Lifetime Studies on Magnesium Rydberg Atom

Submitted by

Adithyan Radhakrishnan

A report outlining the work carried out during May-July 2018
as a part of a summer internship

supervised by
Dr Matt Grau
Prof Dr Jonathan Home
Trapped Ion Quantum Information Group
ETH Zurich

November 2018

Abstract

Rydberg atoms are a suitable candidate for realising qubits owing to their long lifetimes and coherent optical control near autoionisation states thanks to their strong atomic polarisability [1]. My work during the course of this internship involved designing a detection chamber to study autoionisation channel in a Magnesium atom in order to obtain the angular momentum profile and working towards obtaining the lifetime values by numerically integrating the radial Schrödinger equation using python. I used a model potential (namely the Klapisch potential [2]) and optimised its internal parameters for solving the Schrodinger equation. The results and roadblocks encountered in the implementation of the optimisation scheme constitute most of my findings and are therefore discussed in detail in this report.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Motivation

An atom whose outermost electron is excited to a high principal quantum number n is said to be a Rydberg atom. At these high n, the Rydberg atom behaves similar to a hydrogen atom (i.e a two body system involving a positive core and a negative charge in the potential of the core). The energy levels of this system can be given by the Rydberg-Ritz formula:

$$E_{n,l,j} = -\frac{R_{Mg}}{(n - \delta_{n,l,j})^2} \tag{1.1}$$

where

$$\delta_{n,l,j} = E_{l,j} + \frac{A}{n'^2} + \frac{B}{n'^4} + \frac{C}{n'^6} \tag{1.2}$$

n, l, j are the quantum numbers corresponding to a hydrogen atom $n' = n - E_{l,j}$

R_{Mg} is the scaled Rydberg constant for Magnesium (Note : $E_{l,j}$ is a quantum defect parameter depending on l, j and is therefore different from $E_{n,l,j}$ in eq 1.1.

Since the electron is reasonably far away from the positive core, it develops a strong atomic polarizability and therefore becomes susceptible to interaction with electromagnetic fields. Electrons in Rydberg states also have long lifetimes owing to decreasing overlap between wavefunction of the electron and the core with increasing orbital angular momentum (l) values. These properties are used in carrying out quantum information studies, for e.g, by entangling two Rydberg atoms using Lasers and encoding qubits in them while trapping them on a lattice.[3] .

The project on neutral atom trapping at the Trapped Ion Quantum Information (TIQI) group at ETH Zürich has a working Magneto-Optical Trap (MOT) for conducting studies on Rydberg states in Magnesium [4]. Gaseous Magnesium atoms are trapped in the MOT using the singlet S (3^1S_0) to the singlet P (3^1P_1) valence electron transition which is achieved using a 285 nm laser. A 375 nm laser is used for further excitation to the

Rydberg states. When the 375 nm laser is not active, a camera inside the chamber would be constantly recording the spontaneous emission fluorescence from the singlet P back to the S state. The loss in fluorescence recorded by the camera as the 375nm laser is switched on as the electron goes to a Rydberg state is used to detect the Rydberg transitions and obtain the spectrum.

1.0.1 Anomalies in the Rydberg Spectra

As per dipole selection rules, the Rydberg transitions from the singlet P state 3^1P_1 can take the electron either to an s or a d orbital angular momentum state for every principal quantum n in the Rydberg series. The transition frequencies for these can be calculated from the quantum defect parameters given in [5]. Using the 375nm laser, the experiment was carried out at TIQI, the position of peaks in the recorded spectra (loss in fluorescence intensity vs frequency) was compared with the theoretically calculated values. It was observed that there were peaks at the position of the calculated values corresponding to d transitions but no peaks at the values for the calculated position for the s transitions were observed. The experiment was repeated for various other principal quantum numbers in the Rydberg series but the s peaks however were not observed in any of those. This can be either due to the these s states being too short-lived to be detected by the camera or that the transitions do not actually take place, defying the selection rules.

My project, therefore, was conceived to investigate this anomaly further and get a better idea of the angular momentum profile of the Rydberg atoms. The project investigates the anomaly using an experimental as well as theoretical approach . The experimental approach involves subjecting the Rydberg atoms to field ionization in order to characterize the angular momentum profile and the theoretical approach involves numerically evaluating the radial wavefunction in order to obtain the lifetime values for these states. The angular momentum profile from the experiment would throw light on the distribution of angular momentum along electric field direction (z-direction). L_z (z component of orbital angular momentum) of s can take only the value of 0. whereas L_z of d can take values from $\{-2, -1, 0, 1, 2\}$. Therefore one can take a look at the L_z profile from this experiment and determine whether the s states are present depending on whether L_z takes values other than 0. The magnitude of L_z can be determined using a time-of-flight evaluation of the detection at the MCP. Atoms with the Rydberg electron having higher L_z values would ionise faster [6].

On the other hand, by numerically calculating the radial wavefunction from the Schrödinger equation one can go on to calculate the dipole matrix elements for the Magnesium atom

and thereby calculate the lifetime or transition probability from it. Therefore, this would help us get closer to explaining the anomaly of the missing lines.

Chapter 2

Detector Chamber Design

The chamber was conceived as an extension to the existing Mangeto-Optical Trap (MOT) [4] to detect Magnesium Rydberg species by field ionising them and detecting the resulting ions using a Multi Channel Plate (MCP) detector. A low voltage pulse then ionizes these excited atoms after which a high voltage pulse accelerates the resulting ion to the MCP.

2.1 Parts

The set-up consists of four levels of Stainless Steel flat electrodes and a circular mesh above them. The mesh helps shield the effect of the field due to electrodes on the ion as it escapes to the MCP. The electrodes and mesh are mounted on four blocks made of Macor (referred to as spacers here onwards) (Fig 2.1). The set up is housed in a octagonal chamber having 8 flange holes on the sides with a standard mating specification of CF35 and two CF100 flange mating from the top and bottom. The MCP is mounted on the top flange hole. Two of the side flange holes are used as viewports for the 375nm and 285nm lasers.

2.1.1 Electrodes

2.1.1.1 Design Requirements and Challenges

The electrodes are responsible for causing the field induced ionisation in the incoming Rydberg atom and also providing acceleration to the resulting ion towards the MCP. There are holes drilled on each of these electrodes along their common axis. The holes, in addition to providing a passage for the resulting ion, also have an effect on the field
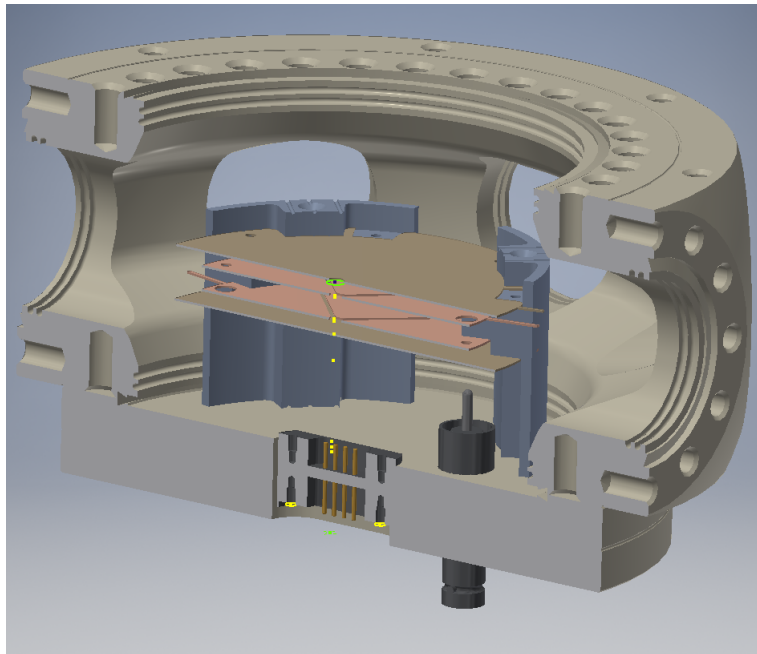
FIGURE 2.1: Chamber Assembly consisting of two high voltage electrodes (brown), two low voltage electrodes (sandstone red) and spacers (grey).

