    G = Gbound + Gunbound
    return G

'Eigenstate Plotting Parameters'
Resolution = 1000
z1max = 95 #95
z2max = 60 #60
z3max = 30 #30
z4max = 75 #50

'Initial Calculations, Bound Electron Density'
z1 = np.linspace(0 + 1e-6, z1max, Resolution)
PsiBoundElectronDensityn3 = np.zeros(len(z1))
PsiBoundElectronDensityn4 = np.zeros(len(z1))
PsiBoundElectronDensityn5 = np.zeros(len(z1))
PsiBoundElectronDensityn6 = np.zeros(len(z1))
for i in range(len(z1)):
    PsiBoundElectronDensityn3[i] = (psin(1, z1[i]))
    PsiBoundElectronDensityn4[i] = (psin(2, z1[i]))
    PsiBoundElectronDensityn5[i] = (psin(3, z1[i]))
    PsiBoundElectronDensityn6[i] = (psin(4, z1[i]))

'Plotting, Bound Electron Density'
fig, ax1 = plt.subplots(figsize=(7.5,5))
ax1.set_xlabel(r"$\mathrm{Distance} \ \mathrm{From} \ \mathrm{Substrate}, \ z$", fontsize=16)
ax1.set_ylabel(r"$\mathrm{Wave} \ \mathrm{Function}, \ \psi_n$", fontsize=16)
ax1.set_xlim([0, 20])
ax1.set_ylim([0, 0.105])
ax1.tick_params(axis='x', labelsize=12)
ax1.tick_params(axis='y', labelsize=12)
ax1.grid('true')
ax1.plot(z1, PsiBoundElectronDensityn3, label=r"$n = 1$")
ax1.plot(z1, PsiBoundElectronDensityn4, label=r"$n = 2$")
ax1.plot(z1, PsiBoundElectronDensityn5, label=r"$n = 3$")
ax1.plot(z1, PsiBoundElectronDensityn6, label=r"$n = 4$")
ax1.legend(fontsize=16, ncol=2)
plt.savefig("BoundElectronDensityPlot190225.pdf", dpi=300)

'Initial Calculations, Unbound Electron Density'
z2 = np.linspace(0 + 1e-6, z2max, Resolution)
PsiUnboundElectronDensityk14 = np.zeros(len(z2))
PsiUnboundElectronDensityk18 = np.zeros(len(z2))
PsiUnboundElectronDensitynk112 = np.zeros(len(z2))
PsiUnboundElectronDensitynk116 = np.zeros(len(z2))
for i in range(len(z2)):
    PsiUnboundElectronDensityk14[i] = np.real(psik(1/4, z2[i]))
    PsiUnboundElectronDensityk18[i] = np.real(psik(1/8, z2[i]))
    PsiUnboundElectronDensitynk112[i] = np.real(psik(1/12, z2[i]))
    PsiUnboundElectronDensitynk116[i] = np.real(psik(1/16, z2[i]))

'Plotting, Unbound Electron Density'
fig, ax2 = plt.subplots(figsize=(7.5,5))
ax2.set_xlabel(r"$\mathrm{Distance} \ \mathrm{From} \ \mathrm{Substrate}, \ z$", fontsize=16)
ax2.set_ylabel(r"$\mathrm{Wave} \ \mathrm{Function}, \ \psi_k$", fontsize=16)
ax2.set_xlim([0, 20])
```

```
ax2.set_ylim([-0.65,0.78])
ax2.tick_params(axis='x', labelsize=12)
ax2.tick_params(axis='y', labelsize=12)
ax2.grid('true')
ax2.plot(z2, PsiUnboundElectronDensityk14, label=r"$k = 1/4$")
ax2.plot(z2, PsiUnboundElectronDensityk18, label=r"$k = 1/8$")
ax2.plot(z2, PsiUnboundElectronDensitynk112, label=r"$k = 1/12$")
ax2.plot(z2, PsiUnboundElectronDensitynk116, label=r"$k = 1/16$")
ax2.legend(fontsize=16, ncol=3, loc="upper left")
plt.savefig("UnboundElectronDensityPlot190225.pdf", dpi=300)

'Initial Calculations, Bound Defect Density'
z3 = np.linspace(0 + 1e-6, z3max, Resolution)
PsiBoundDefectDensityn0 = np.zeros(len(z3))
PsiBoundDefectDensityn02 = np.zeros(len(z3))
PsiBoundDefectDensityn01 = np.zeros(len(z3))
PsiBoundDefectDensityn05 = np.zeros(len(z3))
PsiBoundDefectDensityn09 = np.zeros(len(z3))
PsiBoundDefectDensityn1 = np.zeros(len(z3))
for i in range(len(z3)):
   PsiBoundDefectDensityn0[i] = np.real(Psin(2, 0, z3[i]))
   PsiBoundDefectDensityn02[i] = np.real(Psin(2, 0.2, z3[i]))
   PsiBoundDefectDensityn01[i] = np.real(Psin(2, 0.1, z3[i]))
   PsiBoundDefectDensityn05[i] = np.real(Psin(2, 0.5, z3[i]))
   PsiBoundDefectDensityn09[i] = np.real(Psin(2, 0.9, z3[i]))
   PsiBoundDefectDensityn1[i] = np.real(Psin(1, 0, z3[i]))

'Plotting, Bound Defect Density'
fig, ax3 = plt.subplots(figsize=(7.5,5))
ax3.set_xlabel(r"$\mathrm{Distance} \ \mathrm{From} \ \mathrm{Substrate}, \ z$", fontsize=16)
ax3.set_ylabel(r"$\mathrm{Wave} \ \mathrm{Function}, \ \psi_{n - \delta}$", fontsize=16)
ax3.set_xlim([0, 20])
ax3.set_ylim([-0.75,0.8])
ax3.tick_params(axis='x', labelsize=12)
ax3.tick_params(axis='y', labelsize=12)
ax3.grid('true')
ax3.plot(z3, PsiBoundDefectDensityn0, label=r"$n = 2, \ \delta = 0$")
ax3.plot(z3, PsiBoundDefectDensityn02, label=r"$n = 2, \ \delta = 0.2$")
ax3.plot(z3, PsiBoundDefectDensityn01, label=r"$n = 2, \ \delta = 0.1$")
ax3.plot(z3, PsiBoundDefectDensityn05, label=r"$n = 2, \ \delta = 0.5$")
ax3.plot(z3, PsiBoundDefectDensityn09, label=r"$n = 2, \ \delta = 0.9$")
ax3.plot(z3, PsiBoundDefectDensityn1, label=r"$n = 1, \ \delta = 0$")
ax3.legend(fontsize=16,ncol=2)
plt.savefig("BoundDefectDensityPlot190225.pdf", dpi=300)

'Initial Calculations, Unound Defect Density'
z4 = np.linspace(0 + 1e-6, z4max, Resolution)
PsiUnboundDefectDensityk0 = np.zeros(len(z4))
PsiUnboundDefectDensityk01 = np.zeros(len(z4))
PsiUnboundDefectDensityk02 = np.zeros(len(z4))
PsiUnboundDefectDensityk05 = np.zeros(len(z4))
PsiUnboundDefectDensityk09 = np.zeros(len(z4))
for i in range(len(z4)):
   PsiUnboundDefectDensityk0[i] = np.real(Psik(1/16, 0, z4[i]))
   PsiUnboundDefectDensityk01[i] = np.real(Psik(1/16, 0.1, z4[i]))
   PsiUnboundDefectDensityk02[i] = np.real(Psik(1/16, 0.2, z4[i]))
   PsiUnboundDefectDensityk05[i] = np.real(Psik(1/16, 0.5, z4[i]))
   PsiUnboundDefectDensityk09[i] = np.real(Psik(1/16, 0.9, z4[i]))

'Plotting, Unbound Defect Density'
fig, ax4 = plt.subplots(figsize=(7.5,5))
ax4.set_xlabel(r"$\mathrm{Distance} \ \mathrm{From} \ \mathrm{Substrate}, \ z$", fontsize=16)
ax4.set_ylabel(r"$\mathrm{Wave} \ \mathrm{Function}, \ \psi_{1/16}$", fontsize=16)
ax4.set_xlim([0, 20])
ax4.set_ylim([-0.4, 0.55])
ax4.tick_params(axis='x', labelsize=12)
```

```
ax4.tick_params(axis=’y’, labelsize=12)
ax4.grid(’true’)
ax4.plot(z4, PsiUnboundDefectDensityk0, label=r"$\delta = 0$")
ax4.plot(z4, PsiUnboundDefectDensityk01, label=r"$\delta = 0.1$")
ax4.plot(z4, PsiUnboundDefectDensityk02, label=r"$\delta = 0.2$")
ax4.plot(z4, PsiUnboundDefectDensityk05, label=r"$\delta = 0.5$")
ax4.plot(z4, PsiUnboundDefectDensityk09, label=r"$\delta = 0.9$")
ax4.legend(fontsize=16, ncol=3, loc="upper left")
plt.savefig("UnboundDefectDensityPlot190225.pdf", dpi=300)

’Initial Calculations, Bound Matrix Element’
StateCount = 40
StateList = np.zeros(StateCount)
for i in range(len(StateList)):
    StateList[i] = i + 1
NStates = np.array([5, 10, 15, 20, 25, 30, 35])
ZnmMatrix = np.zeros((len(NStates)+1, StateCount))
for i in range(len(NStates)):
    for j in range(StateCount):
        ZnmMatrix[i,j] = Znm(NStates[i], j+1, 0)
for i in range(StateCount):
    ZnmMatrix[-1, i] = Znm(i+1, i+1, 0)

’Bound Matrix Element Plotting’
fig, ax5 = plt.subplots(figsize=(7.5,5))
ax5.set_xlabel(r"$\mathrm{Principle} \ \mathrm{Quantum} \ \mathrm{Number}, \ m$", fontsize=16)
ax5.set_ylabel(r"$\mathrm{Matrix} \ \mathrm{Elements}, \ Z_{nm}$", fontsize=16)
ax5.set_xlim([2, 40])
ax5.set_ylim([-430, 1870])
ax5.tick_params(axis=’x’, labelsize=12)
ax5.tick_params(axis=’y’, labelsize=12)
ax5.grid(’true’)

ax5.scatter(StateList, ZnmMatrix[7, :], label=r"$n = m$", color=’grey’)
ax5.scatter(StateList, ZnmMatrix[0, :], label=r"$n = 5$", marker=’x’, color=’green’)
ax5.scatter(StateList, ZnmMatrix[1, :], label=r"$n = 10$", marker=’x’, color=’blue’)
ax5.scatter(StateList, ZnmMatrix[2, :], label=r"$n = 15$", marker=’x’, color=’red’)
ax5.scatter(StateList, ZnmMatrix[3, :], label=r"$n = 20$", marker=’x’, color=’orange’)
ax5.scatter(StateList, ZnmMatrix[4, :], label=r"$n = 25$", marker=’x’, color=’cyan’)
ax5.scatter(StateList, ZnmMatrix[5, :], label=r"$n = 30$", marker=’x’, color=’magenta’)
ax5.scatter(StateList, ZnmMatrix[6, :], label=r"$n = 35$", marker=’x’, color=’lime’)

ax5.plot(StateList, ZnmMatrix[0, :], linestyle=’dotted’, color=’green’)
ax5.plot(StateList, ZnmMatrix[1, :], linestyle=’dotted’, color=’blue’)
ax5.plot(StateList, ZnmMatrix[2, :], linestyle=’dotted’, color=’red’)
ax5.plot(StateList, ZnmMatrix[3, :], linestyle=’dotted’, color=’orange’)
ax5.plot(StateList, ZnmMatrix[4, :], linestyle=’dotted’, color=’cyan’)
ax5.plot(StateList, ZnmMatrix[5, :], linestyle=’dotted’, color=’magenta’)
ax5.plot(StateList, ZnmMatrix[6, :], linestyle=’dotted’, color=’lime’)
ax5.plot(StateList, ZnmMatrix[7, :], linestyle=’dotted’, color=’grey’)
ax5.legend(fontsize=16, ncol=2)
plt.savefig("BoundBoundMatrixElementsPlot190225.pdf", dpi=300)

’Initial Calculations, Unbound Matrix Element’
StateList = np.linspace(1e-6, 1.25, 2500)
StateCount = len(StateList)
NStates = np.array([1, 2, 3, 4, 5, 6, 7])
ZnkMatrix = np.zeros((len(NStates), StateCount))
for i in range(len(NStates)):
    for j in range(StateCount):
        ZnkMatrix[i, j] = mp.re(Zkm(StateList[j], NStates[i], 0))

