-5 for 1 problem late
50.5+55+5-5=105.5

105.5/117

Phys 304: Homework 6

Mary Smith*
Haverford College Department of Physics
(Dated: March 25, 2024)

[In this problem set, I learned how to code linear algebra and got extremely familiar with typesetting equations in Latex. This problem set took me about 7 hours. My collaborators were Luke Smithberg, Melanie Santiago, and Claire Jones.]
✓+5

1. PROBLEM 1: ASYMMETRIC QUANTUM WELL

Suppose there is a particle of mass M in a one-dimensional, non-square quantum well of width L. The potential V(x) varies somehow inside the well. In a pure state of energy E, the spatial part $\psi(x)$ of the wavefunction obeys the time-independent Schrödinger equation $\hat{H}\psi(x) = E\psi(x)$, where the Hamiltonian operator $\hat{H}$ is given by

$$\hat{H} = -\frac{\hbar^2}{2M}\frac{d^2}{dx^2} + V(x). \tag{1}$$

Assuming that the walls of the well are infinitely high, so that the wavefunction is zero outside the well, this means it must go to zero at $x = 0$ and $x = L$. This means that the wavefunction can be expressed as a Fourier sine series

$$\psi(x) = \sum_{n=1}^{\infty} \psi_n \sin(\frac{\pi n x}{L}), \tag{2}$$

where $\psi_1, \psi_2, ...$ are the Fourier coefficients.
The pseudocode for this section can be found in Figure 1.

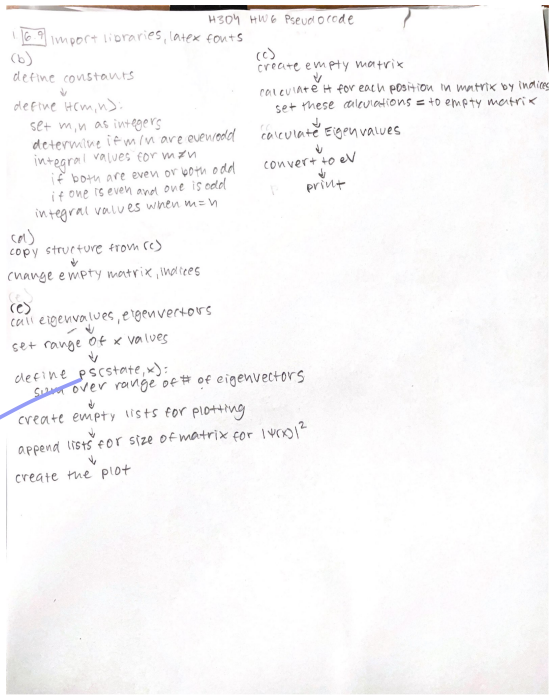

FIG. 1: [The pseudocode for the asymmetric quantum well.]

1.1. Part (a): Schrodinger's equation in matrix form

Starting with the definition of the time-independent Schrödinger equation

$$\hat{H}\psi(x) = E\psi(x), \tag{3}$$

the Fourier sine series in Equation 2 can be plugged in for $\psi(x)$. This gives

$$\sum_{n=1}^{\infty} \psi_n \hat{H} \sin(\frac{\pi n x}{L}) = E\sum_{n=1}^{\infty} \psi_n \sin(\frac{\pi n x}{L}) \tag{4}$$

The Hamiltonian operator $\hat{H}$ can be moved inside of the summation, because the Hamiltonian operator is spatially dependent and the Fourier coefficient is not spatially dependent. This expression needs to be evaluated analytically, and in order to do so, the sine function must be dealt with. To do this, $\sin(\frac{m\pi x}{L}dx$ was multiplied to both sides. Then an integral was calculated:

$$\sum_{n=1}^{\infty} \psi_n \int_0^L \sin(\frac{m\pi x}{L})\psi_n \hat{H} \sin(\frac{\pi n x}{L})dx = E\sum_{n=1}^{\infty} \psi_n \int_0^L \sin(\frac{m\pi x}{L}) \tag{5}$$

Noting that, for m, n positive integers:

$$\int_0^L \sin(\frac{m\pi x}{L})\psi_n \sin(\frac{\pi n x}{L})dx = \left\{ \begin{array}{ll} L/2 & \text{if } m = n \\ 0 & \text{otherwise.} \end{array} \right\} \tag{6}$$