strength along the common axis. For eg, if the holes on each of the electrodes are exactly congruent, the outer electrodes will be completely shielded by the inner ones; or if the inner hole is too big with respect to the outer ones, it could lead to a diminished field strength near the axis when the low voltage is switched on. One has to therefore optimise the spacing between the electrodes and the radii of the holes on the common axis in such a way that it keeps the shielding of the field caused by one set of electrodes on another at a minimum while ensuring a decent voltage gradient (field strength) at the atom's intended position. These radii and the spacing between electrodes were determined numerically using C (included in the appendix). Particularly for this experiment, we decided to segment those electrodes that are responsible for field induced ionisation to have better control of the field. As a consequence, I had to look for appropriate mounting methods(explained in a later section) to ensure proper isolation of the electrodes without compromising on the mechanical stability of the entire set up.

2.1.1.2 Numerical Calculations for Spacing Parameters

To determine the axial spacing between these levels of electrodes (along the common axis of these electrodes) and hole radii having optimum voltage gradient/electric field experienced by the Mg atom, one needs to solve the free space Laplace equation

$$\nabla^2 \phi(x, y, z) = 0 \tag{2.1}$$
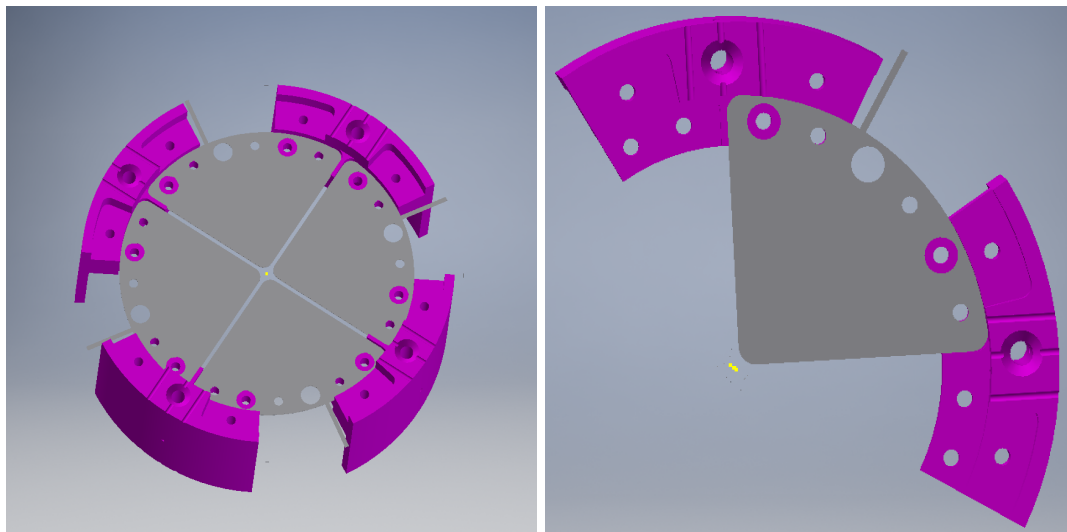
FIGURE 2.2: Top view of the segmented electrodes. Right: Single segment (metallic colored) placed on two spacers (magenta). Left: all four electrodes on spacers.

with certain boundary conditions at the positiom of the electrodes. If the exact geometry (with holes) of the electrodes are taken into consideration, this boundary value problem becomes difficult to solve analytically since finding a suitable Green's function for the geometry is a daunting task. Therefore one needs to solve it numerically. To make the problem even simpler and get only a rough estimate of the separation, I assumed the plates to be square plates having holes at the center such that the dimensions of the plates is much greater than the axial separation and the hole radii. This approximation results in field lines being axial in the space between the holes. Taking into account this resulting symmetry along the radial plane, the equation reduces to (with appropriate boundary conditions):

$$\nabla^2 \phi(z) = 0 \tag{2.2}$$

where ϕ is the potential function, z is the axial direction

Keeping in mind the assembly picture (Fig 2.1) there are 4 parameters to be optimized namely h_1, r_1, h_2, r_2 corresponding to distance between the inner plates, radius of the holes of the inner plates, distance between the outer and inner plates, radius of the holes of the outer plates respectively. (Fig 2.3).The boundary conditions for equation 2.2 therefore becomes:

FIGURE 2.3: Simplified Plate configuration used in solving the Laplace equation. The vertical lines correspond to the plates, vertical spacing between them correspond to holes and horizontal spacing correspond to axial separation between the plates. The dotted line is the imaginary line perpendicular to the common axis of the plates about which the entire configuration is symmetric. h_1,h_2 are axial separation parameters while r_1, r_2 are length of the square holes
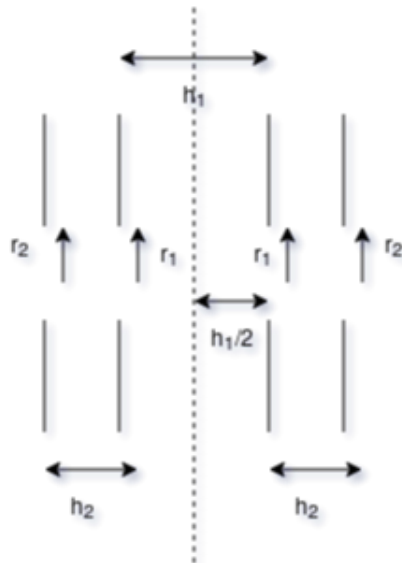
.

$$\phi(z = -\tfrac{h_1}{2}) = -V_1$$
$$\phi(z = \tfrac{h_1}{2}) = V_1$$
$$\phi(z = -\tfrac{h_1}{2} - h_2) = -V_2$$
$$\phi(z = \tfrac{h_1}{2} + h_2) = V_2$$

The numerical integration of this differential equation was carried out along z direction assuming symmetry in the y and x directions. V_1 and V_2 were set as 5V and 1000V respectively. The gradient of the solution (i.e, the field strength) is calculated by implementing a simple forward differentiation method. The values for h_1, h_2, r_1 and r_2 was set at a starting value of 2 mm each and was varied individually for each parameter to observe the field strength variations experienced by the atom.

As mentioned previously, the job of the calculation was to look for a set of values for h_1, h_2, r_1 such that shielding of one set of electrodes (high/low voltage) on another was minimum when those electrodes are acting on the atom. The plots of the field at the atom's position vs the plates' hole radius showed that the field grew slowly to its maximum value post 75% of the maximum value when the hole sizes were varied separately for the individual plates. Therefore I fixed the values of the hole lengths such that the combination of r_1 and r_2 gave the field value to be 75% of the maximum value. Fixing this value of the field, I looked for the combination of h_1 and h_2 that retains this value. From the available data, the set of values that gave that satisfied this criterion

was $h_1 = 3mm$, $h_2 = 5mm$, $r_1 = 5mm$, $r_2 = 2mm$. The figure 2.4 shows the variation of field strength vs the r_1 when all the other three parameters are kept constant.The mesh is placed at distance of 6 mm from the top most electrode (Fig 2.1) to allow the escape of all atoms contained in the solid angle region formed by the outer hole of the mesh and the center of the setup.
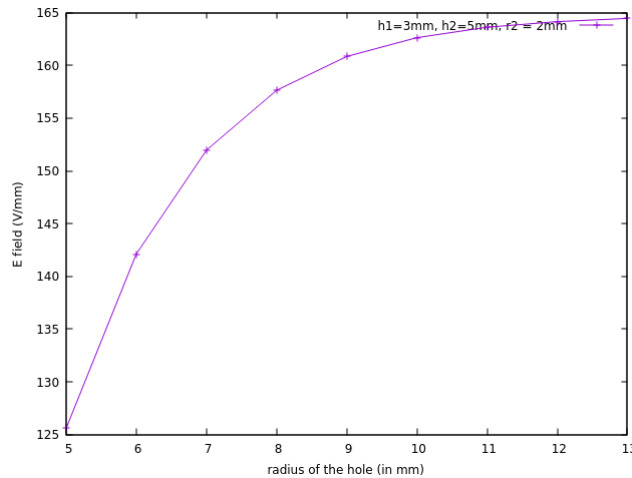
FIGURE 2.4: Variation of electric field with $r_1 (radius of the hole)$ keeping all other parameters constant. At $r_1 = 5$ the value of the field is 125 V/mm which is roughly 75% of the maximum value i.e 165 V/mm.