’Unbound Matrix Element Plotting’
fig, ax6 = plt.subplots(figsize=(7.5,5))
ax6.set_xlabel(r"$\mathrm{Wavenumber}, \ k$", fontsize=16)
ax6.set_ylabel(r"$\mathrm{Matrix} \ \mathrm{Elements}, \ Z_{km}$", fontsize=16)
```

```
ax6.set_xlim([0.00, 0.8])
ax6.set_ylim([-8.5, 0.2])
ax6.tick_params(axis='x', labelsize=12)
ax6.tick_params(axis='y', labelsize=12)
ax6.grid('true')
ax6.plot(StateList, ZnkMatrix[0, :], label=r"$m = 1$")
ax6.plot(StateList, ZnkMatrix[1, :], label=r"$m = 2$")
ax6.plot(StateList, ZnkMatrix[2, :], label=r"$m = 3$")
ax6.plot(StateList, ZnkMatrix[3, :], label=r"$m = 4$")
ax6.plot(StateList, ZnkMatrix[4, :], label=r"$m = 5$")
ax6.plot(StateList, ZnkMatrix[5, :], label=r"$m = 6$")
ax6.plot(StateList, ZnkMatrix[6, :], label=r"$m = 7$")
ax6.legend(fontsize=16, ncol=2)
plt.savefig("BoundUnboundMatrixElementsPlot190225.pdf", dpi=300)
```

Appendix E

Perturbed Energy Plotter Code

The following code generates plots of the perturbed energy shown in section 3.5. It is used to make figures Figure 3.2, Figure 3.3, Figure 3.6, Figure 3.7, Figure 3.5 and Figure 3.4. The code is written for Python.

Listing E.1: Perturbed Energy Plotter Code

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import mpmath as mp
from scipy.linalg import eig

'LaTeX'
plt.rcParams['text.usetex'] = True # Enable LaTeX rendering
plt.rcParams['font.family'] = 'serif' # Use a serif font
plt.rcParams['font.serif'] = ['Computer Modern'] # LaTeX's default serif font
plt.rcParams['text.latex.preamble'] = r'\usepackage{amsmath}' # Optional: for more LaTeX symbols

'Functions'
def Poch(a, b):
    Poch = mp.gamma(a+b)/mp.gamma(a)
    return Poch
def kdelta(n,m):
    if n == m:
        kdelta = 1
    else:
        kdelta = 0
    return kdelta

'Hypergeometric Approximant Coefficients & 2F1 Function'
#n=1, Coulomb
def HypApproxN1C(E0):
    C1n1 = -1/2
    C2n1 = -(1247 + 5*mp.sqrt(3817))/2534
    C3n1 = -(1247 - 5*mp.sqrt(3817))/2534
    C4n1 = 48/29
    C5n1 = -1267/58
    return C1n1 * mp.hyp2f1(C2n1, C3n1, C4n1, C5n1*E0)

#n=2, Coulomb
def HypApproxN2C(E0):
    C1n2 = -1/8
    C2n2 = (-501539-5j*mp.sqrt(76756895))/946058
    C3n2 = (-501539+5j*mp.sqrt(76756895))/946058
    C4n2 = 11163/8009
    C5n2 = -1892116/8009
    return C1n2 * mp.hyp2f1(C2n2, C3n2, C4n2, C5n2*E0)

#n=3, Coulomb
def HypApproxN3C(E0):
    C1n3 = - 1/18
    C2n3 = - (24240121+5j*mp.sqrt(291559959335))/45863962
    C3n3 = - (24240121-5j*mp.sqrt(291559959335))/45863962
    C4n3 = 40032/36101
    C5n3 = - 68795943/72202
    return C1n3 * mp.hyp2f1(C2n3, C3n3, C4n3, C5n3*E0)

#n=4, Coulomb
def HypApproxN4C(E0):
    C1n4 = - 1/32
    C2n4 = - (503957+5*mp.sqrt(209041297)) / 971454
    C3n4 = - (503957-5*mp.sqrt(209041297)) / 971454
    C4n4 = 32007/35561
    C5n4 = -93259584/35561
    return C1n4 * mp.hyp2f1(C2n4, C3n4, C4n4, C5n4*E0)

#n=5, Coulomb
def HypApproxN5C(E0):
    C1n5 = -1/50
    C2n5 = -(51689+mp.sqrt(143953681))/101274
    C3n5 = -(51689-mp.sqrt(143953681))/101274
    C4n5 = 1248/1609
    C5n5 = -18988875/3218
    return C1n5 * mp.hyp2f1(C2n5, C3n5, C4n5, C5n5*E0)

#n=1, Kratzer
def HypApproxN1K(delta, E0):
    C1n1 = -1/(2*(1-delta)**(2))
    c2n11 =
          53248*delta**(6)-592384*delta**(5)+2716560*delta**(4)-6615248*delta**(3)+9148284*delta**(2)-6985900*delta+2385625
    C2n1 = (-512*delta**(3)+3604*delta**(2)-8306*delta - mp.sqrt(c2n11) + 6235) /
          (2*(544*delta**(3)-3716*delta**(2)+8428*delta-6335))
    C3n1 = (-512*delta**(3)+3604*delta**(2)-8306*delta + mp.sqrt(c2n11) + 6235) /
          (2*(544*delta**(3)-3716*delta**(2)+8428*delta-6335))
    C4n1 = (12*(2*delta**(2)-13*delta+20))/(40*delta**(2)-156*delta+145)
    C5n1 = - (delta-1)**(3)*(544*delta**(3)-3716*delta**(2)+8428*delta-6335)/(2*(40*delta**(2)-156*delta+145))
    return C1n1 * mp.hyp2f1(C2n1, C3n1, C4n1, C5n1*E0)

#n=2, Kratzer
def HypApproxN2K(delta, E0):
    C1n2 = -1 / (2 * (2 - delta) ** 2)
    C2n2 = -((512 *delta**13-25660 *delta**12+493530 *delta**11-5224103 *delta**10+
    35839082 *delta**9-173241681 *delta**8+615689786 *delta**7-1637600502 *delta**6+
    3262154856 *delta**5-4801824208 *delta**4+5071626928 *delta**3-3639312576 *delta**2+
    1591735616 *delta+mp.sqrt(53248 *delta**26-4865536 *delta**25+220906000 *delta**24-
    6323611952 *delta**23+123196658796 *delta**22-1712391092860 *delta**21+17667961442385 *delta**20-
    139757465232048 *delta**19+868765095012202 *delta**18-4320071879129112 *delta**17+
    17384202828444125 *delta**16-56922213390268508 *delta**15+151450951712282512 *delta**14-
    323757335479499240 *delta**13+539389841579808164 *delta**12-644041860148118400 *delta**11+
    379943440501002944 *delta**10+430667072118589504 *delta**9-1620805131652820736 *delta**8+
    2634019941598720256 *delta**7-2895911963677591808 *delta**6+2312754659419547648 *delta**5-
    1343135933310478336 *delta**4+546510927294107648 *delta**3-143103025678954496 *delta**2+
    19917436305817600 *delta-785990604800000)-320984960)/(2*(544 *delta**13-26924 *delta**12+
    528708 *delta**11-5673505 *delta**10+38953468 *delta**9-186154419 *delta**8+648017830 *delta**7-
    1679774226 *delta**6+3259266696 *delta**5-4687666112 *delta**4+4864216496 *delta**3-
    3451153344 *delta**2+1501519168 *delta-302738560)))
    C3n2 = (-512 *delta**13+25660 *delta**12-493530 *delta**11+5224103 *delta**10-
    35839082 *delta**9+173241681 *delta**8-615689786 *delta**7+1637600502 *delta**6-
    3262154856 *delta**5+4801824208 *delta**4-5071626928 *delta**3+3639312576 *delta**2-
    1591735616 *delta+mp.sqrt(53248 *delta**26-4865536 *delta**25+220906000 *delta**24-
    6323611952 *delta**23+123196658796 *delta**22-1712391092860 *delta**21+
    17667961442385 *delta**20-139757465232048 *delta**19+868765095012202 *delta**18-
    4320071879129112 *delta**17+17384202828444125 *delta**16-56922213390268508 *delta**15+
    151450951712282512 *delta**14-323757335479499240 *delta**13+539389841579808164 *delta**12-
    644041860148118400 *delta**11+379943440501002944 *delta**10+430667072118589504 *delta**9-
    1620805131652820736 *delta**8+2634019941598720256 *delta**7-2895911963677591808 *delta**6+
    2312754659419547648 *delta**5-1343135933310478336 *delta**4+546510927294107648 *delta**3-
    143103025678954496 *delta**2+19917436305817600 *delta-785990604800000)+320984960)/ (2*
    (544 *delta**13-26924 *delta**12+528708 *delta**11-5673505 *delta**10+
    38953468 *delta**9-186154419 *delta**8+648017830 *delta**7-1679774226 *delta**6
    +3259266696 *delta**5-4687666112 *delta**4+4864216496 *delta**3-
    3451153344 *delta**2+1501519168 *delta-302738560))
    C4n2 = (12*(2 *delta**11-95 *delta**10+1303 *delta**9-10609 *delta**8+
    61997 *delta**7-266266 *delta**6+817660 *delta**5-1742602 *delta**4+
    2500166 *delta**3-2303828 *delta**2+1237792 *delta-297680))/(40 *delta**11-
    1588 *delta**10+27085 *delta**9-242174 *delta**8+1314085 *delta**7-
    4710492 *delta**6+11682432 *delta**5-20458944 *delta**4+25205304 *delta**3-
    21048752 *delta**2+10780224 *delta-2562880)
    C5n2 = -(((delta-2)**2*(544 *delta**13-26924 *delta**12+528708 *delta**11-
    5673505 *delta**10+38953468 *delta**9-186154419 *delta**8+648017830 *delta**7-
    1679774226 *delta**6+3259266696 *delta**5-4687666112 *delta**4+4864216496 *delta**3-
    3451153344 *delta**2+1501519168 *delta-302738560))/(80 *delta**11-3176 *delta**10+
    54170 *delta**9-484348 *delta**8+2628170 *delta**7-9420984 *delta**6+23364864 *delta**5-
    40917888 *delta**4+50410608 *delta**3-42097504 *delta**2+21560448 *delta-5125760))
    return C1n2 * mp.hyp2f1(C2n2, C3n2, C4n2, C5n2 * E0)

#n=3, Kratzer
def HypApproxN3K(delta, E0):
    C1n3 = -1 / (2 * (3 - delta) ** 2)
    C2n3 = -((4096 *delta**14-334624 *delta**13+10283040 *delta**12-
    171708824 *delta**11+1847321716 *delta**10-13985810376 *delta**9+
    78099431893 *delta**8-329265121734 *delta**7+1056393436968 *delta**6-
    2569607110818 *delta**5+4667935890798 *delta**4-6146327425986 *delta**3+
    5551703608644 *delta**2-3083163040278 *delta+mp.sqrt(3407872 *delta**28-
    506494976 *delta**27+37655593984 *delta**26-1764006062080 *delta**25+
    55887406938624 *delta**24-1255343278942976 *delta**23+20869291184109888 *delta**22-
    266243717492491008 *delta**21+2682151279399339216 *delta**20-21806404639741428784 *delta**19+
    145429022081615953640 *delta**18-805057854574555196256 *delta**17+3729648996698748423097 *delta**16-
    14533005658292439404652 *delta**15+47729685373841611780452 *delta**14-131989933474720825454820 *delta**13+
    305934794686410708774276 *delta**12-588766667365929343398588 *delta**11+924527653358476482248652 *delta**10-
    1145736525297690283509108 *delta**9+1038841962876324945336726 *delta**8-527327437883898313420164 *delta**7-
    182028566402349335303076 *delta**6+703218794214445758006804 *delta**5-792046836512609467996332 *delta**4+
    543935867742267127455708 *delta**3-241037818940377813342644 *delta**2+64070998601879537460900 *delta-
    7844187640165681584375)+795197169405)/(2*(4352 *delta**14-351008 *delta**13+11060976 *delta**12-
    187965376 *delta**11+2032222544 *delta**10-15283843494 *delta**9+84039262817 *delta**8-
    347023211202 *delta**7+1088416038384 *delta**6-2590360855650 *delta**5+4616027849502 *delta**4-
