Applied Math 205 - Trefethen
Final Project Written Report:
Numerical Computation for Eigenvalues of 1D Quantum Systems

Aidan D. Chambers
AMDG

December 5, 2024

Declarations

Textbook Chapter

The work in this project is most relevant to Chapter 10 from Heath [1], with a specific emphasis on the method outlined in Section 10.7. Additionally the topic of eigenvalue derivation aligns with Chapter 4, although in this paper we emphasize the derivation and calculation of a discrete matrix moreso than the process of computing its eigenvalues.

Contributions of the Authors

I was the only author of this project.

Use of AI Tools

Used assistance from ChatGPT solely to resolve errors with `matplotlib` or other plotting features.

Other Sources

Consulted with Professor Trefethen for ideas when deriving project.

I Introduction

The use of eigenvectors and eigenvalues to describe linear transformations is an incredibly powerful approach with near-universal applicability across many fields of science, math, and engineering. It is understandable then that a significant part of scientific computing is dedicated to processes for the efficient calculation of eigensystems; however most approaches in this realm – including the primary concepts in this class – emphasize finding solutions to discrete systems of the form

$$\mathbf{A}\mathbf{v}_n = \lambda_n \mathbf{v}_n, \tag{1.1}$$

where in the most general case $\mathbf{A} \in \mathbb{C}^{N\times N}$, $\mathbf{v} \in \mathbb{C}^N$, and $\lambda \in \mathbb{C}$ for finite integer N with $n \in [1, N]$. Despite this, the fundamental concept of eigenvalue analysis can be extended beyond matrices though to include eigenvalues of more general linear operators. A classic example of this general extension includes finding eigenvalues of the differential operators,

$$\frac{\mathrm{d}}{\mathrm{d}t} f(t) = \lambda f(t), \tag{1.2}$$

where we often describe $f(t)$ as the eigenfunction of the differential operator. In this sense we can regard Eq. (1.2) as either a continuous eigensystem or an infinite dimensional extension of the discrete case in Eq. (1.1). In either case, the same numerical approaches established for the discrete case must be amended to derive eigenfunctions and eigenvalues of differential operators.

Within physics, the predominant use for solving systems of eigenfunctions is to find solutions of the time independent Schrödinger equation,

$$\hat{\boldsymbol{H}}\boldsymbol{\psi}_n = E_n \boldsymbol{\psi}_n. \tag{1.3}$$

Comparing to Eq. (1.2), the eigenfunctions become the eigenstates ψ, our eigenvalues are the discrete energy values of our system E, and our differential operator is referred to as the Hamiltonian $\hat{\boldsymbol{H}}$. Expressed in position space for one dimension, the Hamiltonian has the form

$$\hat{\boldsymbol{H}} = -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + V(x), \tag{1.4}$$

where $\hbar$ is the reduced Planck constant, m is the mass of the particle being considered, and $V(x)$ is the potential that characterizes the system our particle is placed in. In this form, our eigenstates become functions of position $\psi(x)$ and solving Eq. (1.3) becomes analogous to solving a second order ordinary differential equation. We then run into the issue of $V(x)$ presenting a variable coefficient in our differential equation, which severely limits the types of potentials for which we can derive analytic solutions. As a result, physicists often resort to computational methods to obtain numerical results for the eigenvalue spectrum of a given Hamiltonian.

This paper focuses on exploring this computational process for deriving the eigenvalues of a quantum system described by a single variable. In Section II we will first derive a computational method for obtaining the eigenvalues and eigenstates from a Hamiltonian before using it to numerically solve the eigensystem of four different physical potentials. The first system in Section III.1 has well-defined analytic solutions, giving us a useful reference metric to evaluate our method's performance by comparing to the exact results. The remaining systems in Sections III.2 and IV.1 allow us to explore additional phenomena and extend to a true physical system with the Hydrogen atom. This segues into the final system in Section IV.2, where we will then explore applying our method to a simplified Hamiltonian for a composite particle with no higher order analytic solutions, where we will instead compare the predictions of our results with experimental data.