2.1.2 Spacer

The spacer made of Macor supports the electrodes and the mesh and also serves as a reference for aligning them with respect to each other. The overall setup consists of circular arrangement of four identical spacer blocks with gaps between each of them from where the laser beam could enter.

The segmented electrodes imposed mounting constraints while designing the spacer. The gap between two consecutive segments is 2 mm.Two consecutive blocks of spacer hold one segment together while ensuring that this gap is achieved. The alternate pattern of screw holes (refer fig 2.5) is to ensure that the mounting screws of the top electrode segment do not come in contact with the below segment and vice versa. Thus, the screw heads will be facing up and down in an alternate matter if viewed from top. Sufficient number of grooves are cut out on the spacer to ensure easy escape of air trapped between the screw threads and in the gaps between the mesh and the spacer, during outgassing to achieve vacuum in the chamber.

FIGURE 2.5: Side view of the spacer illustrating the levels of electrodes placed on it

Chapter 3

Numerical Calculations Using Python

Rydberg atoms are suitable candidates for realising quantum computers owing to their long lifetimes and coherent control. The information about the lifetimes and coherent control could be extracted from the radial wavefunction of the system. To obtain the radial wavefunction through numerical integration from the Radial Schrodinger Equation (RSE)
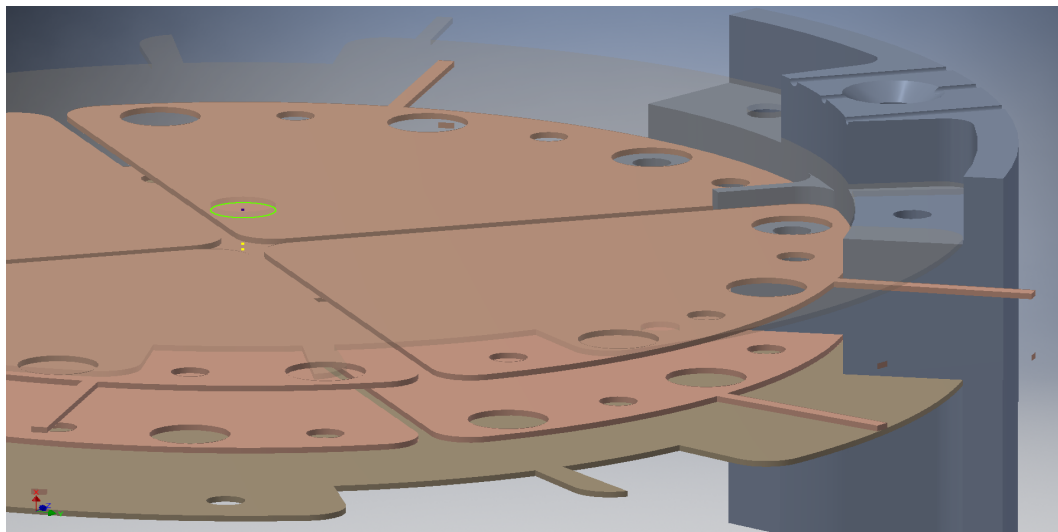
$$\frac{d^2\psi}{dx^2} = (V + \frac{l(l+1)}{r^2} - E)\psi(x) \tag{3.1}$$

the potential term V in the system Hamiltonian was modelled as the Klapisch potential [2] :

$$V(r) = -\frac{1}{r} - \frac{(Z-1)e^{-ar}}{r} + be^{-cr} \tag{3.2}$$

where the term containing the parameter a depicts the shielding of the outer electrons (excluding the Rydberg electron) as one moves away from the nucleus; the last term depicts the repulsive potential contribution from the Rydberg electron. The parameters a, b and c are obtained by fitting the eigenvalues of the Hamiltonian against the experimentally obtained energies for various Magnesium l, s, j energy states using the method of simulated annealing. The integrated wavefunctions are then used to calculate the dipole matrix elements and lifetimes for various Mg Rydberg states. This chapter details the results and roadblocks of the optimisation algorithm implementation for fitting a, b, c including the various sub-routines that make up the entire implementation. The calculations has been done in python in an object oriented fashion and I have made use of the existing git repositories Alkali Metal Rydberg Calculator and simulated annealer making use of their parent class results.

3.1 Relevant Theory

3.1.1 Simulated Annealing

Simulated annealing is a global optimisation process which is particularly useful when the global optimum is hidden within poorly resolved local optima. Simulated annealing could be intuitively understood as an analogy with thermodynamics, specifically with the way that liquids freeze and crystallize or metals cool and anneal [7] At high temperatures, the molecules of a liquid move freely with respect to one another. If the liquid is cooled slowly, thermal mobility is lost. The atoms are often able to line themselves up and form a pure crystal that is completely ordered over a distance up to billions of times the size of an individual atom in all directions. This crystal is the state of minimum energy for this system. In nature, the slowly cooled crystal are at a lower energy as compared to a liquid metal that is cooled quickly or "quenched," - it is known to end up in a polycrystalline or amorphous state having somewhat higher energy.

Expanding on this analogy, the objective function that we would like to optimise/minimise becomes the energy of the system and this energy is optimised in a series of iterations as the temperature decreases with each iteration. The routine starts with a set of initial conditions for the temperature (set to a high value) and number of steps per each iteration. During these iterations, the routine scans over the objective function and converges slowly on the region of the global optima ignoring insignifcant local optima at high temperatures. As it closes down on the exact region of interest as at low temperatures, the local optima in that region is scrutinised more carefully using a steepest-descent heuristic.

In this particular problem of optimising the Klapisch parameters, firstly a subroutine provides the heuristic eigenvalues for a set of (a, b, c) as a column vector X which is then subtracted from the column vector containing the experimental eigenvalues X' such that

$$O = X - X' \tag{3.3}$$

The squared norm of O ($\| O \|^2$) is then set as our objective function. The simulated annealing was implemented and carried out by using the python package 'simanneal.py'.

3.1.2 Solving for Eigenvalues

The bottleneck step in the process of annealing is the subroutine that evaluates the eigenvalues of the Hamiltonian. For this two separate methods were investigated:

1. Canonical Functions Method (CFM) [8] which converts the radial Schrödinger equation from the usual boundary value problem (BVP) to a set of initial value problems.

2. Converting the differential equation into a matrix equation by discretising r:

$$AX = EX \tag{3.4}$$

where A is some n (an integer) dimensional differential operator (expression 3.1.2.2), X is an n dimensional column vector containing function values at consecutive discrete intervals, and E is the eigenvalue corresponding to an eigenfunction X.

The following section describes the methods in detail and also the results of the simulated annealing implementation in each of these methods.