    5982337092366 *delta**3+5336748834480 *delta**2-2936191649544 *delta+752283636705)))
    C3n3 = (-4096 *delta**14+334624 *delta**13-10283040 *delta**12+
    171708824 *delta**11-1847321716 *delta**10+13985810376 *delta**9-
    78099431893 *delta**8+329265121734 *delta**7-1056393436968 *delta**6+
    2569607110818 *delta**5-4667935890798 *delta**4+6146327425986 *delta**3-
    5551703608644 *delta**2+3083163040278 *delta+mp.sqrt(3407872 *delta**28-
    506494976 *delta**27+37655593984 *delta**26-1764006062080 *delta**25+
    55887406938624 *delta**24-1255343278942976 *delta**23+20869291184109888 *delta**22-
    266243717492491008 *delta**21+2682151279399339216 *delta**20-21806404639741428784 *delta**19+
    145429022081615953640 *delta**18-805057854574555196256 *delta**17+3729648996698748423097 *delta**16-
    14533005658292439404652 *delta**15+47729685373841611780452 *delta**14-131989933474720825454820 *delta**13+
    305934794686410708774276 *delta**12-588766667365929343398588 *delta**11+924527653358476482248652 *delta**10-
    1145736525297690283509108 *delta**9+1038841962876324945336726 *delta**8-527327437883898313420164 *delta**7-
    182028566402349335303076 *delta**6+703218794214445758006804 *delta**5-792046836512609467996332 *delta**4+
    543935867742267127455708 *delta**3-241037818940377813342644 *delta**2+64070998601879537460900 *delta-
    7844187640165681584375)-795197169405) / (2*(4352 *delta**14-351008 *delta**13+11060976 *delta**12-
    187965376 *delta**11+2032222544 *delta**10-15283843494 *delta**9+ 84039262817 *delta**8-
    347023211202 *delta**7+1088416038384 *delta**6-2590360855650 *delta**5+4616027849502 *delta**4-
    5982337092366 *delta**3+5336748834480 *delta**2-2936191649544 *delta+752283636705))
    C4n3 = (12*(16 *delta**12-1248 *delta**11+26640 *delta**10-334828 *delta**9+
    3003207 *delta**8-19872651 *delta**7+95851221 *delta**6-332120277 *delta**5+
    815342463 *delta**4-1387270395 *delta**3+1564519293 *delta**2-1059418521 *delta+
    328312440))/(320 *delta**12-20800 *delta**11+581032 *delta**10-8429804 *delta**9+
    74433694 *delta**8-438902833 *delta**7+1819835787 *delta**6-5441785677 *delta**5+
    11786301603 *delta**4-18165997899 *delta**3+18984472929 *delta**2-12094170507 *delta+3552879915)
    C5n3 = -(((delta-3)**2*(4352 *delta**14-351008 *delta**13+11060976 *delta**12-
    187965376 *delta**11+2032222544 *delta**10-15283843494 *delta**9+
    84039262817 *delta**8-347023211202 *delta**7+1088416038384 *delta**6-
    2590360855650 *delta**5+4616027849502 *delta**4-5982337092366 *delta**3+
    5336748834480 *delta**2-2936191649544 *delta+752283636705))/(2*(320 *delta**12-
    20800 *delta**11+581032 *delta**10-8429804 *delta**9+74433694 *delta**8-
    438902833 *delta**7+1819835787 *delta**6-5441785677 *delta**5+11786301603 *delta**4-
    18165997899 *delta**3+18984472929 *delta**2-12094170507 *delta+3552879915)))
    return C1n3 * mp.hyp2f1(C2n3, C3n3, C4n3, C5n3 * E0)

'Perturbative Expansions'
#n=1, Coulomb
def EExpansionN1C(E0):
    ExpansionN1C = -1/2 + 3/2*E0 - 3/2*E0**(2) + 27/4*E0**(3) - 795/16*E0**(4)
    return ExpansionN1C

#n=2, Coulomb
def EExpansionN2C(E0):
    ExpansionN2C = -1/8 + 6*E0 - 66*E0**(2) + 3312*E0**(3) - 271680*E0**(4)
    return ExpansionN2C

#n=3, Coulomb
def EExpansionN3C(E0):
    ExpansionN3C = -1/18 + 27/2*E0 - 1377/2*E0**(2) + 610173/4**E0**(3) - 864772605/16*E0**(4)
    return ExpansionN3C

#n=4, Coulomb
def EExpansionN4C(E0):
    ExpansionN4C = -1/32 + 24*E0 - 3744*E0**(2) + 2469888*E0**(3) - 2553937920*E0**(4)
    return ExpansionN4C

#n=5, Coulomb
def EExpansionN5C(E0):
    ExpansionN5C = -1/50 + 75/2*E0 - 28125/2*E0**(2) + 87890625/4*E0**(3) - 849462890625/16*E0**(4)
    return ExpansionN5C

#n=1, Kratzer
def EExpansionN1K(delta, E0):
    E = - 1/(2*(1-delta)**(2))
    E1 = 1/2*(1-delta)*(3-2*delta)*E0
    E2 = -1/4*(1-delta)**(4)*(2-delta)*(3-2*delta)*E0**(2)
    E3 = 1/8*(1-delta)**(7)*(2-delta)*(3-2*delta)*(9-4*delta)*E0**(3)
    E4 = -1/32*(1-delta)**(10)*(2-delta)*(3-2*delta)*(48*delta**(2)-225*delta+265)*E0**(4)
    ExpansionN1K = E + E1 + E2 + E3 + E4
    return ExpansionN1K

'Known Energies'
EMinusQuarterN1C = -0.91414317937917-0.12477412204721904j
EPlusQuarterN1C = -0.1764014280188822-2.803254811469 66e-14j
EMinusQuarterN2C = -0.9381725148398229-0.7117689604692352j
EPlusQuarterN2C = 0.765530635337687+1.0554996366393326e-13j
EMinusQuarterN3C = 0
EPlusQuarterN3C = 1.335924172843064-3.1807835307557864e-13j
EMinusQuarterN4C = 0
EPlusQuarterN4C = 1.7979280729012332-3.3006381141237694e-13j
EMinusQuarterN5C = 0
EPlusQuarterN5C = 2.2032494848092825+6.139979727080982e-13j

'Sturmian Basis'
def phi(n, delta, kappa, z):
    F1 = (2*kappa)**(1-delta) / mp.sqrt(Poch(n, 1-2*delta))
    phi = F1 * z**(1-delta) * mp.exp(-kappa*z) * mp.laguerre(n-1, 1-2*delta, 2*kappa*z)
    return phi

'Sturmian Inner Products'
def phiphi(m, n, delta, kappa):
    F1 = mp.sqrt(Poch(n, 1-2*delta)) / (2*kappa*mp.sqrt(Poch(m, 1-2*delta)))
    F2 = 2*(m-delta)*kdelta(m,n) - m*kdelta(m,n-1) - (m-2*delta)*kdelta(m-1,n)
    return F1*F2

def phizphi(m, n, delta, kappa):
    if m==1:
        F1 = 2*(m-delta)*phiphi(m, n, delta, kappa) - mp.sqrt(m*(m+1-2*delta))*phiphi(m+1, n, delta, kappa)
        phizphi = F1 / (2*kappa)
        return phizphi
    else:
        F1 = 2*(m-delta)*phiphi(m, n, delta, kappa) - mp.sqrt(m*(m+1-2*delta))*phiphi(m+1, n, delta, kappa)
        F2 = mp.sqrt((m-2*delta)*(m-1))*phiphi(m-1, n, delta, kappa)
        phizphi = (F1 - F2) / (2*kappa)
        return phizphi

'Sturmian Matrix Elements'
def Hnm(n, m, delta, kappa, theta, E0):
    F1 = kappa*(m-delta)*mp.exp(-2j*theta)*kdelta(m, n)
    F2 = 1/2*kappa**(2)*mp.exp(-2j*theta)*phiphi(m, n, delta, kappa)
    F3 = mp.exp(-1j*theta)*kdelta(m,n)
    F4 = mp.exp(1j*theta)*E0*phizphi(m, n, delta, kappa)
    Hnm = F1 - F2 - F3 + F4
    return Hnm
def Knm(n, m, delta, kappa):
    Knm = phiphi(m, n, delta, kappa)
    return Knm

'-------------------- Theta Dependence --------------------'

'Parameters, Theta Dependence'
ThetaT = np.linspace(-np.pi/2, np.pi/2, 251) #Complex Scaling Angle
KappaT = 1 #Sturmian Basis Parameter
DeltaT = np.array([0, 0, -1/4, 1/4]) #Quantum Defect Parameter
E0T = np.linspace(-1/4, 1/4, 2) #Field Strength
NT = 150 #Sturmian Basis Length

'Initial Energies, Theta Dependence'
EInitialApprox = np.zeros((len(E0T), len(DeltaT)), dtype=complex)
for i in range(len(E0T)):
    EInitialApprox[i, 0] = HypApproxN1C(E0T[i])
    EInitialApprox[i, 1] = HypApproxN2C(E0T[i])
    EInitialApprox[i, 2] = HypApproxN1K(DeltaT[2], E0T[i])
    EInitialApprox[i, 3] = HypApproxN1K(DeltaT[3], E0T[i])

NInitialEnergyT = 200
ThetaInitialEnergyT = 0
InitialEnergyT = np.zeros((len(E0T), len(DeltaT)), dtype=complex)
InitialHT = np.zeros((NInitialEnergyT, NInitialEnergyT), dtype=complex)
InitialKT = np.zeros((NInitialEnergyT, NInitialEnergyT), dtype=complex)
for i in range(len(E0T)):
    print(i*100/len(E0T))
    for j in range(len(DeltaT)):
        for k in range(NInitialEnergyT):
            for l in range(NInitialEnergyT):
                    InitialHT[k, l] = complex(Hnm(k+1, l+1, DeltaT[j], KappaT, ThetaInitialEnergyT, E0T[i]))
                    InitialKT[k, l] = complex(Knm(k+1, l+1, DeltaT[j], KappaT))
        E, C = eig(InitialHT[:, :], InitialKT[:, :])
        InitialMinIndex = np.argmin(np.abs(np.real(EInitialApprox[i, j] - E)))
        InitialEnergyT[i, j] = E[InitialMinIndex]

'Storage & Eigensolving, Theta Dependence'
HT = np.zeros((NT, NT, len(ThetaT), len(DeltaT), len(E0T)), dtype=complex)
KT = np.zeros((NT, NT, len(ThetaT), len(DeltaT), len(E0T)), dtype=complex)
EStorageT = np.zeros((NT, len(ThetaT), len(DeltaT), len(E0T)), dtype=complex)

for k in range(len(ThetaT)):
    print(k*100/len(ThetaT))
    for l in range(len(DeltaT)):
        for i in range(NT):
            for j in range(NT):
                HT[i, j, k, l, 0] = complex(Hnm(i+1, j+1, DeltaT[l], KappaT, ThetaT[k], E0T[0]))
                KT[i, j, k, l, 0] = complex(Knm(i+1, j+1, DeltaT[l], KappaT))
```

```python
                HT[i, j, k, l, 1] = complex(Hnm(i+1, j+1, DeltaT[l], KappaT, ThetaT[k], E0T[1]))
                KT[i, j, k, l, 1] = complex(Knm(i+1, j+1, DeltaT[l], KappaT))
        E, C = eig(HT[:, :, k, l, 0], KT[:, :, k, l, 0])
        EStorageT[:, k, l, 0] = E
        E, C = eig(HT[:, :, k, l, 1], KT[:, :, k, l, 1])
        EStorageT[:, k, l, 1] = E

ENumericalT = np.zeros((len(ThetaT), len(DeltaT), len(E0T)), dtype=complex)
for i in range(len(DeltaT)):
        for j in range(len(E0T)):
            ENumericalT[int(len(ThetaT)/2), i, j] = InitialEnergyT[j, i]

for i in range(len(DeltaT)):
        for j in range(len(E0T)):
            for k in range(int(len(ThetaT)/2)-1):
                MinIndex1 = np.argmin(np.abs(np.real(EStorageT[:, int(len(ThetaT)/2) + k + 1, i, j] -
                      ENumericalT[int(len(ThetaT)/2) + k, i, j])))
                ENumericalT[int(len(ThetaT)/2) + k + 1, i, j] = EStorageT[MinIndex1, int(len(ThetaT)/2) + k + 1, i, j]
                MinIndex2 = np.argmin(np.abs(np.real(EStorageT[:, int(len(ThetaT)/2) - k - 1, i, j] -
                      ENumericalT[int(len(ThetaT)/2) - k, i, j])))
