PHY407: Lab 4

Sky Kapoor and Chris Compierchio

October 7th, 2022

Work Distribution: We worked on this assignment together in-person, so it was split quite evenly. We brainstormed pseudocodes, and then the base code for each question was collaborated upon. We reused the code we wrote as necessary to answer all of the questions. We alternated between parts for these questions, and then switched and checked each other's work at the end.

*All Python code and outputs are included in the Quercus submission as .py files.

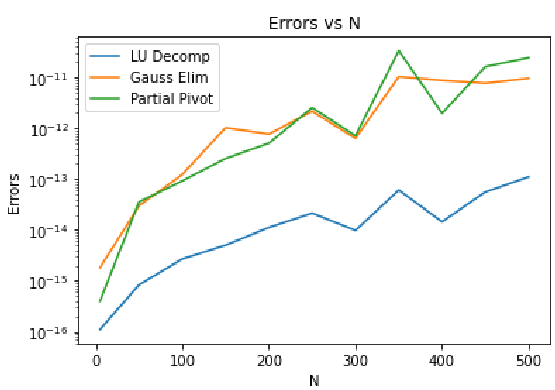
Question 1

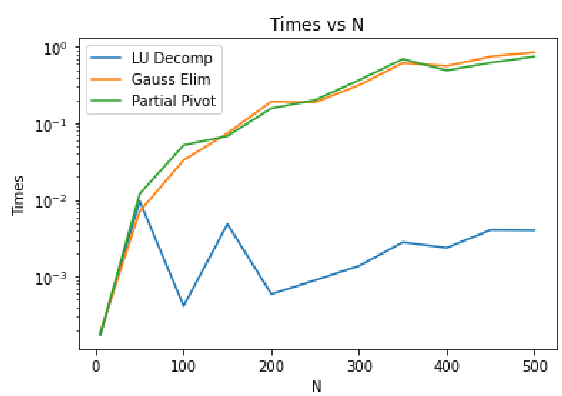
a) Printout

The Python code for part a produces the result: [2. -1. -2. 1.], which agrees with the values given in the textbook.

b) Plots, written answers, pseudocode, code

Plots





Written answers

When values of N become greater, the accuracy of the Gaussian elimination and partial pivot methods are greater than that of the LU decomposition method. However, from the plots above, we can also see that the LU decomposition method takes significantly less time for the greater values of N in comparison to the other two methods.

Pseudocode

1. Import Libraries
2. Define Partial Pivot function
3. Define GaussElim function
4. Define N
5. Define the array from the textbook example
6. Define the vectr from the textbook example
7. Print results
8. Define values
9. Create arrays to store the times for each method
10. Create arrays to store the errors for each method
11. Loop through N
12. Create a random vector and matrix
13. Time LU Decomposition
14. Time Gaussian Elimination
15. Time Partial Pivot
16. Solve method each using np.dot
17. Calculate the errors of each method
18. Plot errors against N
19. Plot times against N

Question 2

a) Nothing to Submit

b) Nothing to Submit

c) Code and printed outputs

Printed output for the code in part c: [5.83642783 11.18115866 18.66294388 29.14422969 42.65508159 59.18523429 78.72930115 101.28538399 126.85123972 155.55513523].

These are the first 10 eigenvalues in electron volts (eV).

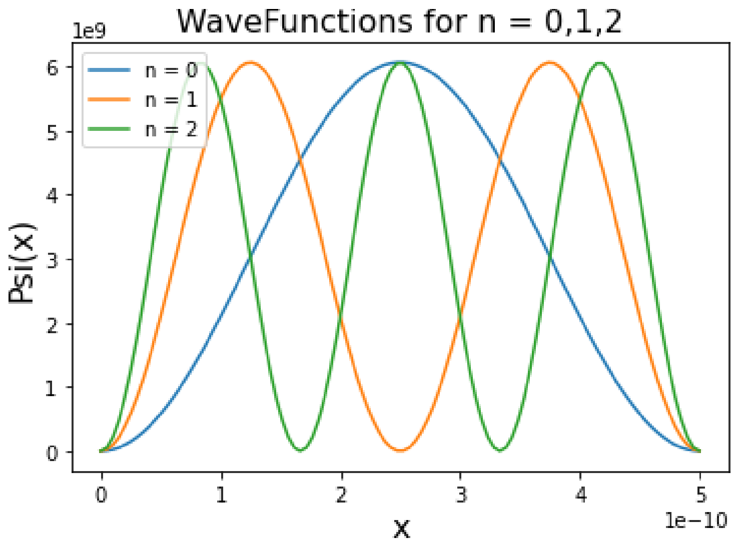
d) Printed outputs and written answer

The printed output for the code in part d produces 100 values, so the first 10 are shown here: [5.83642743e+00 1.11811573e+01 1.86629420e+01 2.91442209e+01 4.26550725e+01 5.91851817e+01 7.87292493e+01 1.01284753e+02 1.26850407e+02 1.55425509e+02].