3.1.2.1   Canonical Functions Method (CFM)

The radial Schrodigner Equation (RSE) for an atom can be converted from a boundary value problem with boundary conditions as

$$\lim_{r \to 0} \psi(r) = 0 \tag{3.5}$$

and

$$\lim_{r \to \infty} \psi(r) = 0 \tag{3.6}$$

to a set of initial value problems (IVP) by writing $\psi(x)$ in terms of the 'canonical' functions $\alpha(E; r)$ and $\beta(E; r)$ as:

$$\psi(r) = \psi(r_0)\alpha(E; r) + \psi'(r_0)\beta(E; r) \tag{3.7}$$

where r_0 is the initial value of the radius for integration such that $\alpha(E; r_0) = 1$; $\alpha'(E; r_0) = 0$; $\beta(E; r_0) = 0$; $\beta'(E; r_0) = 1$.

The eigenvalues of the RSE would be the roots of the function

$$F(E) = \lim_{r \to \infty} \frac{\alpha(E; r)}{\beta(E; r)} - \lim_{r \to 0} \frac{\alpha(E; r)}{\beta(E; r)} \tag{3.8}$$

Applying the boundary conditions (eq 3.5) and (eq 3.6) to the equation 3.7 , we get:

$$\lim_{r \to \infty} \frac{\alpha(E; r)}{\beta(E; r)} = -\frac{\psi'(r_0)}{\psi(r_0)} \tag{3.9}$$

$$\lim_{r \to 0} \frac{\alpha(E;r)}{\beta(E;r)} = -\frac{\psi'(r_0)}{\psi(r_0)} \qquad (3.10)$$

The R.H.S in equations 3.9 and 3.10 are the same given that $\alpha(E;r)$ and $\beta(E;r)$ are canonical functions corresponding to bound state $\psi(r)$. And $\psi(r)$ is a bound state if E is an eigenvalue of the radial Schrödinger equation (3.1). Therefore the terms $\lim_{r\to 0} \frac{\alpha(E;r)}{\beta(E;r)}$ and $\lim_{r\to\infty} \frac{\alpha(E;r)}{\beta(E;r)}$ are equal for every eigenvalue E.

Hence, the roots of the function

$$F(E) = \lim_{r \to \infty} \frac{\alpha(E;r)}{\beta(E;r)} - \lim_{r \to 0} \frac{\alpha(E;r)}{\beta(E;r)}$$

will be the energy eigenvalues of the bound states as stated in Eq 3.8.

The implementation carries out two subroutines calculate the ratio $\frac{\alpha(E;r)}{\beta(E;r)}$- one at r$\to \infty$ and the other at $r \to 0$ - both starting from a particular initial radius r_0. In principle r_0, could be any value but for this experiment I chose $r_0 = 5$ (in Hartree units). It turns out that, one need not carry out the integration until they reach arbitrarily close to $r = 0$ or $r = \infty$ in order to obtain $\frac{\alpha(E;r)}{\beta(E;r)}$. Rather as shown in [8], the ratio reaches saturation and grows slowly well before the limits of 0 and infinity. Therefore, I implemented a separate method to detect the saturation and stop the integration at that point. The integration method used here is the Numerov method [9] where for one (fixed) time step the increment is given by

$$y_{n+1} = \frac{2(1 - \frac{5}{12}h^2 k_n^2)y_n - (1 + \frac{1}{12}h^2 k_{n-1}^2)y_{n-1}}{1 + \frac{1}{12}h^2 k_{n+1}^2} \qquad (3.11)$$

where h is the step size of the integration and k_n in this case would be $V(r_n) + \frac{l(l+1)}{r_n^2} - E$ (eq 3.1).

The implementation was tested for the Hydrogen atom potential. However the calculated values from the implementation was very notably different from that of the known values.

The graphs below show the function F(E) (Eq 2.11) for different values of N i.e, $\frac{1}{stepsize}$. The rough behaviour of F(E) as expected from [8] is seen in the graphs for N=1000 and N=10000. However, the roots, as seen from the graph, clearly do not correspond to the eigenvalues of Hydrogen atom which are supposed to be roughly equal to $\frac{-1}{n^2}$ for n $\in$ (1,2,3...).
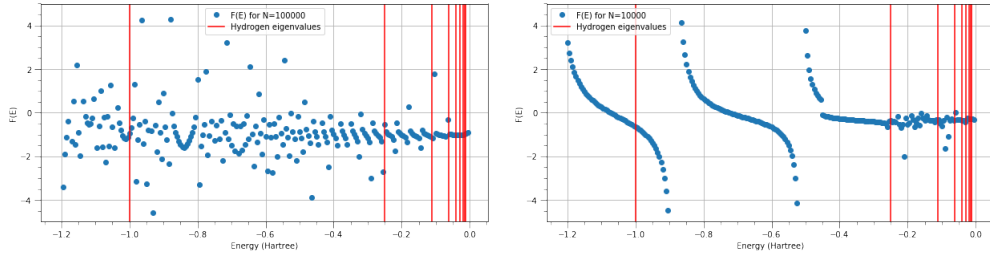
FIGURE 3.1: F(E) vs E for N=10000 and N=100000. The roots of F(E) do not coincide with that of the Hydrogen eigenvalues. In fact, there is a root between the first two eigenvalues.
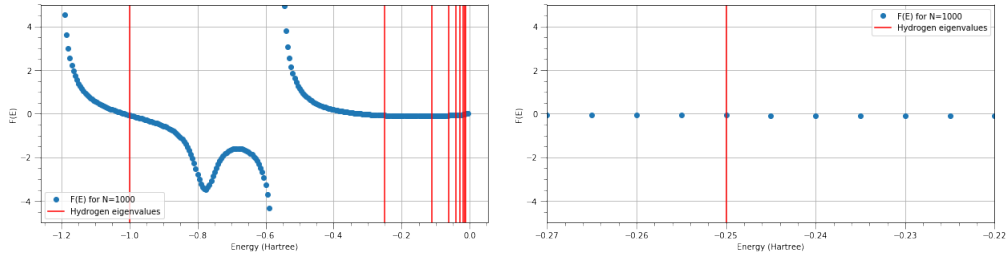
FIGURE 3.2: F(E) vs E for N=1000.The image to the right depicts a magnified version of the plot for the region x = (-0.27, -0.23). Multiple solutions exist in that region and therefore the method is not reliable.

3.1.2.2 Matrix Method For Solving The RSE

The method effectively transforms the ODE into a matrix eigenvalue equation (2.7)

$$AX = EX \tag{3.12}$$

$$.\, A = \begin{Bmatrix} (-2/h^2 + V + \frac{l(l+1)}{r_1^2}) & \frac{-1}{h^2} & 0 & 0 & . & . \\ \frac{-1}{h^2} & (-2/h^2 + V + \frac{l(l+1)}{r_2^2}) & \frac{-1}{h^2} & 0 & . & . \\ 0 & (\frac{-1}{h^2} & (-2/h^2 + V + \frac{l(l+1)}{r_3^2}) & \frac{-1}{h^2} & 0 & . \\ & . & & & & \\ & . & & & & \\ & . & & & & \\ . & . & & 0 & (\frac{-1}{h^2} & -2/h^2 + V + \frac{l(l+1)}{r_n^2}) \end{Bmatrix}$$

$$X = \begin{Bmatrix} \psi(r_1) \\ \psi(r_2) \\ \psi(r_3) \\ \psi(r_4) \\ . \\ . \\ . \\ \psi(r_n) \end{Bmatrix}$$