II Methodology and Convention

The initial scope and approach of this paper was largely driven by the discussion in Chapter 6 of [2] where the authors approach multiple potentials using Chebfun's native `quantumstates` function. This project recreates a similar approach instead using Python and its associated libraries. Our general approach is laid out by Section 10.7 of [1], which we will rephrase and narrow to fit the specific case described in the introduction. To start we write the time independent Schrödinger equation in one dimension on the domain $x \in [a, b]$,

$$-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2}\psi(x) + V(x)\psi(x) = E\psi(x). \tag{2.5}$$

We can then discretize the domain into N mesh points with spacings $\Delta x = (b-a)/(N+1)$ and approximate the second derivative using the finite difference formula. Rearranging slightly gives us

$$-\frac{\hbar^2}{2m(\Delta x)^2}\psi(x_{i+1}) + \frac{\hbar^2}{m}\left[\frac{1}{(\Delta x)^2} + \frac{m}{\hbar^2}V(x_i)\right]\psi(x_i) - \frac{\hbar^2}{2m(\Delta x)^2}\psi(x_{i-1}) = E\psi(x_i). \tag{2.6}$$

If we then define the symmetric tridiagonal matrix

$$\hat{\boldsymbol{H}} = \frac{\hbar^2}{m}\begin{bmatrix} \frac{1}{(\Delta x)^2}+\frac{m}{\hbar^2}V(x_1) & -\frac{1}{2(\Delta x)^2} & 0 & \cdots & 0 \\ -\frac{1}{2(\Delta x)^2} & \frac{1}{(\Delta x)^2}+\frac{m}{\hbar^2}V(x_2) & -\frac{1}{2(\Delta x)^2} & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -\frac{1}{2(\Delta x)^2} & \frac{1}{(\Delta x)^2}+\frac{m}{\hbar^2}V(x_{N-1}) & -\frac{1}{2(\Delta x)^2} \\ 0 & \cdots & 0 & -\frac{1}{2(\Delta x)^2} & \frac{1}{(\Delta x)^2}+\frac{m}{\hbar^2}V(x_N) \end{bmatrix}, \tag{2.7}$$

then we observe that Eq. (2.6) has been discretized to match the typical matrix form of the eigenvalue expression from Eq. (1.1). It follows then that the standard processes for obtaining the eigenvalues of a matrix can then be employed for our discrete Hamiltonian $\hat{\boldsymbol{H}}$. In particular, we will make use of `scipy.linalg.eigh_tridiagonal` [3], which returns computed eigenvalues and

normalized eigenvectors based on procedures similar to the `LAPACK S/DSTEMR` routine. Additionally, if we take the continuous limit $\Delta x \to 0$ – or equivalently, that $N \to \infty$ – we should expect our numerical results to converge on the exact analytic solutions, should they exist.

A key note on this method is that it assumes our system is subject to the boundary conditions $\psi(a) = \psi(b) = 0$. While these boundary conditions are clearly present for some systems like the infinite square well, they are less explicit for the systems considered in this paper. As a result, we turn to the fact that we expect our eigenstates to be square-integrable, i.e.

$$\int_{-\infty}^{\infty} \mathrm{d}x \, |\psi(x)|^2 = C, \tag{2.8}$$

for some finite value C, taken to be 1 if the eigenstates are normalized. This gives us the bounds that our functions should fall to zero far from the origin. As a result, we can accomplish the assumed boundary conditions above provided that our boundary points a and b are sufficiently large compared to the scale of our system with $a < 0 < b$.

On convention, we set $\hbar$ along with other parameters like mass to 1 whenever possible, allowing us to go between the physical energy values E_n and standard eigenvalues λ_n interchangeably. Additionally, we will use the notation $\psi_n(x, N)$ and $E_n(N)$ or $\lambda_n(N)$ to refer to the computationally derived n-th eigenstates and eigenvalues for N mesh points, i.e. $\psi_0(x, 10^2)$ refers to the lowest energy eigenstate derived using 100 mesh points. By contrast, $\psi_n(x)$ and E_n are used to represent the results from exact analytical solution.

III One Dimensional Potentials

III.1 Quantum Harmonic Oscillator

As an initial exploration of our method we apply it to the quantum harmonic oscillator in one dimension, defined by the quadratic potential

$$V_{\mathrm{HO}}(x) = \frac{1}{2}x^2. \tag{3.9}$$

This system is incredibly well studied and has the exact analytical solutions [4]

$$\psi_n(x) = \frac{1}{\sqrt{2^n n! \sqrt{\pi}}} \mathrm{e}^{-x^2/2} H_n(x) \quad \text{and} \quad E_n = (n + 1/2), \tag{3.10}$$

where $H_n(x)$ are the Hermite polynomials. We then apply this potential to our methodology outlined in Section II with the domain $x \in [-15, 15]$ for a variety of different positional steps, N. We show our results in Fig. 1, where we observe that our eigenvalues begin to converge well on the

exact eigenvalues from Eq. (3.10) for 10^3 steps or greater. This convergence continues well until $N \sim 10^6$ steps, when we observe the beginning of divergence between the exact and numerical eigenvalues for increasing N.