                ENumericalT[int(len(ThetaT)/2) - k - 1, i, j] = EStorageT[MinIndex2, int(len(ThetaT)/2) - k - 1, i, j]

'Plotting, Theta Dependence'
fig, ax1 = plt.subplots(figsize=(7.5,5))
ax1.set_xlabel(r"$\mathrm{Complex} \ \mathrm{Scaling} \ \mathrm{Angle}, \ \theta$", fontsize=16)
ax1.set_ylabel(r"$\mathrm{Perturbed} \ \mathrm{Energy}, \ \tilde{E}_n$", fontsize=16)
ax1.set_xlim([ThetaT[0], ThetaT[-1]])
#ax3.set_ylim([-5, 5])
ax1.tick_params(axis='x', labelsize=12)
ax1.tick_params(axis='y', labelsize=12)
ax1.grid('true')
ax1.plot(ThetaT, np.real(ENumericalT[:, 0, 0]), linestyle='dashed', color='red')
ax1.plot(ThetaT, np.real(ENumericalT[:, 0, 1]), label=r"$n=1$", color='red')
ax1.plot(ThetaT, np.real(ENumericalT[:, 1, 0]), linestyle='dashed', color='green')
ax1.plot(ThetaT, np.real(ENumericalT[:, 1, 1]), label=r"$n=2$", color='green')
ax1.plot(ThetaT, np.real(ENumericalT[:, 2, 0]), linestyle='dashed', color='blue')
ax1.plot(ThetaT, np.real(ENumericalT[:, 2, 1]), label=r"$n=1, \ \delta = -1/4$", color='blue')
ax1.plot(ThetaT, np.real(ENumericalT[:, 3, 0]), linestyle='dashed', color='orange')
ax1.plot(ThetaT, np.real(ENumericalT[:, 3, 1]), label=r"$n=1, \ \delta = 1/4$", color='orange')
ax1.legend(fontsize=13, ncol=1)
#plt.savefig("PerturbedEnergyThetaDependence.pdf", dpi=300)

'-------------------- Delta Dependence --------------------'

'Parameters, Delta Dependence'
ThetaD = 0.2 #Complex Scaling Angle
KappaD = 1 #Sturmian Basis Parameter
DeltaD = np.linspace(-0.8, 0.7, 200) #Quantum Defect Parameter
E0D = np.array([-0.75, -0.25, 0, 0.25, 0.75]) #Field Strength
ND = 200 #Sturmian Basis Length

'2F1 Approximants, Delta Dependence'
EApproxD = np.zeros((len(DeltaD), len(E0D)), dtype=complex)
for k in range(len(DeltaD)):
    for l in range(len(E0D)):
        EApproxD[k, l] = HypApproxN1K(DeltaD[k], E0D[l])

'Storage & Eigensolving, Field Strength Dependence'
HD = np.zeros((ND, ND, len(DeltaD), len(E0D)), dtype=complex)
KD = np.zeros((ND, ND, len(DeltaD), len(E0D)), dtype=complex)
EStorageD = np.zeros((ND, len(DeltaD), len(E0D)), dtype=complex)

for k in range(len(DeltaD)):
    print(k*100/len(DeltaD))
    for l in range(len(E0D)):
        for i in range(ND):
            for j in range(ND):
                HD[i, j, k, l] = complex(Hnm(i+1, j+1, DeltaD[k], KappaD, ThetaD, E0D[l]))
                KD[i, j, k, l] = complex(Knm(i+1, j+1, DeltaD[k], KappaD))
        E, C = eig(HD[:, :, k, l], KD[:, :, k, l])
        EStorageD[:, k, l] = np.sort(E)

ENumericalD = np.zeros((len(DeltaD), len(E0D)), dtype=complex)
for k in range(len(DeltaD)):
    for l in range(len(E0D)):
        MinIndex = np.argmin(np.abs(np.real(EApproxD[k, l] - EStorageD[:, k, l])))
        ENumericalD[k, l] = EStorageD[MinIndex, k, l]

'Plotting Energy, Delta Dependence'
fig, ax2 = plt.subplots(figsize=(7.5,5))
ax2.set_xlabel(r"$\mathrm{Quantum} \ \mathrm{Defect} \ \mathrm{Parameter}, \ \delta$", fontsize=16)
ax2.set_ylabel(r"$\mathrm{Perturbed} \ \mathrm{Energy}, \ \tilde{E}_n$", fontsize=16)
ax2.set_xlim([DeltaD[0], DeltaD[-1]])
ax2.set_ylim([-2, 1.5])
ax2.tick_params(axis='x', labelsize=12)
ax2.tick_params(axis='y', labelsize=12)
ax2.grid('true')
ax2.plot(DeltaD, np.real(EApproxD[:, 0]), label=r"$n=1, \ \mathcal{E}_0 = -3/4$", color='red')
ax2.scatter(DeltaD[::5], np.real(ENumericalD[::5, 0]), edgecolors='red', facecolors='none')
ax2.plot(DeltaD, np.real(EApproxD[:, 1]), label=r"$n=1, \ \mathcal{E}_0 = -1/4$", color='green')
ax2.scatter(DeltaD[::5], np.real(ENumericalD[::5, 1]), edgecolors='green', facecolors='none')
ax2.plot(DeltaD, np.real(EApproxD[:, 2]), label=r"$n=1, \ \mathcal{E}_0 = 0$", color='blue')
ax2.scatter(DeltaD[::5], np.real(ENumericalD[::5, 2]), edgecolors='blue', facecolors='none')
ax2.plot(DeltaD, np.real(EApproxD[:, 3]), label=r"$n=1, \ \mathcal{E}_0 = 1/4$", color='orange')
ax2.scatter(DeltaD[::5], np.real(ENumericalD[::5, 3]), edgecolors='orange', facecolors='none')
ax2.plot(DeltaD, np.real(EApproxD[:, 4]), label=r"$n=1, \ \mathcal{E}_0 = 3/4$", color='grey')
ax2.scatter(DeltaD[::5], np.real(ENumericalD[::5, 4]), edgecolors='grey', facecolors='none')
ax2.legend(fontsize=13, ncol=1)
#plt.savefig("PerturbedEnergyPositiveFieldDependence.pdf", dpi=300)

'-------------------- Positive Field Strength Dependence --------------------'

'Parameters, Positive Field Strength Dependence'
ThetaEPlus = 0.2 #Complex Scaling Angle
KappaEPlus = 1 #Sturmian Basis Parameter
DeltaEPlus = np.array([0, 0, 0, 0, 0, -0.25, 0.25]) #Quantum Defect Parameter
E0EPlus = np.linspace(0, 1, 100) #Field Strength
NEPlus = 200 #Sturmian Basis Length

'2F1 Approximants, Positive Field Strength Dependence'
EApproxEPlus = np.zeros((len(E0EPlus), len(DeltaEPlus)), dtype=complex)
for k in range(len(E0EPlus)):
    EApproxEPlus[k, 0] = HypApproxN1C(E0EPlus[k])
    EApproxEPlus[k, 1] = HypApproxN2C(E0EPlus[k])
    EApproxEPlus[k, 2] = HypApproxN3C(E0EPlus[k])
    EApproxEPlus[k, 3] = HypApproxN4C(E0EPlus[k])
    EApproxEPlus[k, 4] = HypApproxN5C(E0EPlus[k])
    EApproxEPlus[k, 5] = HypApproxN1K(DeltaEPlus[-2], E0EPlus[k])
    EApproxEPlus[k, 6] = HypApproxN1K(DeltaEPlus[-1], E0EPlus[k])

'Storage & Eigensolving, Positive Field Strength Dependence'
HEPlus = np.zeros((NEPlus, NEPlus, len(E0EPlus), len(set(DeltaEPlus))), dtype=complex)
KEPlus = np.zeros((NEPlus, NEPlus, len(E0EPlus), len(set(DeltaEPlus))), dtype=complex)
EStorageEPlus = np.zeros((NEPlus, len(E0EPlus), len(set(DeltaEPlus))), dtype=complex)

for k in range(len(E0EPlus)):
    print(k*100/len(E0EPlus))
    for i in range(NEPlus):
        for j in range(NEPlus):
                HEPlus[i, j, k, 0] = complex(Hnm(i+1, j+1, DeltaEPlus[0], KappaEPlus, ThetaEPlus, E0EPlus[k]))
                KEPlus[i, j, k, 0] = complex(Knm(i+1, j+1, DeltaEPlus[0], KappaEPlus))
                HEPlus[i, j, k, 1] = complex(Hnm(i+1, j+1, DeltaEPlus[-2], KappaEPlus, ThetaEPlus, E0EPlus[k]))
                KEPlus[i, j, k, 1] = complex(Knm(i+1, j+1, DeltaEPlus[-2], KappaEPlus))
                HEPlus[i, j, k, 2] = complex(Hnm(i+1, j+1, DeltaEPlus[-1], KappaEPlus, ThetaEPlus, E0EPlus[k]))
                KEPlus[i, j, k, 2] = complex(Knm(i+1, j+1, DeltaEPlus[-1], KappaEPlus))
    E, C = eig(HEPlus[:, :, k, 0], KEPlus[:, :, k, 0])
    EStorageEPlus[:, k, 0] = np.sort(E)
    E, C = eig(HEPlus[:, :, k, 1], KEPlus[:, :, k, 1])
    EStorageEPlus[:, k, 1] = np.sort(E)
    E, C = eig(HEPlus[:, :, k, 2], KEPlus[:, :, k, 2])
    EStorageEPlus[:, k, 2] = np.sort(E)

ENumericalEPlus = np.zeros((len(E0EPlus), len(DeltaEPlus)), dtype=complex)
for k in range(len(E0EPlus)):
    MinIndex1 = np.argmin(np.abs(np.real(EApproxEPlus[k, 0] - EStorageEPlus[:, k, 0])))
    ENumericalEPlus[k, 0] = EStorageEPlus[MinIndex1, k, 0]
    MinIndex2 = np.argmin(np.abs(np.real(EApproxEPlus[k, 1] - EStorageEPlus[:, k, 0])))
    ENumericalEPlus[k, 1] = EStorageEPlus[MinIndex2, k, 0]
    MinIndex3 = np.argmin(np.abs(np.real(EApproxEPlus[k, 2] - EStorageEPlus[:, k, 0])))
    ENumericalEPlus[k, 2] = EStorageEPlus[MinIndex3, k, 0]
    MinIndex4 = np.argmin(np.abs(np.real(EApproxEPlus[k, 3] - EStorageEPlus[:, k, 0])))
    ENumericalEPlus[k, 3] = EStorageEPlus[MinIndex4, k, 0]
    MinIndex5 = np.argmin(np.abs(np.real(EApproxEPlus[k, 4] - EStorageEPlus[:, k, 0])))
    ENumericalEPlus[k, 4] = EStorageEPlus[MinIndex5, k, 0]
    MinIndex6 = np.argmin(np.abs(np.real(EApproxEPlus[k, 5] - EStorageEPlus[:, k, 1])))
    ENumericalEPlus[k, 5] = EStorageEPlus[MinIndex6, k, 1]
    MinIndex7 = np.argmin(np.abs(np.real(EApproxEPlus[k, 6] - EStorageEPlus[:, k, 2])))
    ENumericalEPlus[k, 6] = EStorageEPlus[MinIndex6, k, 2]

'Plotting, Positive Field Strength Dependence'
fig, ax3 = plt.subplots(figsize=(7.5,5))
ax3.set_xlabel(r"$\mathrm{Field} \ \mathrm{Strength}, \ \mathcal{E}_0$", fontsize=16)
ax3.set_ylabel(r"$\mathrm{Perturbed} \ \mathrm{Energy}, \ \tilde{E}_n$", fontsize=16)
ax3.set_xlim([E0EPlus[0], E0EPlus[-1]])
#ax3.set_ylim([-5, 5])
ax3.tick_params(axis='x', labelsize=12)
ax3.tick_params(axis='y', labelsize=12)
ax3.grid('true')
ax3.plot(E0EPlus, np.real(EApproxEPlus[:, 0]), label=r"$n = 1$", color='red')
ax3.scatter(E0EPlus[::6], np.real(ENumericalEPlus[::6, 0]), edgecolors='red', facecolors='none')
ax3.plot(E0EPlus, np.real(EApproxEPlus[:, 1]), label=r"$n = 2$", color='green')
ax3.scatter(E0EPlus[::6], np.real(ENumericalEPlus[::6, 1]), edgecolors='green', facecolors='none')
ax3.plot(E0EPlus, np.real(EApproxEPlus[:, 2]), label=r"$n = 3$", color='blue')
ax3.scatter(E0EPlus[::6], np.real(ENumericalEPlus[::6, 2]), edgecolors='blue', facecolors='none')
ax3.plot(E0EPlus, np.real(EApproxEPlus[:, 3]), label=r"$n = 4$", color='orange')
ax3.scatter(E0EPlus[::6], np.real(ENumericalEPlus[::6, 3]), edgecolors='orange', facecolors='none')
ax3.plot(E0EPlus, np.real(EApproxEPlus[:, 4]), label=r"$n = 5$", color='grey')
ax3.scatter(E0EPlus[::6], np.real(ENumericalEPlus[::6, 4]), edgecolors='grey', facecolors='none')
ax3.plot(E0EPlus, np.real(EApproxEPlus[:, 5]), label=r"$n=1, \ \delta = -1/4$", color='pink')
ax3.scatter(E0EPlus[::6], np.real(ENumericalEPlus[::6, 5]), edgecolors='pink', facecolors='none')
ax3.plot(E0EPlus, np.real(EApproxEPlus[:, 6]), label=r"$n=1, \ \delta = 1/4$", color='black')
ax3.scatter(E0EPlus[::6], np.real(ENumericalEPlus[::6, 6]), edgecolors='black', facecolors='none')