$h = r_i - r_{i-1}$

n = number of discrete points in the range of r $\in [r_0, r_{n+1}]$

Using the above method for the Hydrogen atom potential, following graph illustrates the closeness of the calculated eigenvalues with the theoretical values:
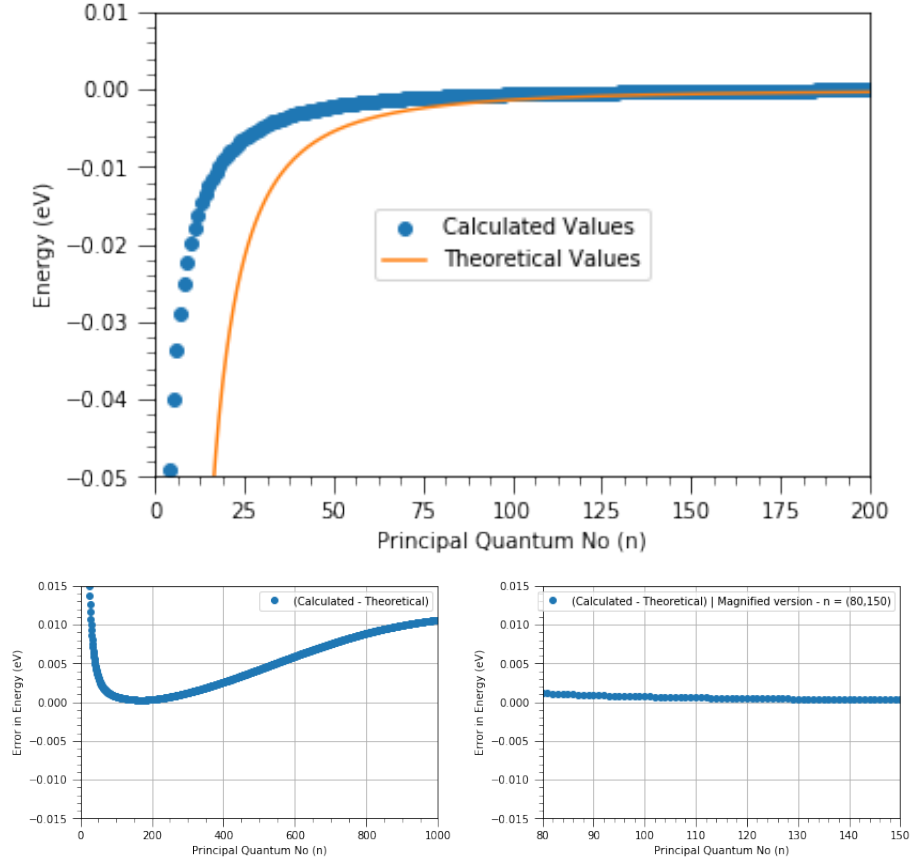
FIGURE 3.3: Plot of energy eigenvalues obtained from the matrix method(top). Figures on the bottom is plot of the error between the two plots shown above.

Since the error associated with the second method is smaller, it was adopted as the subroutine which would extract the eigenvalues for optimization using simulated annealing.

3.2 Annealing Results

The annealing routine was implemented with the radial Schrodinger equation solver subroutine. The annealing returns the set of a, b, c parameters for which the objective function (eq 3.3) is minimum. To ensure more accurate result in terms of objective function optimisation, I ran the simulation for 10 times anticipating convergence of the

parameter solutions towards a particular set of values. The following graphs illustrate the annealing results for the different orbital angular momentum (l) values. The x axis denotes the simulation number and y axis denotes the parameter value. The blue dots correspond to the value obtained from a particular simulation number. The yellow line denotes the average of all the 10 runs. It can be noted from the figures that while a is remains fixed with each simulation run, b and c do not show the tendency to converge to a particular value:

FIGURE 3.4: Annealing results a, b, c (in the order of figures) $l = 0$. Blue dots correspond to the parameter value at each run and the yellow line parallel to the x-axis corresponds to the mean of all the blue dot values.
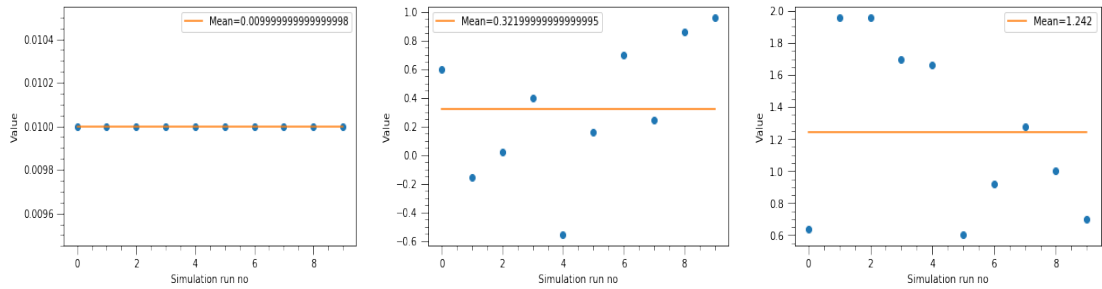
FIGURE 3.5: Annealing results a, b, c (in the order of figures) for $l = 1$. Blue dots correspond to the parameter value at each run and the yellow line parallel to the x-axis corresponds to the mean of all the blue dot values.

FIGURE 3.6: Annealing results a, b, c (in the order of figures) for $l = 2$. Blue dots correspond to the parameter value at each run and the yellow line parallel to the x-axis corresponds to the mean of all the blue dot values.
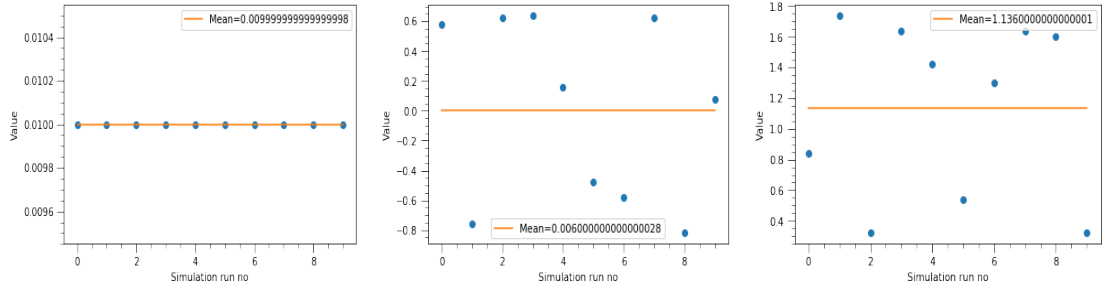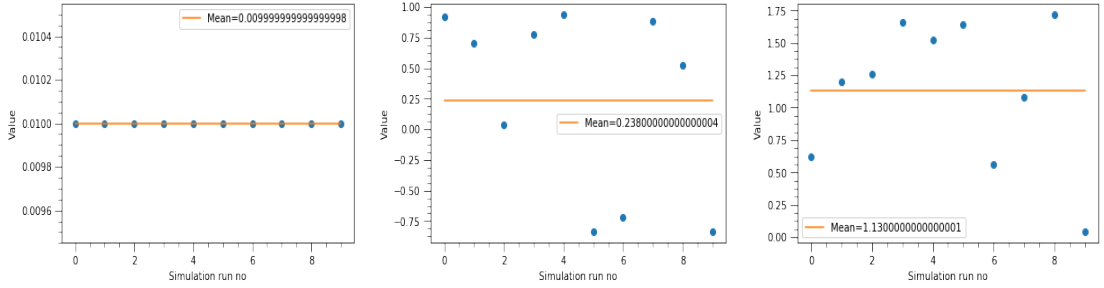