While this behavior seems to be at odds with our earlier statement that our method should converge exactly for $N \to \infty$, this was under the assumption that our calculations were done with exact arithmetic. By contrast, these numerical results were obtained from calculating the eigenvalues of a Hermitian tridiagonal matrix requiring about $\mathcal{O}(n^2)$ scalar operations, each of which are associated with a machine error due to floating point arithmetic. Redoing this process for several differently sized domains recovers this same behavior beginning at $N \sim 10^6$, upholding the idea that this is not an issue with our method but that this is where the additional convergence offered by increasing N is offset from the compounding floating point error from the calculation of an $N \times N$ matrix. This is further supported by observing the same discrepancies between the exact and numerical eigenstates in Appendix A. As a result, we will use $N = 10^5$ to compute our numerical eigenstates and eigenvalues for the remainder of our studies.

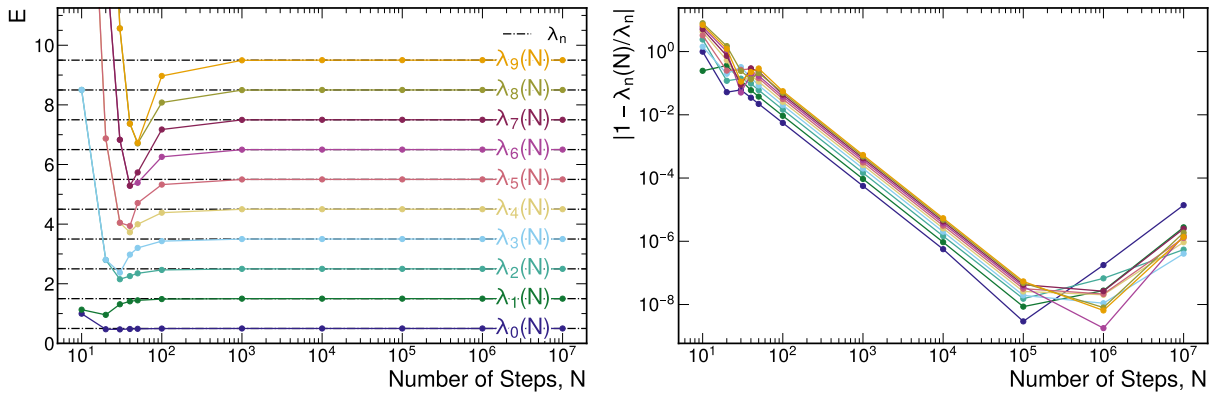

Figure 1: First ten energy eigenvalues for quantum harmonic oscillator defined by Eq. (3.9). Left plot compares computed result as a function of number of steps N to exact results, Eq. (3.10). Right plot shows percent error between computational and exact results on a log-log scale.

For $N = 10^5$ our numerical results suitably approximate the exact solutions for the harmonic oscillator, serving as a useful litmus test of our method and application. In Fig. 2 we show that the first ten eigenstates display the expected qualitative behaviors compared with previous results such as the computational results in [2]. As a final check of the behavior of our numerical eigenstates we ensure that they obey orthonormality, requiring that

$$\langle \psi_i | \psi_j \rangle = \int_{-\infty}^{\infty} \mathrm{d}x \, \psi_i(x) \psi_j(x) = \delta_{ij}, \tag{3.11}$$

where δ_{ij} is the Kronecker delta. We confirm this for our numerically derived states for $N = 10^5$ in Fig. 2, where our integral is replaced by the dot product of our two discrete vectors.