ax3.legend(fontsize=13, ncol=2)
#plt.savefig("PerturbedEnergyPositiveFieldDependence.pdf", dpi=300)

'-------------------- Field Strength Dependence --------------------'

'Parameters, Field Strength Dependence'
ThetaE = 0.2 #Complex Scaling Angle
KappaE = 1 #Sturmian Basis Parameter
DeltaE = np.array([0, 0, -0.25, 0.25]) #Quantum Defect Parameter
E0E = np.linspace(-1, 1, 200) #Field Strength
NE = 150 #Sturmian Basis Length

'2F1 Approximants, Field Strength Dependence'
EApproxE = np.zeros((len(E0E), len(DeltaE)), dtype=complex)
for k in range(len(E0E)):
    EApproxE[k, 0] = HypApproxN1C(E0E[k])
    EApproxE[k, 1] = HypApproxN2C(E0E[k])
    EApproxE[k, 2] = HypApproxN1K(DeltaE[-2], E0E[k])
    EApproxE[k, 3] = HypApproxN1K(DeltaE[-1], E0E[k])

'Perturbative Expansions, Field Strength Dependence'
EExpansionE = np.zeros((len(E0E), len(DeltaE)), dtype=complex)
for k in range(len(E0E)):
    EExpansionE[k, 0] = EExpansionN1C(E0E[k])
    EExpansionE[k, 1] = EExpansionN2C(E0E[k])
    EExpansionE[k, 2] = EExpansionN1K(DeltaE[-2], E0E[k])
    EExpansionE[k, 3] = EExpansionN1K(DeltaE[-1], E0E[k])

'Storage & Eigensolving, Field Strength Dependence'
HE = np.zeros((NE, NE, len(E0E), len(set(DeltaE))), dtype=complex)
KE = np.zeros((NE, NE, len(E0E), len(set(DeltaE))), dtype=complex)
EStorageE = np.zeros((NE, len(E0E), len(set(DeltaE))), dtype=complex)

for k in range(len(E0E)):
    print(k*100/len(E0E))
    for i in range(NE):
        for j in range(NE):
              HE[i, j, k, 0] = complex(Hnm(i+1, j+1, DeltaE[0], KappaE, ThetaE, E0E[k]))
              KE[i, j, k, 0] = complex(Knm(i+1, j+1, DeltaE[0], KappaE))
              HE[i, j, k, 1] = complex(Hnm(i+1, j+1, DeltaE[-2], KappaE, ThetaE, E0E[k]))
              KE[i, j, k, 1] = complex(Knm(i+1, j+1, DeltaE[-2], KappaE))
              HE[i, j, k, 2] = complex(Hnm(i+1, j+1, DeltaE[-1], KappaE, ThetaE, E0E[k]))
              KE[i, j, k, 2] = complex(Knm(i+1, j+1, DeltaE[-1], KappaE))
    E, C = eig(HE[:, :, k, 0], KE[:, :, k, 0])
    EStorageE[:, k, 0] = np.sort(E)
    E, C = eig(HE[:, :, k, 1], KE[:, :, k, 1])
    EStorageE[:, k, 1] = np.sort(E)
    E, C = eig(HE[:, :, k, 2], KE[:, :, k, 2])
    EStorageE[:, k, 2] = np.sort(E)

ENumericalE = np.zeros((len(E0E), len(DeltaE)), dtype=complex)
for k in range(len(E0E)):
    MinIndex1 = np.argmin(np.abs(np.real(EApproxE[k, 0] - EStorageE[:, k, 0])))
    ENumericalE[k, 0] = EStorageE[MinIndex1, k, 0]
    MinIndex2 = np.argmin(np.abs(np.real(EApproxE[k, 1] - EStorageE[:, k, 0])))
    ENumericalE[k, 1] = EStorageE[MinIndex2, k, 0]
    MinIndex3 = np.argmin(np.abs(np.real(EApproxE[k, 2] - EStorageE[:, k, 1])))
    ENumericalE[k, 2] = EStorageE[MinIndex3, k, 1]
    MinIndex4 = np.argmin(np.abs(np.real(EApproxE[k, 3] - EStorageE[:, k, 2])))
    ENumericalE[k, 3] = EStorageE[MinIndex4, k, 2]
ENumericalE[80, 1] = EStorageE[111, 80, 0]

'Plotting Energy, Field Strength Dependence'
fig, ax4 = plt.subplots(figsize=(7.5,5))
ax4.set_xlabel(r"$\mathrm{Field} \ \mathrm{Strength}, \ \mathcal{E}_0$", fontsize=16)
ax4.set_ylabel(r"$\mathrm{Perturbed} \ \mathrm{Energy}, \ \tilde{E}_n$", fontsize=16)
ax4.set_xlim([E0E[0], E0E[-1]])
ax4.set_ylim([-2, 1])
ax4.tick_params(axis='x', labelsize=12)
ax4.tick_params(axis='y', labelsize=12)
ax4.grid('true')
ax4.plot(E0E, np.real(EApproxE[:, 3]), label=r"$n=1, \ \delta = 1/4$", color='orange')
ax4.plot(E0E, np.real(EExpansionE[:, 3]), color='orange', linestyle='dashed')
ax4.scatter(E0E[::5], np.real(ENumericalE[::5, 3]), edgecolors='orange', facecolors='none')
ax4.plot(E0E, np.real(EApproxE[:, 2]), label=r"$n=1, \ \delta = -1/4$", color='blue')
ax4.plot(E0E, np.real(EExpansionE[:, 2]), color='blue', linestyle='dashed')
ax4.scatter(E0E[::5], np.real(ENumericalE[::5, 2]), edgecolors='blue', facecolors='none')
ax4.plot(E0E, np.real(EApproxE[:, 0]), label=r"$n = 1$", color='red')
ax4.plot(E0E, np.real(EExpansionE[:, 0]), color='red', linestyle='dashed')
ax4.scatter(E0E[::5], np.real(ENumericalE[::5, 0]), edgecolors='red', facecolors='none')
ax4.plot(E0E, np.real(EApproxE[:, 1]), label=r"$n = 2$", color='green')
ax4.plot(E0E, np.real(EExpansionE[:, 1]), color='green', linestyle='dashed')
```

```
ax4.scatter(E0E[::5], np.real(ENumericalE[::5, 1]), edgecolors='green', facecolors='none')
ax4.legend(fontsize=13, ncol=1)
#plt.savefig("PerturbedEnergyPositiveFieldDependence.pdf", dpi=300)

'Plotting Ionization Rate, Field Strength Dependence'
fig, ax5 = plt.subplots(figsize=(7.5,5))
ax5.set_yscale("log")
ax5.set_xlabel(r"$\mathrm{Field} \ \mathrm{Strength}, \ \mathcal{E}_0$", fontsize=16)
ax5.set_ylabel(r"$\mathrm{Ionization} \ \mathrm{Rate}, \ \Gamma_n$", fontsize=16)
ax5.set_xlim([E0E[0], 0.1])
ax5.set_ylim([1e-3, 1e1])
ax5.tick_params(axis='x', labelsize=12)
ax5.tick_params(axis='y', labelsize=12)
ax5.grid('true')
ax5.plot(E0E, 2*np.imag(EApproxE[:, 2]), label=r"$n=1, \ \delta = -1/4$", color='blue')
ax5.scatter(E0E[::5], -2*np.imag(ENumericalE[::5, 2]), edgecolors='blue', facecolors='none')
ax5.plot(E0E, 2*np.imag(EApproxE[:, 3]), label=r"$n=1, \ \delta = 1/4$", color='orange')
ax5.scatter(E0E[::5], -2*np.imag(ENumericalE[::5, 3]), edgecolors='orange', facecolors='none')
ax5.plot(E0E, 2*np.imag(EApproxE[:, 0]), label=r"$n = 1$", color='red')
ax5.scatter(E0E[::5], -2*np.imag(ENumericalE[::5, 0]), edgecolors='red', facecolors='none')
ax5.plot(E0E, 2*np.imag(EApproxE[:, 1]), label=r"$n = 2$", color='green')
ax5.scatter(E0E[::5], -2*np.imag(ENumericalE[::5, 1]), edgecolors='green', facecolors='none')
ax5.legend(fontsize=13, ncol=1)
plt.savefig("PerturbedEnergyPositiveFieldDependence.pdf", dpi=300)

'-------------------- Frequency vs Voltage --------------------'

'Parameters, Frequency vs Voltage'
EngToFreq = 318.2 #Energy Frequency Conversion
FieldUnit = 0.16 #Electric Field Unit, Volts/micrometers
VMax = 60 #Max Voltage Difference, Volts
rc = 3200 #Cell Height, Micrometers
EMax = (VMax / rc) #Max Field Strength, Atomic Units
DeltaF = 0.01730692276910764

'Data Import, Frequency vs Voltage'
file_path = r'C:\Users\jonat\Desktop\Universitet\Kandidat\Speciale r\Speciale\Figures &
      Code\FrequencyVsVolts.xlsx' # Replace with your file path
data = pd.read_excel(file_path)
VoltData12 = np.array(data["Volts12"].tolist())
VoltData13 = np.array(data["Volts13"].tolist())[:10]
FreqData12 = np.array(data["GHz12"].tolist())
FreqData13 = np.array(data["GHz13"].tolist())[:10]

'Delta Optimization, Frequency vs Voltage'
def f12(delta, E):
    T1 = np.real(HypApproxN2K(delta, E / (3200 * FieldUnit)) - HypApproxN1K(delta, E / (3200 *
          FieldUnit)))*EngToFreq
    return T1
def f13(delta, E):
    T1 = np.real(HypApproxN3K(delta, E / (3200 * FieldUnit)) - HypApproxN1K(delta, E / (3200 *
          FieldUnit)))*EngToFreq
    return T1

DeltaLin = np.linspace(-0.05, 0.05, 2500)
Error = np.zeros(len(DeltaLin))
for i in range(len(DeltaLin)):
    print(i)
    X12 = np.zeros(len(FreqData12))
    for j in range(len(FreqData12)):
        X12[j] = (f12(DeltaLin[i], VoltData12[j]) - FreqData12[j])**2
    X13 = np.zeros(len(FreqData13))
    for j in range(len(FreqData13)):
        X13[j] = (f13(DeltaLin[i], VoltData13[j]) - FreqData13[j])**2
    Error[i] = np.sum(X12) + np.sum(X13)
DeltaF = DeltaLin[np.argmin(Error)]
```

```
'2F1 Approximants, Frequency vs Voltage'
E0V = np.linspace(0, EMax, 1000)
f0V = np.zeros((6, len(E0V)))
for j in range(len(E0V)):
   f0V[0, j] = np.real(HypApproxN2K(0, E0V[j]/FieldUnit) - HypApproxN1K(0, E0V[j]/FieldUnit))*EngToFreq
   f0V[1, j] = np.real(HypApproxN2K(DeltaF, E0V[j]/FieldUnit) - HypApproxN1K(DeltaF, E0V[j]/FieldUnit))*EngToFreq
   f0V[2, j] = np.real(HypApproxN3K(0, E0V[j]/FieldUnit) - HypApproxN1K(0, E0V[j]/FieldUnit))*EngToFreq
   f0V[3, j] = np.real(HypApproxN3K(DeltaF, E0V[j]/FieldUnit) - HypApproxN1K(DeltaF, E0V[j]/FieldUnit))*EngToFreq
   f0V[4, j] = np.real(HypApproxN2K(0.0237, E0V[j]/FieldUnit) - HypApproxN1K(0.0237, E0V[j]/FieldUnit))*EngToFreq
   f0V[5, j] = np.real(HypApproxN3K(0.0237, E0V[j]/FieldUnit) - HypApproxN1K(0.0237, E0V[j]/FieldUnit))*EngToFreq

'Plotting, Frequency vs Voltage'
fig, ax6 = plt.subplots(figsize=(7.5,5))
ax6.set_xlabel(r"$\mathrm{Potential} \ \mathrm{Difference}, \ \Delta V \ \mathrm{[V]}$", fontsize=16)
ax6.set_ylabel(r"$\mathrm{Transition} \ \mathrm{Frequency}, \ f_{n \rightarrow m} \ \mathrm{[GHz]}$", fontsize=16)
ax6.tick_params(axis='x', labelsize=12)
ax6.tick_params(axis='y', labelsize=12)
ax6.grid('true')
ax6.scatter(VoltData12, FreqData12, edgecolors='red', facecolors='none', label=r"$f_{1 \rightarrow 2} \ \mathrm{Exp.} \ \mathrm{Data}$")
ax6.scatter(VoltData13, FreqData13, edgecolors='green', facecolors='none', label=r"$f_{1 \rightarrow 3} \ \mathrm{Exp.} \ \mathrm{Data}$")
ax6.plot(E0V*rc, f0V[0, :], color='red', linestyle='dashed', label=r"$\mathrm{Coulomb} \ {}_2 F_1, \ \delta = 0$")
ax6.plot(E0V*rc, f0V[1, :], color='red', label=r"$\mathrm{Kratzer} \ {}_2 F_1, \ \delta = 0.0173$")
ax6.plot(E0V*rc, f0V[2, :], color='green', linestyle='dashed', label=r"$\mathrm{Coulomb} \ {}_2 F_1, \ \delta = 0$")
ax6.plot(E0V*rc, f0V[3, :], color='green', label=r"$\mathrm{Kratzer} \ {}_2 F_1, \ \delta = 0.0173$")
ax6.set_xlim([0, 40])
ax6.set_ylim([119, 225])
ax6.legend(fontsize=13, ncol=1)
plt.savefig("TransFreqVsVoltage.pdf", dpi=300)
```