These values are mostly similar to the values produced from a 10x10 matrix. There are some slight differences, but the values seem to be accurate up to (at the very least) 3 significant figures.

e) Graph, pseudocode, code

Graph



Pseudocode

1.Import libraries
2. Define given constants
3. Define Hmn function
4. Define function to integrate sing simpsons rule as in Lab 2
5. Define maximum number of indices for the H matrix
6. Define H
7. Calculate each entry of H
8. Calculate the eigenvalues of H
9. Print results in eV
10. Repeat for a 100x100 matrix
11. Calcualte the eigenvalues and eugenvectors for H
12. Define position points for the particle
13. Loop for n up to 3
14. Define psi
15. Calculate psi using the eigenvectors of H
16. Integrate psi
17. Normalize psi
18. Plot results

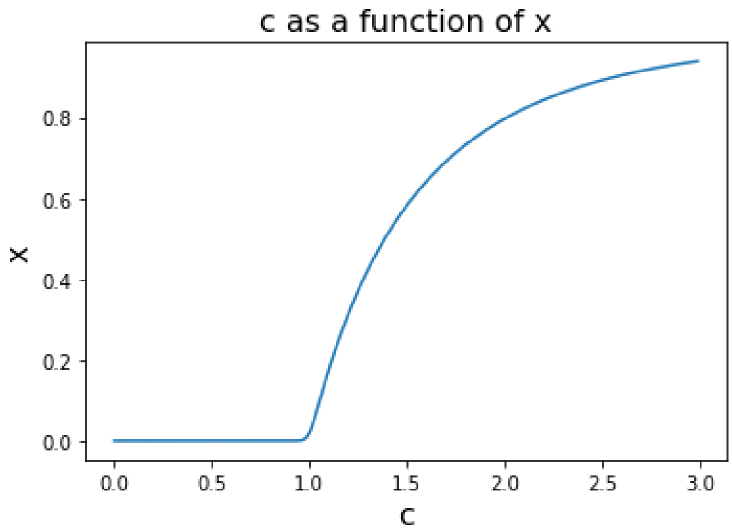
Question 3

a) Pseudocode, code, plot

Pseudocode

1. Define x and c
2. Calculate x for an accuracy of 10e-6
3. Repeat but plot for C = 0-3

Plot



b) Printouts, pseudocode, written answer

Printouts

Printout for part b (number of iterations): 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

Printout for part c (number of iterations + result):
0 0.7232837444202624
1 0.7931473623659533
2 0.7968147025704155
3 0.7968121217190013
4 0.7968121300467907
5 0.7968121300199338
6 0.7968121300200203
7 0.79681213002002
8 0.79681213002002
9 0.79681213002002

Pseudocode

(part c)
1. Define constants
2. Use a for loop with the overrelaxtion equation in Ex. 6.11 to find x up until an accuracy of 10e-6
3. print iterations and x

Written answer

Using the overrelaxation method, it took about 3 iterations to converge to the desired value. This is signficantly better than the relaxation method, which took around 15 iterations. The value of ω

that required the least number of iterations was found to be 0.68 when the value of $x_0 = 0.5$.

We can use values of $\omega < 0$ for cases where the value of $f' < 0$ as well.

c) Pseudocode and written answers

Pseudocode

1. Define f
2. Define Constants
3. Perform binary search algorithm described in textbook page 264
4. print results
5. Calculate b
6. Calculate and print the surface temperature of the sun

Written answer

Using the binary search method, the value of x was found to be 4.96511, and it took 14 iterations to get this value. Using the equations provided in the textbook and the calculation of the Wien displacement constant (which was found to be 0.00289), the temperature of the Sun was calculated to be 5772.45 K, which is accurate.