This means that Equation 5 can be rewritten as

*Electronic address: masmith@haverford.edu

$$\sum_{n=1}^{\infty}\psi_n\int_0^L \sin(\frac{m\pi x}{L})\psi_n\hat{H}\sin(\frac{\pi n x}{L})dx = E\sum_{n=1}^{\infty}\psi_n\frac{L}{2}. \tag{7}$$

Using the fact that matrix multiplication is defined as

$$(AC)_{j,k} = \sum_{r=1}^{n} A_{j,r}C_{r,k}, \tag{8}$$

Equation 7 can then be written as

$$\sum_{n=1}^{\infty}\psi_n\int_0^L \sin(\frac{m\pi x}{L})\psi_n\hat{H}\sin(\frac{\pi n x}{L})dx = \frac{1}{2}LE\psi_m. \tag{9}$$

Hence, this defines a matrix **H** with elements

$$H_{mn} = \frac{2}{L}\int_0^L \sin(\frac{m\pi x}{L})(-\frac{\hbar^2}{2M}\frac{d^2}{dx^2} + V(x))\sin(\frac{n\pi x}{L})dx. \tag{10}$$

This means that Schrödinger's equation can be written in matrix form $\mathbf{H}\psi = E\psi$, where ψ is the vector $(\psi_1, \psi_2, ...)$. Thus ψ is an eigenvector of the Hamiltonian matrix **H** with eigenvalue E. If the eigenvalues of this matrix are calculated, the allowed energies of the particle in the well can be calculated.

1.2. Part (b): Expression for H_{mn} for arbitrary m and n

For the case $V(x) = ax/L$, the integral in H_{mn} can be evaluated analytically and thus a general expression for the matrix element H_{mn}. Plugging in $V(x) = ax/L$ into Equation 9, separating the integral, and using the fact below:

$$\int_0^L x\sin(\frac{m\pi x}{L})\sin(\frac{n\pi x}{L})dx =$$
$$\left\{\begin{array}{ll} 0 & m\neq n, \text{ both even or both odd} \\ -\frac{2L}{\pi}^2\frac{mn}{(m^2-n^2)^2} & m\neq n, \text{ one even, one odd} \\ L^2/4 & \text{if } m = n, \end{array}\right\} \tag{11}$$

a general expression for H_{mn} can be built analytically. This general expression is below:

show steps

$$H_{mn} = \left\{\begin{array}{ll} \frac{\hbar^2 n^2\pi^2}{2ML^2} + \frac{a}{2} & \text{if } m = n \\ -\frac{8amn}{\pi^2(m^2-n^2)^2} & m\neq n, \text{ one even, one odd} \\ 0 & m\neq n, \text{ both even or both odd.} \end{array}\right\} \tag{12}$$

This matrix is real because there are no imaginary parts to the general expression. Matrix symmetry is proven by if the matrix itself is equal to its transpose. The transpose is built by swapping the position indices in the matrix. For this general expression Equation 12, swapping the indices will just involve replacing every m with an n and replacing every n with an m. This does not change the values of the matrix, and so H_{mn} is symmetric.

In order to evaluate Equation 12 for arbitrary m and n when the particle in the well is an electron, the pseudocode in Figure 1 was followed. Using Python, a function for $H(m, n)$ was created, with took integer values for m and n and calculated their remainders to determine if the numbers given were even or odd. Then, using the parameters given in Equation 12, an expression to find the value of H given different position indices was generated.

1.3. Part (c): The 10x10 array for H

The matrix **H** is in theory infinitely large, so all of its eigenvalues cannot be calculated. The code in Section 1.1 can be modified to create a 10 x 10 array of the elements of **H** up to $m, n = 10$. The eigenvalues of this matrix can be used to find the first ten energy levels of the quantum well within this approximation. The pseudocode in Figure 1 was followed in order to generate Python code for this section. An empty 10 x 10 matrix was created and then filled for the position indices m, n by applying the function for $H(M, n)$. The eigenvalues were then calculated and then converted to eV. The first ten energy levels of the quantum well are, in units of eV: 5.837, 11.18, 18.66, 29.15, 42.66, 59.19, 78.74, 101.3, 126.9, and 155.6.