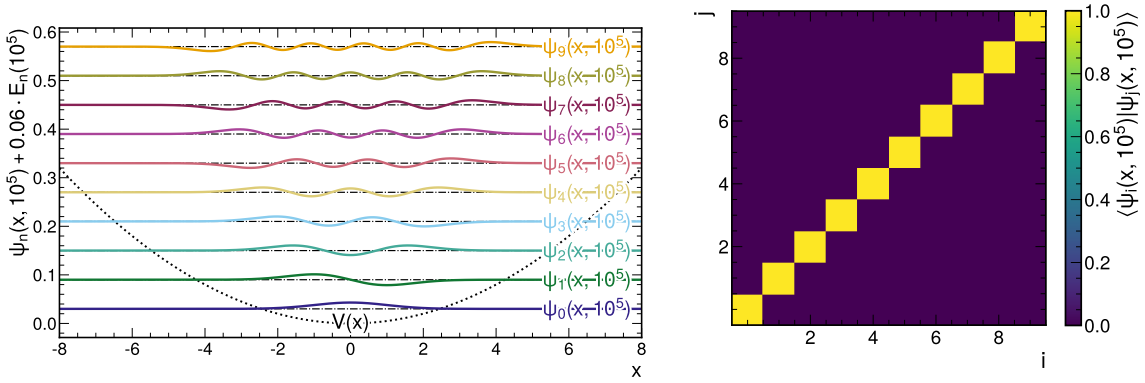

Figure 2: Left plot: Harmonic oscillator potential and first ten computationally obtained eigenstates displaced vertically by their eigenvalues for easier visualization. Right plot: Table of inner products – see Eq. (3.11) – between different computational eigenstates to demonstrate orthonormality.

III.2 Double Well Potential

After establishing our method's success with the harmonic oscillator we move onto the double well potential, defined by

$$V_{\text{DW}}(x) = x^2 + V_0 \mathrm{e}^{-(kx)^4}. \tag{3.12}$$

This potential describes our harmonic oscillator being split by a symmetric barrier with height V_0 and width parameter k. This system has no analytic solution, but is still described qualitatively [4]; the barrier divides the system into two symmetric wells, creating states localized to either side of the boundary with nearly degenerate eigenvalues. For states with eigenvalues greater than V_0 the barrier no longer isolates the eigenstates, resulting in the near degeneracy being broken.

To recreate this qualitative behavior we apply the potential from Eq. (3.12) to our method on the domain $x \in [-15, 15]$ using $V_0 = 8$ and two different values for k showing our results in Fig. 3. For $k = 4$ our observations align with our qualitative expectations with the first four eigenstates existing in nearly degenerate pairs. Additionally, eigenvalues become more distinct once they approach or exceed the height of the barrier. Repeating this for $k = 1$ which corresponds to a wider barrier, we observe a closer degeneracy of our eigenvalues for $E_n < V_0$, aligning with our intuition that a larger barrier should correspond greater degeneracy between our eigenvalues.

IV Central Force Potentials

Following our application to one dimensional systems we extend our method to include spherically symmetrical potentials in three dimensions. This does not require a significant reworking of our method due to an argument from [4] we will paraphrase here. For an isotropic potential $V(\mathbf{r}) = V(r)$

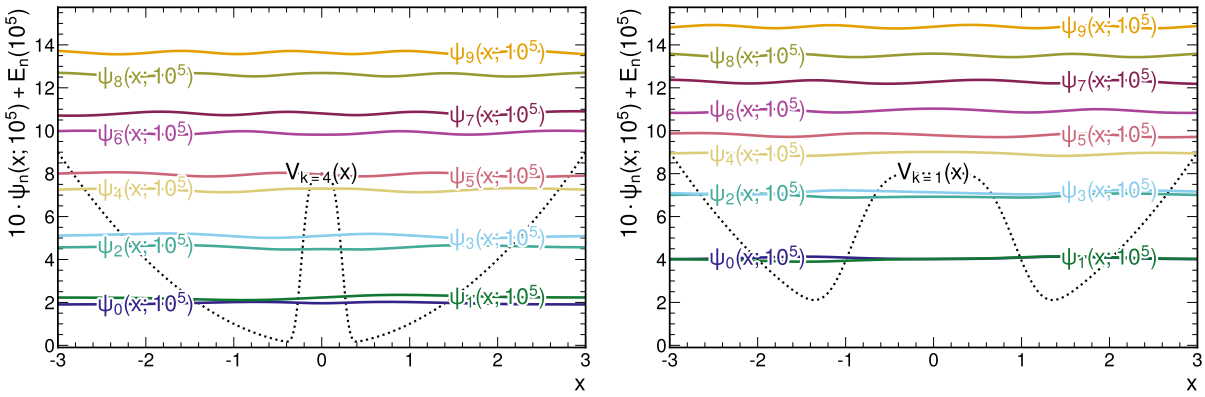

Figure 3: First ten eigenstates of double well potential from Eq. (3.12) with states displaced by their eigenvalues to show near degeneracy for states with eigenvalues below the height of the barrier.

we can factor the n-th eigenstate ψ_n into its radial and angular components,

$$\psi_{n\ell m}(\mathbf{r}) = R_{n\ell}(r) Y_\ell^m(\theta, \varphi) \tag{4.13}$$

where $Y_\ell^m(\theta, \varphi)$ are the spherical harmonics parametrized by quantum numbers ℓ and m. As an aside, our three quantum numbers n, ℓ, and m are integers satisfying the conditions

$$\begin{aligned} n &= 1, 2, 3, \dots \\ \ell &= 0, 1, 2, \dots, n-1 \\ m &= 0, \pm 1, \pm 2, \dots, \pm \ell. \end{aligned} \tag{4.14}$$