FIGURE 3.7: Annealing results a, b, c (in the order of figures) for $l = 3$. Blue dots correspond to the parameter value at each run and the yellow line parallel to the x-axis corresponds to the mean of all the blue dot values.
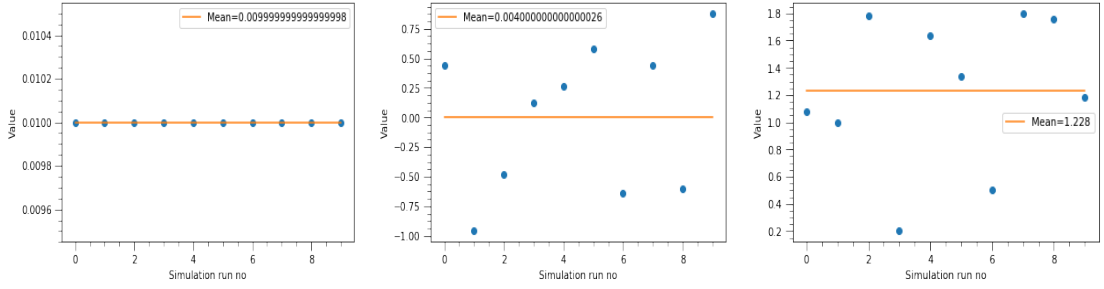
3.3 Simulated Annealing for Quantum Defect parameters

For Magnesium Rydberg states with a single excited electron, the energy level of a particular state can be given by the Rydberg Ritz formula:

$$E_{n,l,j} = -\frac{R_{Mg}}{(n - \delta_{n,l,j})^2} \tag{3.13}$$

where

$$\delta_{n,l,j} = E_{l,j} + \frac{A}{n'^2} + \frac{B}{n'^4} + \frac{C}{n'^6} \tag{3.14}$$

n' =n - $E_{l,j}$

R_{Mg} = scaled Rydberg constant for Magnesium (Note : $E_{l,j}$ is a quantum defect parameter depending on l, j and is therefore different from $E_{n,l,j}$ in eq 3.13.

Although the data for quantum defect parameters $E_{l,j}, A_{l,j}, B_{l,j}, C_{l,j}$ is available in the work by S.F Dyubko et al[5], we decided to run a simulated annealing scheme to obtain our own set of quantum defect values for the Rydberg energy levels measured in our lab. The closeness of our values with that of [5] would help us determine the accuracy of the annealing implementation. In addition to serving as a double check for annealing, it

would also serve to shed some light on the mysterious peaks observed in our lab which do not follow the calculations using the defect parameters given in [5]. The results are illustrated in the following graphs:
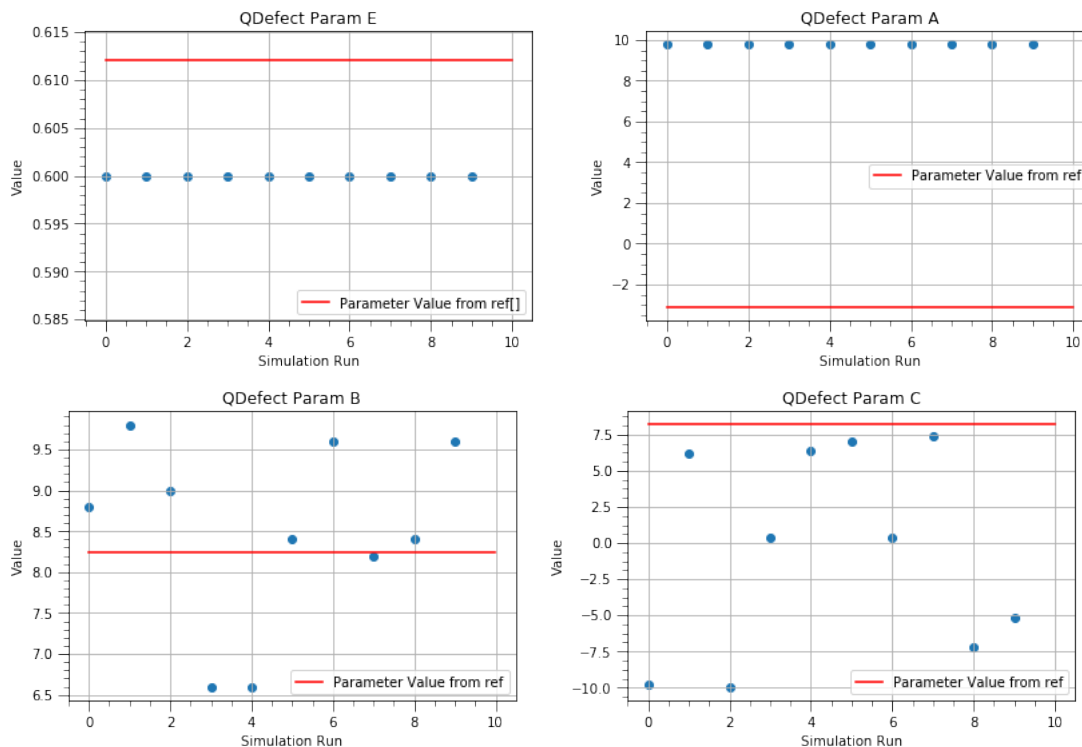
FIGURE 3.8: Annealing results for Quantum Defect parameters E (top left), A (top right), B (bottom left), C (bottom right)

Two of the parameters E, A converges where as B, C do not. While this might indicate a possible error in the simulated annealing implementation as suspected earlier, one could not rule out the possibility of having multiple solutions to the optimization problem of a particular objective function. To test it, I calculated and plotted the energy values corresponding to the defect values from all the 10 simulation runs (figures 3.8). These plots were compared with the plot of energy values corresponding to the defect calculated from [5] on the same graph.
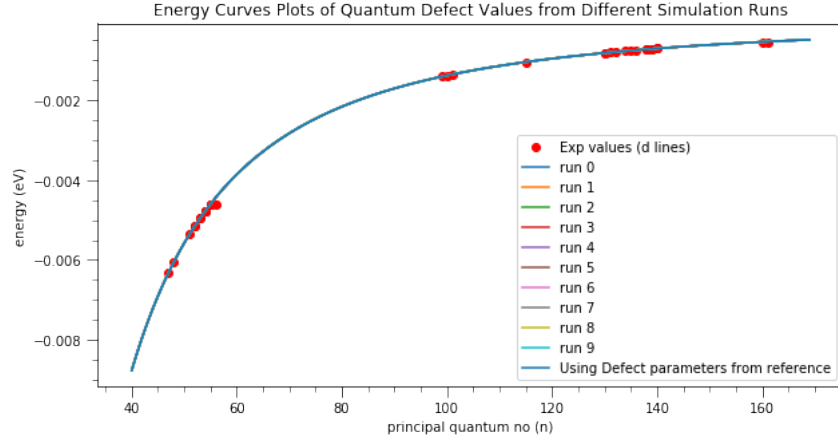
FIGURE 3.9: Plots of energy values $E_{n,l,j}$ using quantum defect values obtained from simulation runs (figures 3.8) and superposed
. Here 'reference' in the last plot refers to ref [5].

The figure goes on to show that there is very little change in the energy levels with respect to large changes in the quantum defect parameters. The plot of difference between energies obtained from annealing and those obtained from the values given in [5] (Fig 2.1)makes this point clear:
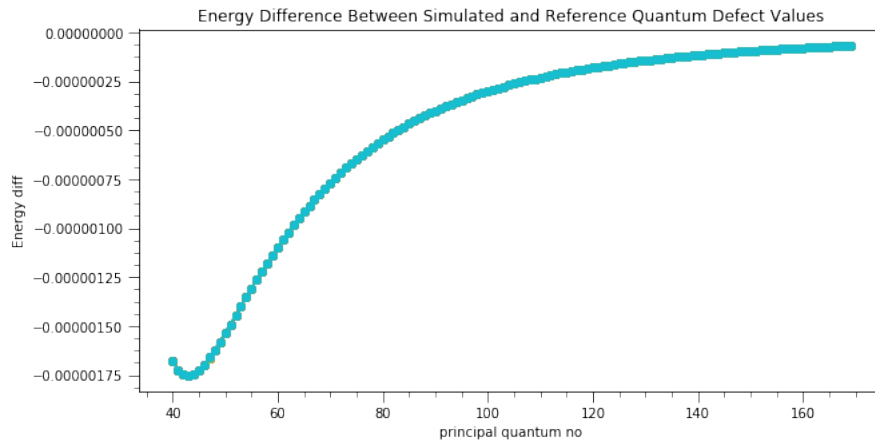
FIGURE 3.10: Energy Difference Between Simulated and Reference Quantum Defect Values

Therefore, it can be concluded that the simulated annealing for quantum defect parameters does not yield a convergent solution owing to multiple solutions for the optimization problem. One needs to come up with an alternative optimisation algorithm which is an improvement upon simulated annealing (stochastic tunneling, dual phase evolution to name a few).