Appendix F

Linear Dynamic Polarizability Code

The following code is used to compute and plot the linear dynamic polarizability for the ground state and first excited state for both the Coulomb and Kratzer potential. It also calculates the dynamic Pockels polarizability for the Coulomb ground state. The code is written for Python.

Listing F.1: Linear Dynamic Polarizability Calculator

```python
import numpy as np
import sympy as sp
import mpmath as mp
import matplotlib.pyplot as plt
from matplotlib.lines import Line2D

'Latex Font'
plt.rcParams['text.usetex'] = True # Enable LaTeX rendering
plt.rcParams['font.family'] = 'serif' # Use a serif font
plt.rcParams['font.serif'] = ['Computer Modern'] # LaTeX's default serif font
plt.rcParams['text.latex.preamble'] = r'\usepackage{amsmath}' # Optional: for more LaTeX symbols

'Functions'
def Poch(a, b):
    return mp.gamma(a+b)/mp.gamma(a)

'Analytical Alpha'
def AlphaK1(delta, omega):
    def AlphaK1Plus(delta, omega):
        E1 = - 1 / (2*(1-delta)**2)
        kappa = mp.sqrt(-2 * (E1 + omega))
        Z = ((delta - 1)*kappa + 1)**2 / ((delta - 1)*kappa - 1)**2
        term1 = (1/(((delta - 1)*kappa + 1)**5*(-(delta*kappa) + kappa + 1)**8))*2**(4 - 4*delta)*(1 - delta)**(2*(delta + 3))*kappa**(1 - 2*delta)*(1/(1 - delta) + kappa)**(4*delta)*((delta - 1)*kappa - 1)**3*(((delta - 1)*kappa + 1)**2/(-(delta*kappa) + kappa + 1)**2)**(delta + 1/kappa)
        term2 = ((delta - 1)*kappa - 1)**3*(2*delta**2 + (delta - 2)*(delta - 1)**2*(2*delta - 3)*kappa**2 + 2*(delta - 2)*(delta - 1)*(2*delta - 3)*kappa - 7*delta + 4)*sp.betainc(-delta - 1/kappa + 1, 2*delta - 1, 0, Z)
        term3 = - 2*(delta - 1)*kappa*((delta - 2)*(delta - 1)**2*kappa**2 - delta)*((delta - 1)**2*(2*delta - 3)*kappa**2 - 2*delta + 7)*sp.betainc(-delta - 1/kappa + 1, 2*delta - 2, 0, Z)
        return sp.N(term1 * (term2 + term3))
    return AlphaK1Plus(delta, omega)+AlphaK1Plus(delta, -omega)
def AlphaK2(delta, omega):
    def AlphaK2Plus(delta, omega):
        E2 = - 1 / (2*(2-delta)**2)
        kappa = mp.sqrt(-2 * (E2 + omega))
        Z = ((delta - 2)*kappa + 1)**2 / ((delta - 2)*kappa - 1)**2
        term1 = (- (2**(7 - 4*delta)*(2 - delta)**(2*(delta + 2))*(delta - 1)*(2*delta - 3)**2 *(kappa + 1/(2 - delta))**(4*delta)*((delta - 2)*kappa + 1)**6 *(Z)**(delta + 1/kappa)*(delta + (delta - 2)*(delta - 1)*kappa - 3)**2 *sp.betainc(-delta - 1/kappa + 1, 2*delta - 1, 0, Z) * kappa**(1 - 2*delta)) /((delta - 2)**2*kappa**2 - 1)**8)
        term2 = ((2**(10 - 4*delta)*(2 - delta)**(2*(delta + 2))*(delta - 2)*(2*delta - 3)**2 *(kappa + 1/(2 - delta))**(4*delta)*((delta - 2)*kappa + 1)**4 *(Z)**(delta + 1/kappa)*(2*delta + (delta - 2)*(delta - 1)*kappa - 5) *((delta - 1)*kappa**2*(delta - 2)**3 + (2*(delta - 4)*delta + 9)*kappa*(delta - 2) + delta**2 - 4*delta + 3) *sp.betainc(-delta - 1/kappa + 2, 2*(delta - 1), 0, Z) * kappa**(2 - 2*delta)) /((delta - 2)**2*kappa**2 - 1)**8)
        term3 = ((2**(10 - 4*delta)*(2 - delta)**(2*(delta + 2))*(delta - 2)**2*(2*delta - 3) *(kappa + 1/(2 - delta))**(4*delta)*((delta - 2)*kappa + 1)**2 *(Z)**(delta + 1/kappa) *((delta - 1)**2*(6*delta -
              13)*kappa**4*(delta - 2)**5 +2*(delta - 1)*(4*delta*(5*delta - 22) + 101)*kappa**3*(delta - 2)**4
              +(delta*(delta*(88*delta**2 - 718*delta + 2233) - 3144) + 1694)*kappa**2*(delta - 2)**2 +2*(2*delta -
              5)*(2*delta - 3)*(delta*(10*delta - 43) + 48)*kappa*(delta - 2) +26*delta**4 - 205*delta**3 +
              587*delta**2 - 723*delta + 324) *sp.betainc(-delta - 1/kappa + 3, 2*delta - 3, 0, Z) * kappa**(3 -
              2*delta)) /((delta - 2)**2*kappa**2 - 1)**8)
        term4 = ((2**(12 - 4*delta)*(2 - delta)**(2*(delta + 2))*(delta - 2)**5*(2*delta - 3) *(kappa + 1/(2 -
              delta))**(4*delta)*(Z)**(delta + 1/kappa) *((delta - 2)**4*(delta - 1)**2*(2*delta - 5)*kappa**5
              +(delta - 2)**3*(delta - 1)*(delta*(22*delta - 109) + 138)*kappa**4 +2*(delta - 2)**2*(2*delta -
              5)*(delta*(19*delta - 82) + 95)*kappa**3 +2*(delta - 2)*(delta*(delta*(58*delta - 407) + 964) -
              768)*kappa**2 +(2*delta - 5)*(delta*(41*delta - 174) + 181)*kappa +22*delta**2 - 91*delta + 87)
              *sp.betainc(-delta - 1/kappa + 4, 2*(delta - 2), 0, Z) * kappa**(4 - 2*delta)) /((delta -
              2)**2*kappa**2 - 1)**8)
        term5 = ((2**(11 - 4*delta)*(2 - delta)**(2*(delta + 2))*(delta - 2)**5*(2*delta - 5)*(2*delta - 3) *(kappa
              + 1/(2 - delta))**(4*delta)*(Z)**(delta + 1/kappa) *((delta - 2)**4*(delta - 1)**2*kappa**4 +20*(delta
              - 3)*(delta - 2)**3*(delta - 1)*kappa**3 +2*(delta - 2)**2*(delta*(39*delta - 212) + 305)*kappa**2
              +20*(delta - 3)*(delta - 2)*(5*delta - 11)*kappa +41*delta**2 - 174*delta + 181) *sp.betainc(-delta -
              1/kappa + 5, 2*delta - 5, 0, Z) * kappa**(5 - 2*delta)) /((delta - 2)**2*kappa**2 - 1)**8)
        term6 = ((2**(14 - 4*delta)*(2 - delta)**(2*(delta + 2))*(delta - 3)*(delta - 2)**6*(2*delta - 5)*(2*delta -
              3) *(kappa + 1/(2 - delta))**(4*delta)*(Z)**(delta + 1/kappa) *((delta - 2)**2*(delta - 1)*kappa**2 +
              3*(delta - 2)*(2*delta - 7)*kappa + 5*delta - 11) *sp.betainc(-delta - 1/kappa + 6, 2*(delta - 3), 0,
              Z) * kappa**(6 - 2*delta)) /((delta - 2)**2*kappa**2 - 1)**8)
        term7 = ((2**(14 - 4*delta)*(2 - delta)**(2*(delta + 2))*(delta - 3)*(delta - 2)**7*(2*delta - 7)*(2*delta -
              5)*(2*delta - 3) *(kappa + 1/(2 - delta))**(4*delta)*(Z)**(delta + 1/kappa) *sp.betainc(-delta -
              1/kappa + 7, 2*delta - 7, 0, Z) * kappa**(7 - 2*delta)) /((delta - 2)**2*kappa**2 - 1)**8)
        return sp.N(term1 - term2 - term3 - term4 - term5 - term6 - term7)
    return AlphaK2Plus(delta, omega)+AlphaK2Plus(delta, -omega)
def AlphaC11(omega):
    def alphaC11Plus(omega):
        kappa = mp.sqrt(1 - 2*omega)
        term1 = 9*(31 + 2*kappa**2 - kappa**4) / (4 * (1-kappa**2)**4)
        term2 = 36*(kappa**4 - 8*kappa**2 - 1) / ((1-kappa)**5*(1+kappa)**6)
        term3 = mp.hyp2f1(1, 1, (2*kappa-1)/kappa, -(1-kappa)**2 / (4*kappa)) - 2
        return term1 + term2*term3
    return -alphaC11Plus(omega)-alphaC11Plus(-omega)
def AlphaNul(n, delta):
    scriptn = n - delta
    alpha = 1/4*scriptn**2*(7*scriptn**4 + 5*scriptn**2 - 3*delta**2*(1-delta)**2)
    return alpha
def alphan2Stat(n, delta):
    scriptn = n - delta
    F1 = -3/16 * scriptn**4
    F2 = (33*scriptn**6 + 75*scriptn**4 - 7*scriptn**2*delta**2*(1-delta)**2 + 10*delta**3*(1-delta)**3)
    return F1*F2

'Quantum Defect Energies'
def En(n, delta):
    scriptn = n - delta
    en = - 1 / (2*scriptn**(2))
    return en
def Ek(k):
    Ek = k**(2) / 2
    return Ek

'Quantum Defect Matrix Elements'
def Znm(n, m, delta):
    if n == m:
        Znm = 3 * n**(2) / 2 - delta * (3 * n - 1/2 - delta)
    else:
        scriptn = n - delta
        scriptm = m - delta
        Factor1 = (-1)**(n) * 2**(3-2*delta) * (scriptn*scriptm)**(2-delta) * mp.sqrt(Poch(n, 1-2*delta) * Poch(m,
              1-2*delta))
        Factor2 = (scriptm + scriptn)**(-2*delta) * (scriptm - scriptn)**(4) * mp.gamma(2-2*delta)
        Factor3 = ((scriptm - scriptn)/(scriptm + scriptn))**(n + m)
        Factor4 = scriptm * (scriptn - scriptm) / (scriptn + scriptm) * mp.hyp2f1(1-n, 1-m, 2-2*delta,
              -4*scriptn*scriptm / (scriptn - scriptm)**(2))
        Factor5 = (m-1) * mp.hyp2f1(1-n, 2-m, 2-2*delta, -4*scriptn*scriptm / (scriptn - scriptm)**(2))
        Znm = Factor1 / Factor2 * Factor3 * (Factor4 + Factor5)
    return Znm
def Zkm(k, m, delta):
    scriptm = m - delta
    N = (2*k)**(1-delta) * mp.fabs(mp.gamma(1-delta-1j/k)) / (mp.sqrt(2*mp.pi) * mp.exp(-mp.pi/(2*k)) * mp.gamma(2-2*delta))
    Factor1 = (-1)**(m) * (2*scriptm)**(2-delta) * N * mp.sqrt(Poch(m, 1-2*delta))
    Factor2 = (1 + 1j*k*scriptm)**(4-2*delta)
    Factor3 = ((1-1j*k*scriptm) / (1 + 1j*k*scriptm))**(-1j/k - scriptm)
    Factor4 = scriptm * (1 + 1j*k*scriptm) / (1 - 1j*k*scriptm) * mp.hyp2f1(1 - delta - 1j/k, 1 - m, 2 - 2*delta, 4j*k*scriptm / (1 + 1j*k*scriptm)**(2))
    Factor5 = (m-1) * mp.hyp2f1(1 - delta - 1j/k, 2 - m, 2 - 2*delta, 4j*k*scriptm / (1 + 1j*k*scriptm)**(2))