From here, rather than solving the full Schrödinger equation in spherical coordinates we can just focus on the radial components, where we find that introducing the substitution $u(r) = rR(r)$ allows us to reduce the radial equation to a one-dimensional differential equation,

$$-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial r^2} u_{n\ell}(r) + \underbrace{\left[V(r) + \frac{\ell(\ell+1)\hbar^2}{2mr^2}\right]}_{V_{\text{eff}}(r,\ell)} u_{n\ell}(r) = E_{n\ell} u_{n\ell}(r). \tag{4.15}$$

Comparing to Eq. (2.5) we see that this is functionally equivalent to the equation we are already able to solve numerically, with our single variable r replacing x and our potential being replaced with the effective potential $V_{\text{eff}}(r, \ell)$, parameterized by the angular momentum number ℓ. As a final note before switching to finding solutions of radial potentials, our boundary conditions now switch to $u(0) = u(b) = 0$ for sufficiently large b, taking advantage of the fact that our eigenstate should still be square integrable. This means that we will define our system on the domain $r \in [0, b]$.

IV.1 Hydrogen Atom

As our first example of a radial potential, we consider the hydrogen atom as our next physical system. This system is dominated by the electric interaction between the electron and nucleus, giving us the Coulomb potential

$$V_{\mathrm{H}}(r) = -\frac{e^2}{r}, \tag{4.16}$$

where e is the electric charge of both the electron and nucleus and r is the radial distance between them. Once again, this system is well described with the analytic solutions to the eigenvalues of the Coulomb potential in Eq. (4.16) given by [4]

$$E_n = -\frac{e^2}{2a_0}\frac{1}{n^2}, \tag{4.17}$$

where our answer in terms of the Bohr radius, $a_0 = \hbar^2/m_e e^2$ where m_e is the mass of the electron. Importantly, we note that our energy levels are only a function of the principal quantum number n, and should be independent of the angular momentum ℓ that defines our effective potential from Eq. (4.15).

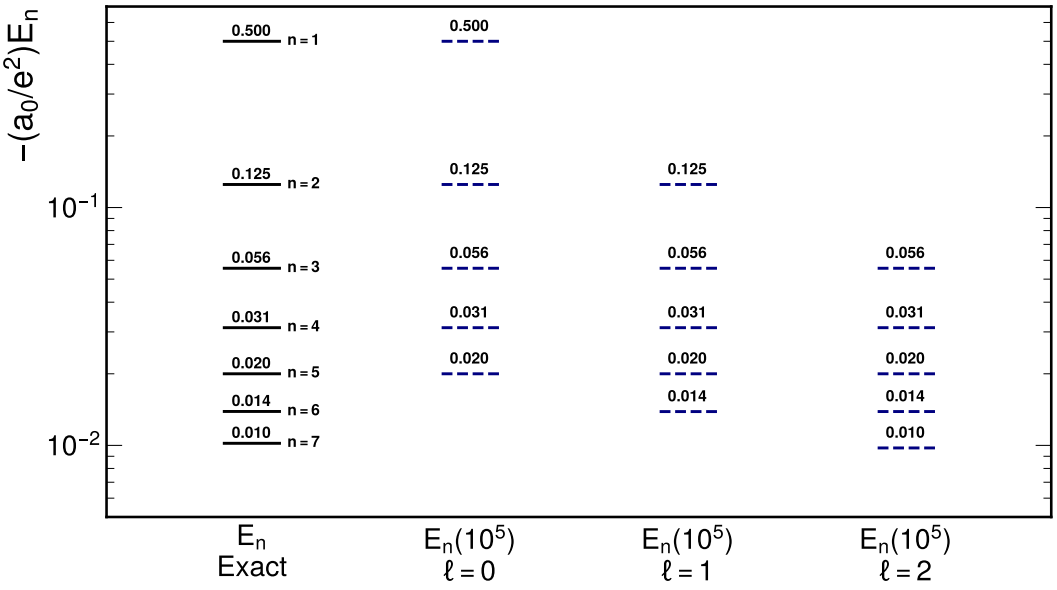

Figure 4: Hydrogen atom energy spectrum with exact results from Eq. (4.17) in left column, followed by computational results for three different values of ℓ which parametrizes $V_{\mathrm{eff}}(r, \ell)$ from Eq. (4.15).