Chapter 4

Results, Conclusions and Outlook

l value	a	b	c
0	0.099	0.006	1.136
1	0.099	0.322	1.242
2	0.099	0.004	1.228
3	0.099	0.238	1.13

TABLE 4.1: Mean value of Klapisch Parameters

l value	a	b	c
0	0	0.579	0.545
1	0	0.453	0.514
2	0	0.591	0.525
3	0	0.723	0.532

TABLE 4.2: Standard Deviation of Klapisch Parameters for each l

1. The results of the simulated annealing optimisation for both Klapisch parameters (4.1) and quantum defects do not seem to converge to a particular set of values as the simulation routine iterated . This indicates on first glance, a lack of enough constraints to bound the solution to a small enough region in the parameter space. But according to [10], simulated annealing being a stochastic method can indeed give multiple solutions in each run. Moreover simulated annealing is best used for optimising in a discrete parameter space. Since Klapisch parameter optimisation was rather carried out in a continuous parameter space, the objective function needs to satisfy the convergent conditions in [11] which were not checked before running the annealing. To improve the existing annealing implementation, one can use the method of adaptive simulated annealing [12].

21

2. Once the Klapisch parameters are in hand, one could integrate the radial Schrödinger equation to get the radial dipole matrix elements and thereafter calculate the lifetime of Mg species after considering Multi-Channel quantum defect theory. For a single valence electron Rydberg species, the matrix element for a field operator E could be written as:

$$\left\langle j, m \left| E_q^k \right| j', m' \right\rangle = \left\langle j, m | j', m' \right\rangle \left\langle j \left| E^k \right| j' \right\rangle \tag{4.1}$$

where, k indicates the order of approximation (dipole, quadrupole etc) and q is the direction of polarisation. The last term on the right is the reduced matrix element.

However, for a multi valence electron Rydberg species like Mg, the fine structure splitting involves both valence electrons and the core which makes the situation a bit more complicated and equation 4.1 is no longer valid for the Rydberg electron of such a system. Hence, it requires a more in depth treatment involving multi channel quantum defect theory to get more information about the important atomic physics parameters.

3. Another noticeable anomaly with the annealing implementation is that the first parameter stays constant in each simulation run as other values change around a mean value. This could be a possible bug in the implementation of the package or some bug in the source code itself in that the 'move()' function which moves to a supposedly random state in the parameter space (to check for the objective function value) is not completely random.