1.4. Part (d): The 100x100 array for H

The code in Section 1.3 can be modified according to the pseudocode in Figure 1 in order to use a 100 x 100 array instead and again calculate the first ten energy eigenvalues. This was done by changing the size of the empty matrix and range for position indices to 100. The first ten energy levels of the quantum well for a 100 x 100 matrix are, in units of eV: 5.837, 11.18, 18.66, 29.15, 42.66, 59.19, 78.74, 101.3, 126.9, and 155.4. Compared with the values obtained in Section 1.3, it can be concluded that the accuracy of the calculation increases as the size of the matrix increases.

1.5. Part (e): Plotting for different states

The code in Section 1.4 can once again be modified to calculate the wavefunction $\psi(x)$ for the ground state and the first two excited states of the well. Using these results, a graph can be generated with three curves showing the probability density $|\psi(x)|^2$ as a function of x in each

of these three states. Following the pseudocode in Figure 1, the above can be done using Python code. A wavefunction $\psi(x)$ was generated, using Equation 2. Empty lists were then created and appended for the range of values of the matrix **H** for each of the energy states. This creates the graph in Figure 2.

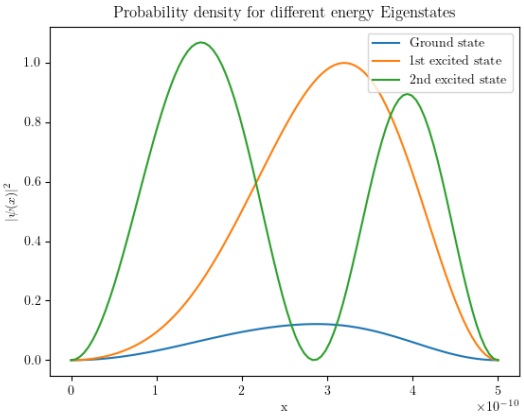

FIG. 2: [The probability density as a function of position for different energy Eigenstates. Note how the number of nodes change as the particle enters higher states. The asymmetric behavior of the graphs are because of the variance of the potential $V(x)$ inside the well.]

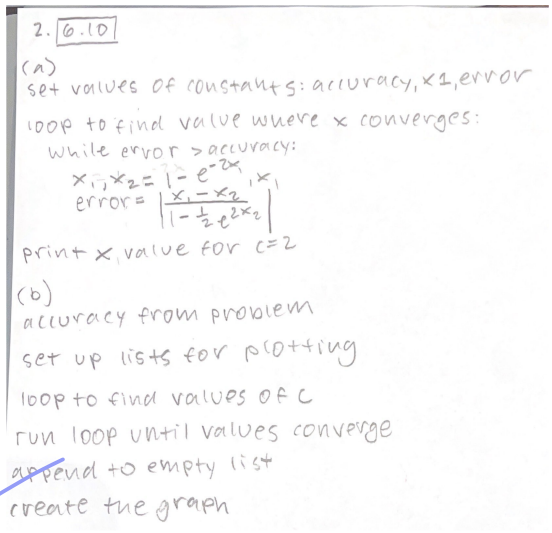

FIG. 3: [The pseudocode for using the relaxation method to solve Equation 13.]

2. PROBLEM 2: THE RELAXATION METHOD

Consider the equation

$$x = 1 - e^{-cx}, \tag{13}$$

where c is a known parameter and x is unknown. This form of equation appears in a variety of processes, including the phsycis of contact processes, mathematical models of epidemics, and the theory of random graphs. The pseudocode for this section can be found in Figure 3.

2.1. Part (a): Solving for a specific case

A program was created to solve Equation 13 using the relaxation method for the case where c=2, to an accuracy of at least 10^{-6}. In order to do this, the method outlined in Figure 3 was used. A while loop to find the value where x converges for all values of error¿accuracy. The error was determined using the equation

$$\epsilon' = \frac{x - x'}{1 - 1/f'(x)}. \tag{14}$$

For the value c=2, the error can be determined to be $\epsilon' = \frac{1}{2}e^{2x}$. Using this process for when c=2, x=0.7968195968986895.

2.2. Part (b): Plotting x as a function of c

The program from above was modified to calculate the solution for values of c from 0 to 3 in steps of 0.01, in order to make a plot of x as a function of c. This is an example of a phase transition, known in physics as the percolation transition.