To study this using our computational method we use the Coulomb potential from Eq. (4.16), analyzing the first five eigenvalues for $N = 10^5$ steps with the boundary conditions $r \in [0, 100]$. Additionally, we repeat our procedure for the first three values of ℓ to observe how the spectrum changes with response to the angular momentum quantum number. The resulting energy spectrum

(the arrangement of eigenvalues) is shown in Fig. 4, with the energy levels uniformly scaled and placed on a log scale for easier visualization. As a first observation, we see that our computational results converge incredibly well on the exact results for $\ell = 0$. We also notice that the first five energy levels for each value of ℓ are shifted relative to one another, which initially appears to violate our assumptions that the energy levels should be independent of this quantum number. In reality, it appears that our method has recovered the relationship between the quantum numbers from Eq. (4.14), in particular the condition that between n and ℓ that can be rewritten as $n \geq \ell+1$. By setting $\ell = 1$ in the effective potential we have restricted our Hamiltonian to only present the higher energy eigenstates that can support this angular momentum.

IV.2 Charmonium

From here we introduce the last system we will analyze in this paper, in which we study the pairing of two particles known as charm and anti-charm quarks[1], which we'll refer to as $c\bar{c}$ states. The resulting bound state forms a composite particle conventionally referred to as "charmonium" which was first discovered in 1974 [5]. While this discovery was incredibly important for the world of particle physics, it is relevant for our study as an experimental system of heavy quarks.

The formal mathematical description of composite particles requires the relativistic theory of quantum chromodynamics (QCD). The relatively higher mass of the charm quarks means that this system can be and has historically been [5] reasonably approximated as non-relativistic, allowing us to use the same approach outlined in Section II with the Schrödinger equation for a central force potential, similar to the Hydrogen atom approach in Section IV.1. In particular, the Coulomb potential is replaced by a simple linear model for the nuclear strong force,

$$V_{c\bar{c}}(r) = \frac{r}{\alpha_s}, \tag{4.18}$$

where α_s is a parameter for the strength of the interaction between the quarks which we will take to be $\alpha_s \simeq 1$ fm/ GeV ≈ 6 GeV^{-2} using the particle physics convention of natural units. Additionally we set our mass to the reduced mass of the two quarks, $\mu_{c\bar{c}} = m_c/2 \approx 636.5$ MeV using the value from [6]. Finally, in order to compare with real data we will treat the eigenvalues obtained as the binding energy of the system. Using some basic relativity, our numerical results will be mass spectrum described by the sum of the quarks' masses and the binding energy, i.e.

$$M_{n\ell}(N) = 2m_c + E_{n\ell}(N). \tag{4.19}$$

As a reminder, n describes the energy ordering of eigenstates for a given angular momentum ℓ approximated by our computational method for N mesh points.

[1] Quarks are subatomic particles that are often bound to one another by the nuclear strong force, making up composite particles like protons. Anti-quarks are identical in every way apart from having opposite electric charge.

We then apply Eq. (4.18) to our numerical method for $N = 10^5$ mesh points on the domain $r \in [0, 100]$ for $\ell = 0, 1, 2$ to obtain the mass spectrum using Eq. (4.19). We plot our results separated by ℓ in Fig. 5 alongside experimental results from [6, 7]. Despite our simplified potential and non-relativistic approach, our computational results line up reasonably well with the true mass spectrum of observed charmonium states from particle physics data. We could further refine our mass spectrum by including higher order effects into our potential or doing a hyper parameter scan of m_c of α_s to find optimal values to minimize our difference from experimental data, but these ideas would bring us into the realm of ongoing research, far beyond the scope of this project.