Appendix

Code Used for Solving Free Space Laplace Equation in one dimension

```c
#include<stdio.h>
#include<math.h>
#include<stdlib.h>

double calc_pot(int,int,int, int, int, int,int,int);
int main()
{
  int no;
  printf("simulation no\n");
  scanf("%d",&no);


  double l_x, l_y, l_z;
  int N_x, N_y, N_z, h1,h2, r1,r2,r3;


  //setting no of points of division /decides the scale
  N_x = 30;
  N_y = 30;
  h1= 5;
  h2= 5;
  N_z = h2+2*h1;
  r1 = 5;
  r2 = 4;
  r3=0;
  //allocating memory for Vx,Vy,Vz
  /*****Electric Field Calc ******/

  double ****E;

  E = malloc(sizeof(double)*(N_x-1));
```

```
//print to file

FILE *fp, *fp2, *fp3;
//fp = fopen("potential.txt","w");
char name[50];
sprintf(name,"simulation_%d",no);
fp2 = fopen(name,"w");
//fp3 = fopen("log_z_field.txt","w");
double E_temp;

    //simulation starts

    int h1temp, h2temp, r1temp, r2temp; //temporary variables
    h1temp = 3;
    h2temp=  5;
    r1temp= 5;
    r2temp = 2;
    //r3temp = 2;
    fprintf(fp2,"h1 \t h2 \t r2 \t r1 \t Ez \n");
    int counter=0;

    while(1)

    {
       E_temp = calc_pot(N_x,N_y,2*h1temp+h2temp, h1temp, h2temp,r1temp,
   r2temp,0);


       fprintf(fp2,"%d \t%d\t%d\t %d \t %lf\n",h1temp, h2temp,r2temp,
   r1temp,-E_temp);

       //simulation increments
       r1temp++;
       counter++; printf("step %d\n", counter);
       //simulation end conditions
       if(r1temp>13) break;

    }
  return 0;

}



double calc_pot(int N_x, int N_y, int N_z, int h1, int h2, int r1,int r2,
    int r3)
{
```

```c
double ***V, ***Vtemp;

V = malloc(sizeof(double)*N_x);
Vtemp = malloc(sizeof(double)*N_x);


for(int i=0;i<N_x; i++)
{
  V[i] = malloc(sizeof(double)*N_y);
  Vtemp[i] = malloc(sizeof(double)*N_y);

  for(int j=0; j<N_y; j++) { V[i][j] = malloc(sizeof(double)*N_z); Vtemp
  [i][j] = malloc(sizeof(double)*N_z);}

}

/*****ini_conditions******/

double V1,V2,V3,V4;
V1=-1000;
V2=-5;
V3=5;
V4=1000;

//At z=0
for(int i=0;i<N_x; i++)
{
  for(int j=0;j<N_y;j++){
  if(((i>(N_x/2-r3)) && (i<=N_x/2 + r3)) && ((j<=N_y/2 + r3) && (j>N_y/2
   - r3))) {V[i][j][h1-1]=0;}//printf("first cavity\n");}
  else V[i][j][0]=V1;
}}

//At z=h2
for(int i=0;i<N_x; i++)
{
  for(int j=0;j<N_y;j++)
  {
    if(((i>(N_x/2-r1)) && (i<=N_x/2 + r1)) && ((j<=N_y/2 + r1) && (j>N_y
  /2 - r1))) {V[i][j][h1-1]=0;}//printf("first cavity\n");}
    else  {V[i][j][h1-1]=V2; //printf("z=h2\n");}

  }
}}

//At z=h2+h1
for(int i=0;i<N_x; i++)
{
```

```
    for(int j=0;j<N_y;j++)
    {
      if(((i>(N_x/2-r1)) && (i<=N_x/2 + r1)) && ((j<=N_y/2 + r1) && (j>N_y
  /2 - r1))){;} //printf("second cavity\n");}
      else  {V[i][j][h1+h2-1]=V3;}//printf("z=h2+h1\n");


    }
}
//At z=h2+h1+h1
for(int i=0;i<N_x; i++)
{
  for(int j=0;j<N_y;j++){
      if(((i>(N_x/2-r2)) && (i<=N_x/2 + r2)) && ((j<=N_y/2 + r2) && (j>N_y
  /2 - r2))){;}
      else V[i][j][h2+h1+h1-1]=V4; //printf("not here\n");
}
}
/****** Jacobi Algorithm ******/

int iterations=0;
double diff = 10.;

//simulate first region
while((iterations<=500))
{

  for(int i=0; i<N_x; i++)
  {
    for(int j=0; j<N_y; j++)
    {
      for(int k=0; k<N_z; k++) Vtemp[i][j][k] = V[i][j][k];


    }

  }

  diff = 0;
  for(int i=1; i<N_x -1 ; i++)
  {
    for(int j=1; j<N_y - 1 ;j++)
    {
    if(!((i>(N_x/2-r1)) && (i<=N_x/2 + r1)) && ((j<=N_y/2 + r1) && (j>
  N_y/2 - r1)))   //leaving out  cavity
    {for(int k=1; k<h1-1 ;k++) { //first region z range
            V[i][j][k] =(V[i+1][j][k] + V[i-1][j][k] + V[i][j-1][k] + V[
  i][j+1][k] + V[i][j][k-1] + V[i][j][k+1])/6.0;
          diff =diff+fabs(V[i][j][k]-Vtemp[i][j][k]);//printf("first
  regions simulating..\n");
```

```
        }
      }
    }


  }

  diff = diff/(h1*N_x*N_y - h2*pow(2*r1,2));
  //printf("iteration %d \t diff %f\n", iterations,diff);
  iterations++;
  if(diff<0.0001) break;
}

//simulate 2nd region
iterations=0;
diff=10;
while((iterations<=500) && (diff>0.0001))
{

  for(int i=0; i<N_x; i++)
  {
    for(int j=0; j<N_y; j++)
    {
      for(int k=0; k<N_z; k++) Vtemp[i][j][k] = V[i][j][k];


    }

  }

  diff = 0;
  for(int i=1; i<N_x-1; i++)
  {
    for(int j=1; j<N_y-1;j++)
    {
      if(!((i>(N_x/2-r1)) && (i<=N_x/2 + r1)) && ((j<=N_y/2 + r1) && (j>
  N_y/2 - r1)))   //leaving out  cavity
      {
        for(int k=h1; k<h1+h2-1;k++) { //first region z range
            V[i][j][k] =(V[i+1][j][k] + V[i-1][j][k] + V[i][j-1][k] + V[
  i][j+1][k] + V[i][j][k-1] + V[i][j][k+1])/6.0;
          diff =diff+fabs(V[i][j][k]-Vtemp[i][j][k]);//printf("2nd
  region simulating...\n");
        }
      }
    }


  }
  diff = diff/(h2*N_x*N_y - h2*pow(2*r1,2));
  //printf("iteration %d \t diff %f\n", iterations,diff);
```

```
    iterations ++;


}


//simulate 3rd region
iterations=0;
diff=10;
while(iterations<=500 && diff>0.0001)
{

  for(int i=0; i<N_x; i++)
  {
    for(int j=0; j<N_y; j++)
    {
      for(int k=0; k<N_z; k++) Vtemp[i][j][k] = V[i][j][k];


    }

  }

  diff = 0;
  for(int i=1; i<N_x-1; i++)
  {
    for(int j=1; j<N_y-1;j++)
    {
    if(!((i>(N_x/2-r1)) && (i<=N_x/2 + r1)) && ((j<=N_y/2 + r1) && (j>
 N_y/2 - r1))) //leaving out  cavity
      {for(int k=h1+h2; k<h1+h2+h1-1;k++) { //first region z range
            V[i][j][k] =(V[i+1][j][k] + V[i-1][j][k] + V[i][j-1][k] + V[
 i][j+1][k] + V[i][j][k-1] + V[i][j][k+1])/6.0;
         diff =diff+fabs(V[i][j][k]-Vtemp[i][j][k]);if(iterations
 ==1000)printf("%d %d %d %lf\n", i,j,k,V[i][j][k]);
          }
      }

    }

  }
  diff = diff/(h1*N_x*N_y - h1*pow(2*r1,2)); //averaged by no of steps
  //printf("iteration %d \t diff %f\n", iterations,diff);
  iterations ++;

}


//simulate cavity region
iterations=0;
diff = 10;
while((iterations<=500) && (diff>0.0001))
```

```
  {

    for(int i=0; i<N_x; i++)
    {
      for(int j=0; j<N_y; j++)
      {
        for(int k=0; k<N_z; k++) Vtemp[i][j][k] = V[i][j][k];

      }

    }

    diff = 0;
    for(int i=1; i<N_x; i++)
    {
      for(int j=1; j<=N_y;j++)
      {
        if(((i>(N_x/2-r1)) && (i<=N_x/2 + r1)) && ((j<=N_y/2 + r1) && (j>
  N_y/2 - r1)))
        {   //if(iterations==2) printf("simulaing cavity\n");
            for(int k=1; k<h1+h2+h1-1;k++) {
                V[i][j][k] =(V[i+1][j][k] + V[i-1][j][k] + V[i][j-1][k] +
  V[i][j+1][k] + V[i][j][k-1] + V[i][j][k+1])/6.0;
                diff =diff+fabs(V[i][j][k]-Vtemp[i][j][k]);//printf("%d %d %
  d %lf\n", i,j,k, V[i][j][k]);
            }
        }
      }

    }


    diff = diff/pow(2*r1,2)/(h1+h2+h1);
    iterations ++;
  }

  double Etemp = -(V[N_x/2][N_y/2][N_z/2+1]-V[N_x/2][N_y/2][N_z/2-1])/2;
  return Etemp;
}
```

Listing 4.1: Laplace Solver

Appendix

Bibliography

[1] Mark Saffman. Quantum computing with atomic qubits and rydberg interactions: progress and challenges. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 49(20):202001, 2016.

[2] M Klapisch. A program for atomic wavefunction computations by the parametric potential method. *Computer Physics Communications*, 2(5):239–260, 1971.

[3] Sarah E Anderson, KC Younge, and Georg Raithel. Trapping rydberg atoms in an optical lattice. *Physical review letters*, 107(26):263001, 2011.

[4] Matt Grau, Christoph Fischer, Oliver Wipfli, and Jonathan Home. Designing an optical lattice trap for ions. In *APS Division of Atomic, Molecular and Optical Physics Meeting Abstracts*, 2017.

[5] SF Dyubko, VA Efremov, AS Kutsenko, NL Pogrebnyak, and KB MacAdam. Laser-microwave spectroscopy of singlet mg i atoms in s, p, d, f, g rydberg state. In *Advanced Optoelectronics and Lasers (CAOL), 2013 International Conference on*, pages 358–360. IEEE, 2013.

[6] HJ Korsch and R Möhlenkamp. Field ionization of rydberg atoms: A semiclassical treatment of complex energy states in intense electric fields. *Zeitschrift für Physik A Atoms and Nuclei*, 314(3):267–273, 1983.

[7] Vincent Granville, Mirko Krivánek, and J-P Rasson. Simulated annealing: A proof of convergence. *IEEE transactions on pattern analysis and machine intelligence*, 16 (6):652–656, 1994.

[8] Charbel Tannous, K Fakhreddine, and Jacques Langlois. The canonical function method and its applications in quantum physics. *Physics Reports*, 467(6):173–204, 2008.

[9] Veerle Fack and G Vanden Berghe. (extended) numerov method for computing eigenvalues of specific schrodinger equations. *Journal of Physics A: Mathematical and General*, 20(13):4153, 1987.

[10] David S Johnson, Cecilia R Aragon, Lyle A McGeoch, and Catherine Schevon. Optimization by simulated annealing: An experimental evaluation; part i, graph partitioning. *Operations research*, 37(6):865–892, 1989.

[11] M Locatelli. Simulated annealing algorithms for continuous global optimization: convergence conditions. *Journal of Optimization Theory and applications*, 104(1): 121–133, 2000.

[12] Lester Ingber. Adaptive simulated annealing (asa): Lessons learned. *arXiv preprint cs/0001018*, 2000.