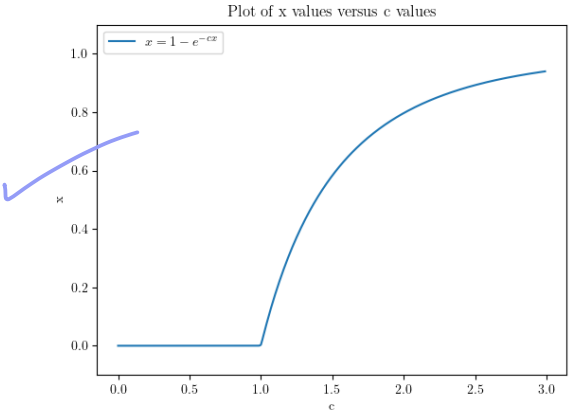

FIG. 4: [The plot of x as a function of c for the Equation 13. There is a clear transition from a regime in which x=0 to a regime of nonzero x.]

Using the method outlined in Figure 3, the plot in Figure 4 was generated. A loop was used to append an empty list for each value of x and c until the error in Equation 13 reaches the accuracy of 10^{-6}.

6.9 50.5/56

Computational Physics/Astrophysics, Winter 2024:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 56 points will be available per problem. Partial credit available on all non-1 items.

4 1. Does the program complete without crashing in a reasonable time frame? (+4 points)

1 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? (+2 points) printed answers need a description, such as "evals for 10x10 matrix" -1

2.5 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) (+3 points) only needed first 10 evals -0.5 are ..."

5 4. Is the algorithm appropriate for the problem? If a specific algorithm was requested in the prompt, was it used? (+5 points)

4 5. If relevant, were proper parameters/choices made for a numerically converged answer? (+4 points)

3 6. Is the output answer correct? (+4 points). -1 an indexing error (should be j-1 not j) and missing normalization

3 7. Is the code readable? (+3 points)

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

---

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

3 8. Is the code well documented? (+3 points)

. 6.1. Is the code author named?

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

9. Write-up (up to 28 points)

5 . Is the problem-solving approach clearly indicated through a flow-chart, pseudo-code, or other appropriate schematic? (+5 points)

✓. Is a clear, legible LaTeX type-set write up handed in?

1 . Are key figures and numbers from the problem given? (+ 3 points) numerically define variables and show steps for p1.(b) -2

4 . Do figures and or tables have captions/legends/units clearly indicated. (+ 4 points)

3 . Do figures have a sufficient number of points to infer the claimed/desired trends? (+ 3 points)

2 . Is a brief explanation of physical context given? (+2 points)

0 . If relevant, are helpful analytic scalings or known solutions given? (+1 point) compare to 5.84 eV -1

3 . Is the algorithm used explicitly stated and justified? (+3 points)

2 . When relevant, are numerical errors/convergence justified/shown/explained? (+2 points)

2. Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (+2 points)
1. Are collaborators clearly acknowledged? (+1 point)
2. Are any outside references appropriately cited? (+2 point)

6.10 55/56

Computational Physics/Astrophysics, Winter 2024:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 56 points will be available per problem. Partial credit available on all non-1 items.

4 1. Does the program complete without crashing in a reasonable time frame? (+4 points)

1 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? (+2 points) printed results need a description, such as "for c=2, x=..." -1

3 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) (+3 points)

5 4. Is the algorithm appropriate for the problem? If a specific algorithm was requested in the prompt, was it used? (+5 points)

4 5. If relevant, were proper parameters/choices made for a numerically converged answer? (+4 points)

4 6. Is the output answer correct? (+4 points).

3 7. Is the code readable? (+3 points)

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

---

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

3 8. Is the code well documented? (+3 points)

. 6.1. Is the code author named?

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

9. Write-up (up to 28 points)

5 . Is the problem-solving approach clearly indicated through a flow-chart, pseudo-code, or other appropriate schematic? (+5 points)

✓ . Is a clear, legible LaTeX type-set write up handed in?

3 . Are key figures and numbers from the problem given? (+ 3 points)

4 . Do figures and or tables have captions/legends/units clearly indicated. (+ 4 points)

3 . Do figures have a sufficient number of points to infer the claimed/desired trends? (+ 3 points)

2 . Is a brief explanation of physical context given? (+2 points)

1 . If relevant, are helpful analytic scalings or known solutions given? (+1 point)

3 . Is the algorithm used explicitly stated and justified? (+3 points)

2 . When relevant, are numerical errors/convergence justified/shown/explained? (+2 points)

2. Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (+2 points)
1. Are collaborators clearly acknowledged? (+1 point)
2. Are any outside references appropriately cited? (+2 point)