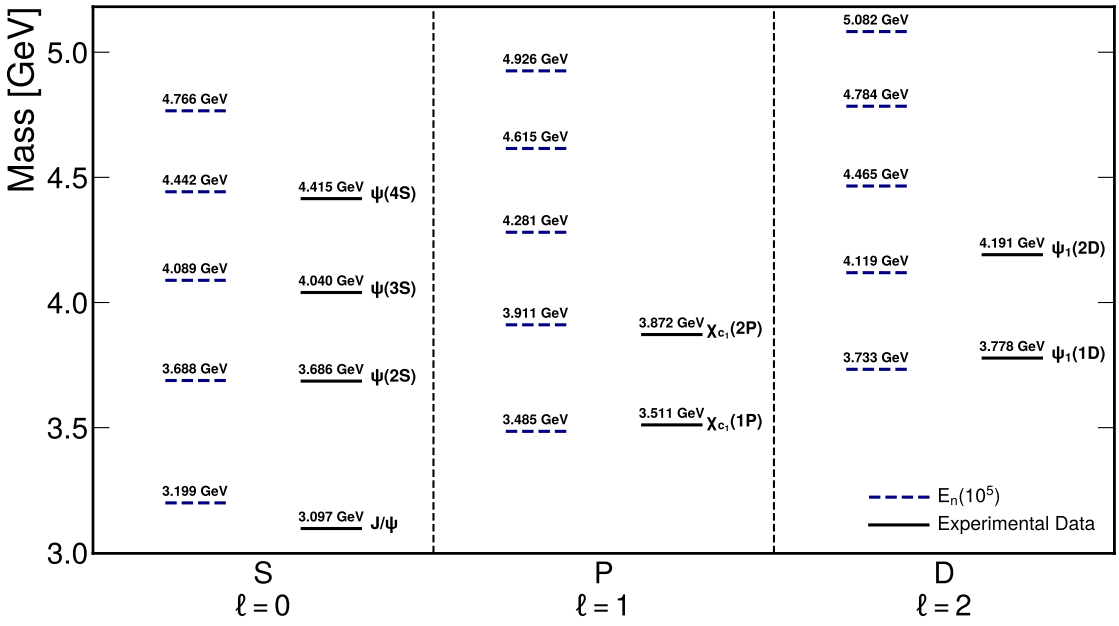

Figure 5: Mass spectrum for bound $c\bar{c}$ states (Charmonium), comparing our computational results using Eqs. (4.18) and (4.19) to experimental results from [6]. Particle names shown next to experimental values and the S, P, D lettering for orbitals follow the conventions used in [7].

V Conclusions

Our paper has presented the key ideas behind a numerical approach to finding the eigenstates and eigenvalues of single dimensional quantum systems by discretizing to a standard matrix and eigenvalue expression. We verified the accuracy and capability of our algorithm using standard potentials with well-defined exact solutions, in which we observed that our numerical method converged well on the exact results. We were also able to successfully recover qualitatively accurate results for the double well potential despite the absence of exact analytic solutions, and demonstrated the feasibility of applying our method to active research areas of particle physics with an approximate recreation of the charmonium spectrum.

When considering future possible progression of this project, there are an infinite amount of different one dimensional potentials with varying levels of usefulness to which we can consider applying our method. Additionally, there are multiple different paths we can consider for modify our approach to each of the systems considered in this paper to further refine our results or to describe more sophisticated variations of these systems. Within the specific focus of numerical analysis, we can do more to investigate the convergence and stability of our method, which we observed in Section III.1 to break down at the scale of $\sim 10^6$ mesh points. An initial next step would be observing if this breakdown still occurs at the same point when using more precise floating point numbers, such as 128 bit floats which are supported within the `numpy` and `scipy` libraries. However, this would require us solve eigensystems of matrices containing 128 bit floating point numbers on the scale of $10^6 \times 10^6$ and beyond, which would quickly depart from the "desktop computing scale" established for the scope of this project.

Acknowledgements

Many thanks to Professor Trefethen, the course teaching fellows, and the rest of the teaching staff for their help and support throughout the term and on this project. AMDG.

The relevant Python code for this project can be found below in Appendix B. Additionally the specific notebook used to produce the figures and results in this paper can be found on Github.

References

[1] M.T. Heath. *Scientific Computing: An Introductory Survey.* McGraw-Hill Education, 2005.

[2] Lloyd N. Trefethen, Asgeir Birkisson, and Tobin A. Driscoll. *Exploring ODEs.* SIAM-Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2017.

[3] P. Virtanen and Others. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[4] J. J. Sakurai. *Modern Quantum Mechanics; Rev. Ed.* Addison-Wesley, Reading, MA, 1994.

[5] E. Eichten, K. Gottfried, T. Kinoshita, K. D. Lane, and T. M. Yan. Charmonium: Comparison with Experiment. *Phys. Rev. D*, 21:203, 1980.

[6] R. L. Workman and Others. Review of Particle Physics. *PTEP*, 2022:083C01, 2022.

[7] Wei-Jun Deng, Hui Liu, Long-Cheng Gui, and Xian-Hui Zhong. Charmonium spectrum and electromagnetic transitions with higher multipole contributions. *Physical Review D*, 95(3), February 2017.

A Harmonic Oscillator Eigenstates

Here we present the convergence of our numerically obtained eigenstates for the harmonic oscillator from Section III.1 to the exact results from Eq. (3.10). We omitted these results from the main paper since there are a few tangential complications to be addressed.