    Zkm = Factor1 / Factor2 * Factor3 * (Factor4 + Factor5)
    return Zkm

'Oscillator Strengths'
def gmn(m, n, delta):
    gmn = 2 * (En(n, delta) - En(m, delta)) * mp.fabs(Znm(n, m, delta))**(2)
    return gmn
def gmk(m, k, delta):
    gmk = 2 * (Ek(k) - En(m, delta)) * mp.fabs(Zkm(k, m, delta))**(2)
    return gmk

'Numerical Alpha'
def AlphaNUM(m, delta, omega):
    AlphaBound = sum(gmn(m, n, delta)/((En(n, delta) - En(m, delta))**2 - omega**2) for n in range(1, 5000) if n != m)
    AlphaFree = mp.quad(lambda k: gmk(m, k, delta)/((Ek(k) - En(m, delta))**2 - omega**2), [1e-6, mp.inf])
    return AlphaBound + AlphaFree

'Initial Calculations, Ground State'
N = 1000 #Resolution
M = 20 #Numerical Spacing
gamma = 0.01 #Line Broadening
Alpha1ReAna = np.zeros(N, dtype='complex')
Alpha1ImAna = np.zeros(N, dtype='complex')
Alpha1ReNum = np.zeros(N, dtype='complex')
Alpha1ImNum = np.zeros(N, dtype='complex')
Omega = np.linspace(0.001, 0.8, N)
for i in range(N):
    Alpha1ReAna[i] = mp.re(AlphaK1(0, Omega[i] + gamma*1j))
    Alpha1ImAna[i] = mp.im(AlphaK1(0, Omega[i] + gamma*1j))
for i in range(int(N/M)):
    Alpha1ReNum[M*i] = mp.re(AlphaNUM(1, 0, Omega[M*i] + gamma*1j))
    Alpha1ImNum[M*i] = mp.im(AlphaNUM(1, 0, Omega[M*i] + gamma*1j))
ResIndex12 = np.argmin(np.abs(Omega-(En(2,0)-En(1,0))))
ResIndex13 = np.argmin(np.abs(Omega-(En(3,0)-En(1,0))))
ResIndex14 = np.argmin(np.abs(Omega-(En(4,0)-En(1,0))))

'Plotting, Ground State'
fig, ax1 = plt.subplots(figsize=(7.5,5))
ax1.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax1.set_ylabel(r"$\mathrm{Linear} \ \mathrm{Polarizability}, \ \tilde{\alpha}_{1,1}$", fontsize=16)
ax1.plot(Omega, Alpha1ReAna, label=r"$\mathrm{Re}(\tilde{\alpha}_{1,1})(\omega)$", color='red')
ax1.plot(Omega, Alpha1ImAna, label=r"$\mathrm{Im}(\tilde{\alpha}_{1,1})(\omega)$", color='green')
ax1.scatter(Omega[::M], Alpha1ReNum[::M], edgecolors='red', facecolors='none')
ax1.scatter(Omega[::M], Alpha1ImNum[::M], edgecolors='green', facecolors='none')
ax1.plot([En(2,0)-En(1,0), En(2,0)-En(1,0)], [Alpha1ImAna[ResIndex12], 35], color='green', linestyle=':', label=r"$\omega = \Delta E_{12}$")
ax1.plot([En(3,0)-En(1,0), En(3,0)-En(1,0)], [Alpha1ImAna[ResIndex13], 35], color='green', linestyle=':', label=r"$\omega = \Delta E_{13}$")
ax1.plot([En(4,0)-En(1,0), En(4,0)-En(1,0)], [Alpha1ImAna[ResIndex14], 35], color='green', linestyle=':', label=r"$\omega = \Delta E_{14}$")
ax1.plot([0, 0.3], [3, 3], color='red', linestyle=':', label=r"$\tilde{\alpha}_{1,1}(0) = 3$")
ax1.set_xlim([0, 0.8])
```

```
ax1.set_ylim([-15, 35])
ax1.tick_params(axis='x', labelsize=12)
ax1.tick_params(axis='y', labelsize=12)
ax1.grid('true')
ax1.legend(fontsize=16, ncol=1)
ax1.tick_params(axis='x', which='both', direction='inout', top=True, labeltop=True)
#plt.savefig("LinPolCoulombn1.pdf", dpi=300)

'Initial Calculations, First Excited State'
N = 1000 #Resolution
M = 20 #Numerical Spacing
gamma = 0.001 #Line Broadening
Alpha2ReAna = np.zeros(N, dtype='complex')
Alpha2ImAna = np.zeros(N, dtype='complex')
Alpha2ReNum = np.zeros(N, dtype='complex')
Alpha2ImNum = np.zeros(N, dtype='complex')
Omega = np.linspace(0.001, 0.45, N)
for i in range(N):
    Alpha2ReAna[i] = mp.re(AlphaK2(0, Omega[i] + gamma*1j))
    Alpha2ImAna[i] = mp.im(AlphaK2(0, Omega[i] + gamma*1j))
for i in range(int(N/M)):
    Alpha2ReNum[M*i] = mp.re(AlphaNUM(2, 0, Omega[M*i] + gamma*1j))
    Alpha2ImNum[M*i] = mp.im(AlphaNUM(2, 0, Omega[M*i] + gamma*1j))
ResIndex21 = np.argmin(np.abs(Omega-(En(2,0)-En(1,0))))
ResIndex23 = np.argmin(np.abs(Omega-(En(3,0)-En(2,0))))
ResIndex24 = np.argmin(np.abs(Omega-(En(4,0)-En(2,0))))
ResIndex25 = np.argmin(np.abs(Omega-(En(5,0)-En(2,0))))
ResIndex26 = np.argmin(np.abs(Omega-(En(6,0)-En(2,0))))

'Plotting, First Excited State'
fig, ax2 = plt.subplots(figsize=(7.5,5))
ax2.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax2.set_ylabel(r"$\mathrm{Linear} \ \mathrm{Polarizability}, \ \tilde{\alpha}_{2,1}$", fontsize=16)
ax2.plot(Omega, Alpha2ReAna, label=r"$\mathrm{Re}(\tilde{\alpha}_{2,1})(\omega)$", color='red')
ax2.plot(Omega, Alpha2ImAna, label=r"$\mathrm{Im}(\tilde{\alpha}_{2,1})(\omega)$", color='green')
ax2.scatter(Omega[::M], Alpha2ReNum[::M], edgecolors='red', facecolors='none')
ax2.scatter(Omega[::M], Alpha2ImNum[::M], edgecolors='green', facecolors='none')
ax2.plot([En(2,0)-En(1,0), En(2,0)-En(1,0)], [-1675, Alpha2ImAna[ResIndex21]], color='green', linestyle=':',
      label=r"$\omega = \Delta E_{21} = 0.375$")
ax2.plot([En(3,0)-En(2,0), En(3,0)-En(2,0)], [Alpha2ImAna[ResIndex23], 3500], color='green', linestyle=':',
      label=r"$\omega = \Delta E_{23} = 0.069$")
ax2.plot([En(4,0)-En(2,0), En(4,0)-En(2,0)], [Alpha2ImAna[ResIndex24], 3500], color='green', linestyle=':',
      label=r"$\omega = \Delta E_{24} = 0.094$")
ax2.plot([En(5,0)-En(2,0), En(5,0)-En(2,0)], [Alpha2ImAna[ResIndex25], 3500], color='green', linestyle=':',
      label=r"$\omega = \Delta E_{25} = 0.105$")
ax2.plot([En(6,0)-En(2,0), En(6,0)-En(2,0)], [Alpha2ImAna[ResIndex26], 3500], color='green', linestyle=':',
      label=r"$\omega = \Delta E_{26} = 0.111$")
ax2.plot([0, 0.06], [132, 132], color='red', linestyle=':', label=r"$\tilde{\alpha}_{2,1}(0) = 132$")
ax2.set_xlim([0, 0.4])
ax2.set_ylim([-1675, 3500])
ax2.tick_params(axis='x', labelsize=12)
ax2.tick_params(axis='y', labelsize=12)
ax2.grid('true')
ax2.legend(fontsize=16, ncol=2)
ax2.tick_params(axis='x', which='both', direction='inout', top=True, labeltop=True)
plt.savefig("LinPolCoulombn2.pdf", dpi=300)

'Initial Calculations, Kratzer Ground States'
N = 1000 #Resolution
M = 50 #Numerical Spacing
Delta = np.array((-0.25, -0.1, 0.1, 0.25), dtype='complex')
gamma = 0.01 #Line Broadening
Alpha1ReAna = np.zeros((4, N), dtype='complex')
Alpha1ImAna = np.zeros((4, N), dtype='complex')
Alpha1ReNum = np.zeros((4, N), dtype='complex')
Alpha1ImNum = np.zeros((4, N), dtype='complex')
```

```
Omega = np.linspace(0.001, 1, N)

for i in range(len(Delta)):
    for j in range(N):
        Alpha1ReAna[i,j] = mp.re(AlphaK1(Delta[i], Omega[j] + gamma*1j))
        Alpha1ImAna[i,j] = mp.im(AlphaK1(Delta[i], Omega[j] + gamma*1j))
    for j in range(int(N/M)):
        Alpha1ReNum[i, M*j] = mp.re(AlphaNUM(1, Delta[i], Omega[M*j] + gamma*1j))
        Alpha1ImNum[i, M*j] = mp.im(AlphaNUM(1, Delta[i], Omega[M*j] + gamma*1j))

'Plotting, Kratzer Ground States'
fig, ax3 = plt.subplots(2, 2, figsize=(10, 8), sharex=True, sharey=True)
ax3 = ax3.flatten()
ax31, ax32, ax33, ax34 = ax3[0], ax3[1], ax3[2], ax3[3]

ax31.plot(Omega, Alpha1ReAna[0,:], label=r"$\mathrm{Re}(\tilde{\alpha}_{1,1})(\omega)$", color='red')
ax31.plot(Omega, Alpha1ImAna[0,:], label=r"$\mathrm{Im}(\tilde{\alpha}_{1,1})(\omega)$", color='green')
ax31.scatter(Omega[::M], Alpha1ReNum[0,::M], edgecolors='red', facecolors='none')
ax31.scatter(Omega[::M], Alpha1ImNum[0,::M], edgecolors='green', facecolors='none')
ax31.plot([0, 0.15], [AlphaNul(1, Delta[0]), AlphaNul(1, Delta[0])], color='red', linestyle=':',
      label=r"$\tilde{\alpha}_{1,1}(0) = 9.61$")
ax31.tick_params(axis='x', labelsize=12)
ax31.tick_params(axis='y', labelsize=12)
ax31.set_ylabel(r"$\mathrm{Linear} \ \mathrm{Polarizability}, \ \tilde{\alpha}_{1,1}$", fontsize=16)
ax31.set_xlim([0, 1])
ax31.set_ylim([-28, 70])
ax31.grid('true')
DeltaEntry1 = Line2D([0], [0], color='none', label=r"$\delta = -0.25$")
ax31.legend(handles=[DeltaEntry1, *ax31.get_legend_handles_labels()[0]], fontsize=16, ncol=1)

ax32.plot(Omega, Alpha1ReAna[1,:], color='red')
ax32.plot(Omega, Alpha1ImAna[1,:], color='green')
ax32.scatter(Omega[::M], Alpha1ReNum[1,::M], edgecolors='red', facecolors='none')
ax32.scatter(Omega[::M], Alpha1ImNum[1,::M], edgecolors='green', facecolors='none')
ax32.plot([0, 0.25], [AlphaNul(1, Delta[1]), AlphaNul(1, Delta[1])], color='red', linestyle=':',
      label=r"$\tilde{\alpha}_{1,1}(0) = 4.92$")
ax32.tick_params(axis='x', labelsize=12)
ax32.tick_params(axis='y', labelsize=12)
ax32.set_xlim([0, 1])
ax32.grid('true')
DeltaEntry2 = Line2D([0], [0], color='none', label=r"$\delta = -0.1$")
ax32.legend(handles=[DeltaEntry2, *ax32.get_legend_handles_labels()[0]], fontsize=16, ncol=1)