To begin, the eigenstates obtained from our method are sometimes off by a negative sign compared to $\psi_n(x)$ from Eq. (3.10). This isn't necessarily problematic for two reasons: from an eigenvector perspective we understand that a solution eigenvector multiplied by a scalar is still a solution eigenvector, from a quantum mechanics perspective our states obey a $U(1)$ symmetry and are invariant under multiplications of $e^{i\alpha}$ for an arbitrary scalar α. We avoid this by plotting the absolute value of our wavefunction $|\psi_n(x)|^2$ rather than $\psi_n(x)$ itself. This still carries physical meaning as the probability density function of finding the particle within a region dx and solves the issue of qualitatively comparing our numerical results with the exact results.

To plot the quantitative error between the exact and numerical solutions we plot the square of the difference, $|\psi_n(x) - \psi_n(x, N)|^2$, but we need to account for the fact that $\psi_n(x, N)$ only has N different points to be compared at. As a result, for this comparison we compute the "exact" result using the exact formula Eq. (3.10) but only for the same N mesh points as the numerical result they will be compared to. The end result of these comparisons is shown for $n = 2$ in Fig. 6.

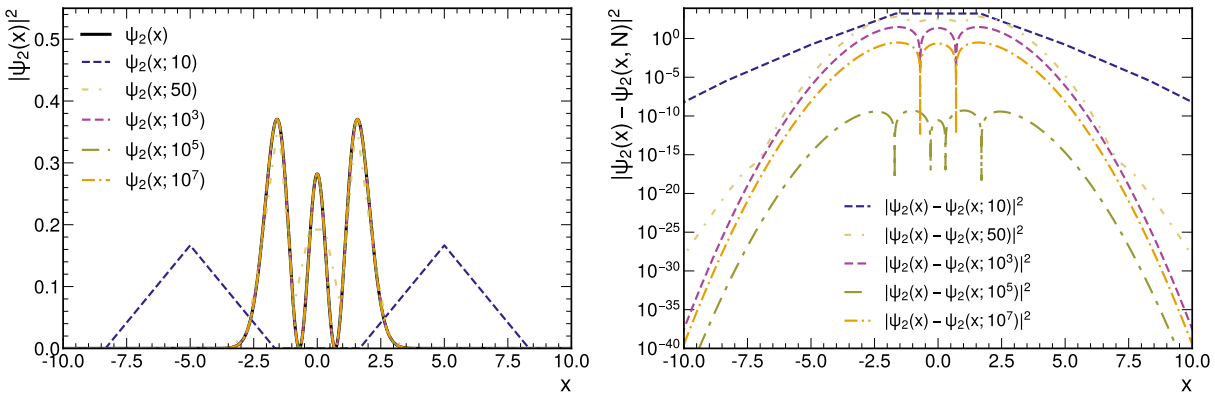

Figure 6: Square of the third level eigenstate for the harmonic oscillator with exact eigenstates compared to numerical results for varying levels of discretization.

In general we observe the same behaviors that we observed with the eigenvalues in Fig. 3. The numerical eigenstates tend to converge well by about $N = 10^3$, but we observe the same divergence begin to occur for values beyond $N = 10^5$, linked to the stability issues from compounding machine error that we discussed in Section III.1. This further solidified our reasons to continue with $N = 10^5$ for later results in this paper.

B Attached Code

The core functionality of the numerical method established in Section II is the following function:

```
import numpy as np
import scipy as sp

def discretized_schrodinger(potentialFunction,
                            initialBoundaryPoint: float, finalBoundaryPoint: float,
                            numPosSteps: int, numEigVals: int,
                            mass: float, hbar = 1.0):
# Calculate discretized points and spacing for [a,a+dx,a+2dx,...,b-dx,b]
xs = np.linspace(initialBoundaryPoint,finalBoundaryPoint,numPosSteps+2)
dx = np.diff(xs)[0]

# Compute potential along discretized points
Vx = potentialFunction(xs)

# Calculate main diagonal, trimming out end points where psi(a)=psi(b)=0
main_diag = (np.power(dx,-2.0) + mass * Vx)[1:-1]
# Calculate sub/supradiagonal terms, constants defined by dx
off_diag  = np.power(dx,-2.0) * -0.5 * np.ones(len(main_diag)-1)

# Solve eigensystem using assumption matrix is symmetric tridiagonal,
# only return first numEigvals to cut down on unnecessary computation
evals, estates = sp.linalg.eigh_tridiagonal(
  d            = main_diag,
  e            = off_diag,
  eigvals_only = False,
  select       = 'i',
  select_range = (0,numEigVals-1)
)

return (hbar**2/mass)*evals, estates.T
```