ax33.plot(Omega, Alpha1ReAna[2,:], color='red')
ax33.plot(Omega, Alpha1ImAna[2,:], color='green')
ax33.scatter(Omega[::M], Alpha1ReNum[2,::M], edgecolors='red', facecolors='none')
ax33.scatter(Omega[::M], Alpha1ImNum[2,::M], edgecolors='green', facecolors='none')
ax33.plot([0, 0.425], [AlphaNul(1, Delta[2]), AlphaNul(1, Delta[2])], color='red', linestyle=':',
      label=r"$\tilde{\alpha}_{1,1}(0) = 1.75$")
ax33.tick_params(axis='x', labelsize=12)
ax33.tick_params(axis='y', labelsize=12)
ax33.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax33.set_ylabel(r"$\mathrm{Linear} \ \mathrm{Polarizability}, \ \tilde{\alpha}_{1,1}$", fontsize=16)
ax33.set_xlim([0, 1])
ax33.grid('true')
DeltaEntry3 = Line2D([0], [0], color='none', label=r"$\delta = 0.1$")
ax33.legend(handles=[DeltaEntry3, *ax33.get_legend_handles_labels()[0]], fontsize=16, ncol=1)

ax34.plot(Omega, Alpha1ReAna[3,:], color='red')
ax34.plot(Omega, Alpha1ImAna[3,:], color='green')
ax34.scatter(Omega[::M], Alpha1ReNum[3,::M], edgecolors='red', facecolors='none')
ax34.scatter(Omega[::M], Alpha1ImNum[3,::M], edgecolors='green', facecolors='none')
ax34.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax34.plot([0, 0.7], [AlphaNul(1, Delta[3]), AlphaNul(1, Delta[3])], color='red', linestyle=':',
      label=r"$\tilde{\alpha}_{1,1}(0) = 0.69$")
ax34.tick_params(axis='x', labelsize=12)
```

```
ax34.tick_params(axis='y', labelsize=12)
ax34.set_xlim([0, 1])
ax34.grid('true')
DeltaEntry4 = Line2D([0], [0], color='none', label=r"$\delta = 0.25$")
ax34.legend(handles=[DeltaEntry4, *ax34.get_legend_handles_labels()[0]], fontsize=16, ncol=1)
plt.savefig("LinPolKratzern1.pdf", dpi=300)

'Initial Calculations, Kratzer Excited States'
N = 1000 #Resolution
M = 50 #Numerical Spacing
Delta = np.array((-0.25, -0.1, 0.1, 0.25), dtype='complex')
gamma = 0.001 #Line Broadening
Alpha2ReAna = np.zeros((4, N), dtype='complex')
Alpha2ImAna = np.zeros((4, N), dtype='complex')
Alpha2ReNum = np.zeros((4, N), dtype='complex')
Alpha2ImNum = np.zeros((4, N), dtype='complex')
Omega = np.linspace(0.001, 0.5, N)

for i in range(len(Delta)):
    for j in range(N):
        Alpha2ReAna[i,j] = mp.re(AlphaK2(Delta[i], Omega[j] + gamma*1j))
        Alpha2ImAna[i,j] = mp.im(AlphaK2(Delta[i], Omega[j] + gamma*1j))
    for j in range(int(N/M)):
        Alpha2ReNum[i, M*j] = mp.re(AlphaNUM(2, Delta[i], Omega[M*j] + gamma*1j))
        Alpha2ImNum[i, M*j] = mp.im(AlphaNUM(2, Delta[i], Omega[M*j] + gamma*1j))

'Plotting, Kratzer Excited States'
fig, ax4 = plt.subplots(2, 2, figsize=(10, 8), sharex=True, sharey=True)
ax4 = ax4.flatten()
ax41, ax42, ax43, ax44 = ax4[0], ax4[1], ax4[2], ax4[3]

ax41.plot(Omega, Alpha2ReAna[0,:], label=r"$\mathrm{Re}(\tilde{\alpha}_{2,1})(\omega)$", color='red')
ax41.plot(Omega, Alpha2ImAna[0,:], label=r"$\mathrm{Im}(\tilde{\alpha}_{2,1})(\omega)$", color='green')
ax41.scatter(Omega[::M], Alpha2ReNum[0,::M], edgecolors='red', facecolors='none')
ax41.scatter(Omega[::M], Alpha2ImNum[0,::M], edgecolors='green', facecolors='none')
ax41.plot([0, 0.045], [AlphaNul(2, Delta[0]), AlphaNul(2, Delta[0])], color='red', linestyle=':',
      label=r"$\tilde{\alpha}_{2,1}(0) = 258.72$")
ax41.tick_params(axis='x', labelsize=12)
ax41.tick_params(axis='y', labelsize=12)
ax41.set_ylabel(r"$\mathrm{Linear} \ \mathrm{Polarizability}, \ \tilde{\alpha}_{2,1}$", fontsize=16)
#ax41.set_xlim([0, 1])
#ax41.set_ylim([-28, 70])
ax41.grid('true')
DeltaEntry1 = Line2D([0], [0], color='none', label=r"$\delta = -0.25$")
ax41.legend(handles=[DeltaEntry1, *ax41.get_legend_handles_labels()[0]], fontsize=16, ncol=1)

ax42.plot(Omega, Alpha2ReAna[1,:], color='red')
ax42.plot(Omega, Alpha2ImAna[1,:], color='green')
ax42.scatter(Omega[::M], Alpha2ReNum[1,::M], edgecolors='red', facecolors='none')
ax42.scatter(Omega[::M], Alpha2ImNum[1,::M], edgecolors='green', facecolors='none')
ax42.plot([0, 0.0525], [AlphaNul(2, Delta[1]), AlphaNul(2, Delta[1])], color='red', linestyle=':',
      label=r"$\tilde{\alpha}_{2,1}(0) = 174.36$")
ax42.tick_params(axis='x', labelsize=12)
ax42.tick_params(axis='y', labelsize=12)
ax42.set_xlim([0, 1])
ax42.grid('true')
DeltaEntry2 = Line2D([0], [0], color='none', label=r"$\delta = -0.1$")
ax42.legend(handles=[DeltaEntry2, *ax42.get_legend_handles_labels()[0]], fontsize=16, ncol=1)

ax43.plot(Omega, Alpha2ReAna[2,:], color='red')
ax43.plot(Omega, Alpha2ImAna[2,:], color='green')
ax43.scatter(Omega[::M], Alpha2ReNum[2,::M], edgecolors='red', facecolors='none')
ax43.scatter(Omega[::M], Alpha2ImNum[2,::M], edgecolors='green', facecolors='none')
ax43.plot([0, 0.07], [AlphaNul(2, Delta[2]), AlphaNul(2, Delta[2])], color='red', linestyle=':',
      label=r"$\tilde{\alpha}_{1,1}(0) = 98.59$")
ax43.tick_params(axis='x', labelsize=12)
```

```
ax43.tick_params(axis='y', labelsize=12)
ax43.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax43.set_ylabel(r"$\mathrm{Linear} \ \mathrm{Polarizability}, \ \tilde{\alpha}_{2,1}$", fontsize=16)
ax43.set_xlim([0, 1])
ax43.grid('true')
DeltaEntry3 = Line2D([0], [0], color='none', label=r"$\delta = 0.1$")
ax43.legend(handles=[DeltaEntry3, *ax43.get_legend_handles_labels()[0]], fontsize=16, ncol=1)

ax44.plot(Omega, Alpha2ReAna[3,:], color='red')
ax44.plot(Omega, Alpha2ImAna[3,:], color='green')
ax44.scatter(Omega[::M], Alpha2ReNum[3,::M], edgecolors='red', facecolors='none')
ax44.scatter(Omega[::M], Alpha2ImNum[3,::M], edgecolors='green', facecolors='none')
ax44.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax44.plot([0, 0.085], [AlphaNul(2, Delta[3]), AlphaNul(2, Delta[3])], color='red', linestyle=':',
     label=r"$\tilde{\alpha}_{2,1}(0) = 61.91$")
ax44.tick_params(axis='x', labelsize=12)
ax44.tick_params(axis='y', labelsize=12)
ax44.set_xlim([0, 1])
ax44.grid('true')
DeltaEntry4 = Line2D([0], [0], color='none', label=r"$\delta = 0.25$")
ax44.legend(handles=[DeltaEntry4, *ax44.get_legend_handles_labels()[0]], fontsize=16, ncol=1)
plt.savefig("LinPolKratzern2.pdf", dpi=300)

'Initial Calculations, Coulomb Pockels'
N = 1000 #Resolution
M = 50 #Numerical Spacing
gamma = 0.01 #Line Broadening
Alpha1RePock = np.zeros(N, dtype='complex')
Alpha1ImPock = np.zeros(N, dtype='complex')
Alpha1ReLim = np.zeros(N, dtype='complex')
Alpha1ImLim = np.zeros(N, dtype='complex')
Omega = np.linspace(0.001, 0.8, N)
for i in range(len(Omega)):
   Alpha1RePock[i] = mp.re(AlphaC11(Omega[i] + gamma*1j))
   Alpha1ImPock[i] = mp.im(AlphaC11(Omega[i] + gamma*1j))
   Alpha1ReLim[i] = -mp.re(AlphaK1(0, Omega[i] + gamma*1j))
   Alpha1ImLim[i] = -mp.im(AlphaK1(0, Omega[i] + gamma*1j))

'Plotting, Coulomb Pockels'
fig, ax5 = plt.subplots(1, 2, figsize=(10, 8), gridspec_kw={'wspace': 0.25})
ax5 = ax5.flatten()
ax51, ax52 = ax5[0], ax5[1]
ax51.plot(Omega, Alpha1RePock, label=r"$\mathrm{Re}(\tilde{\alpha}_{1,11})(\omega)$", color='red')
ax51.plot(Omega, Alpha1ImPock, label=r"$\mathrm{Im}(\tilde{\alpha}_{1,11})(\omega)$", color='green')
ax52.plot(Omega, Alpha1ReLim, label=r"$\mathrm{Re}(\tilde{\alpha}_{1,1})(\omega)$", color='red')
ax52.plot(Omega, Alpha1ImLim, label=r"$\mathrm{Im}(\tilde{\alpha}_{1,1})(\omega)$", color='green')
ax51.plot([0, 0.3], [alphan2Stat(1,0), alphan2Stat(1,0)], color='red', linestyle=':',
     label=r"$\tilde{\alpha}_{2,1}(0) = -\frac{81}{4}$")
ax52.plot([0, 0.3], [-AlphaNul(1, 0), -AlphaNul(1, 0)], color='red', linestyle=':',
     label=r"$-\tilde{\alpha}_{2,1}(0) = -3$")
ax51.set_xlim([0, 0.55])
ax52.set_xlim([0, 0.55])
ax51.set_ylim([-373, 178])
ax52.set_ylim([-32, 13])
ax51.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax51.set_ylabel(r"$\mathrm{Pockels} \ \mathrm{Polarizability}, \ \tilde{\alpha}_{1,11}(\omega)$", fontsize=16)
ax52.set_xlabel(r"$\mathrm{Field} \ \mathrm{Frequency}, \ \omega$", fontsize=16)
ax52.set_ylabel(r"$\mathrm{Linear} \ \mathrm{Polarizability}, \ -\tilde{\alpha}_{1,1}(\omega)$", fontsize=16)
ax51.tick_params(axis='x', labelsize=12)
ax51.tick_params(axis='y', labelsize=12)
ax52.tick_params(axis='x', labelsize=12)
ax52.tick_params(axis='y', labelsize=12)
ax51.grid(True)
ax52.grid(True)
ax51.legend(fontsize=16, ncol=1)
ax52.legend(fontsize=16, ncol=1)
```

```
plt.savefig("PockelsPolCoulombn1.pdf", dpi=300)
```