

PHY 180 - Computational Physics - Spring 2023

Project 8: Bound States by Diagonalization

Due: Monday, April 17th

Trevin Detwiler

1 Introduction

The Time-Independent Schrödinger Equation (TISE) is notoriously difficult to solve in general. Even for the potentials that have analytical solutions, obtaining these solutions come with smart tricks using operators or series. Alternatively, these problems can be solved computationally. This involves solving the TISE “in a box” by diagonalizing the matrix for the Hamiltonian at each point in space.

This project will look at the harmonic oscillator in 1D Cartesian coordinates and 3D spherical coordinates, as well as Coulomb problem for an atom with Z protons and one electron. These potentials simplify the diagonalization process immensely since their Hamiltonian matrices are tridiagonal. We will also implement a variable lattice mesh in the 3D case for better resolution of the wavefunctions $u_{nl}(r)$. The spacial step size in the program is small when r is small and large when r is large, with an increasing step size for the part in between the two.

2 Modifications to the Code

To start, the `diag-box-demo.f90` file was split into three files so to handle the three problems separately. While the 1D harmonic oscillator could have been handled by setting `lmax = 0`, I opted to remove the variables `lmax` and `il` and rewrite the bounds of the `do` loops. The potential was set to `0.5*(r(i) - npts*dx/2)**2`, which is the harmonic oscillator potential centered in the middle of the box. For the output file, the wavefunction check was adjusted for Cartesian coordinates so that it starts out positive when n is even and negative when n is odd.

Very few changes are needed for the 3d harmonic oscillator and the Coulomb problem. The potential is `0.5*r(i)**2` for the former and `-Z_p/r(i)` for the latter, where `Z_p` is the number of protons read in from the user. The wavefunctions produced in these programs are $u_{nl}(r) = rR_{nl}(r)$.

Finally, the variable mesh is introduced into the two programs using spherical coordinates. This uses code from the `hint1.f90` and `hint2.f90` files to handle the spacial step size, Hamiltonian matrix, and norms appropriately. I also opted to read in the box size `rmax` instead of the number of points `npts`. The `calc_npts` subroutine is then used to calculate `npts` and the size of the final spacial step.

3 Results

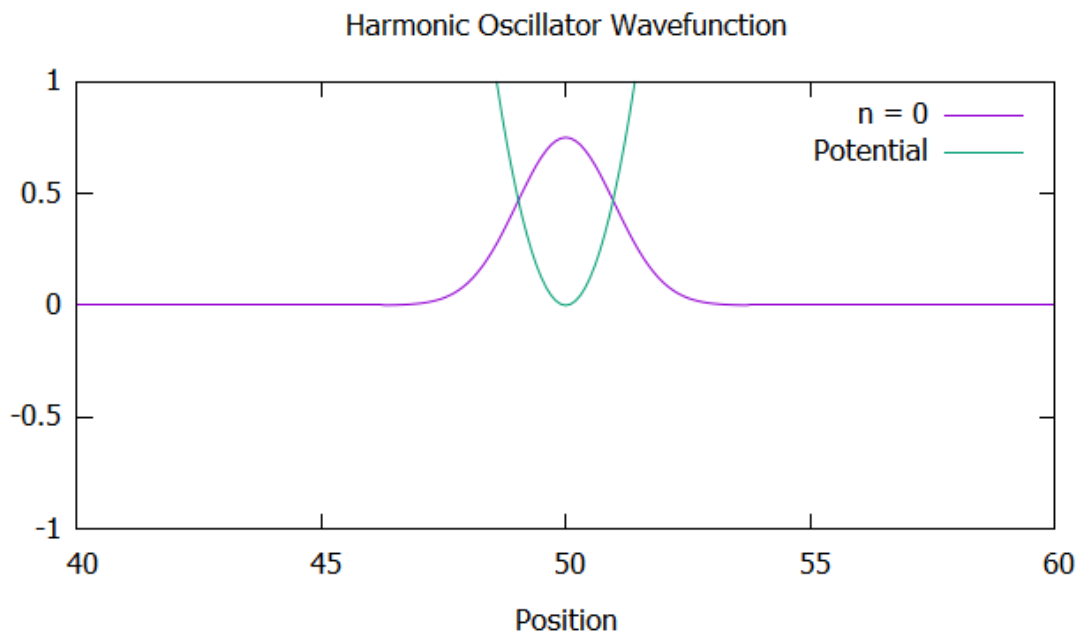


Figure 1

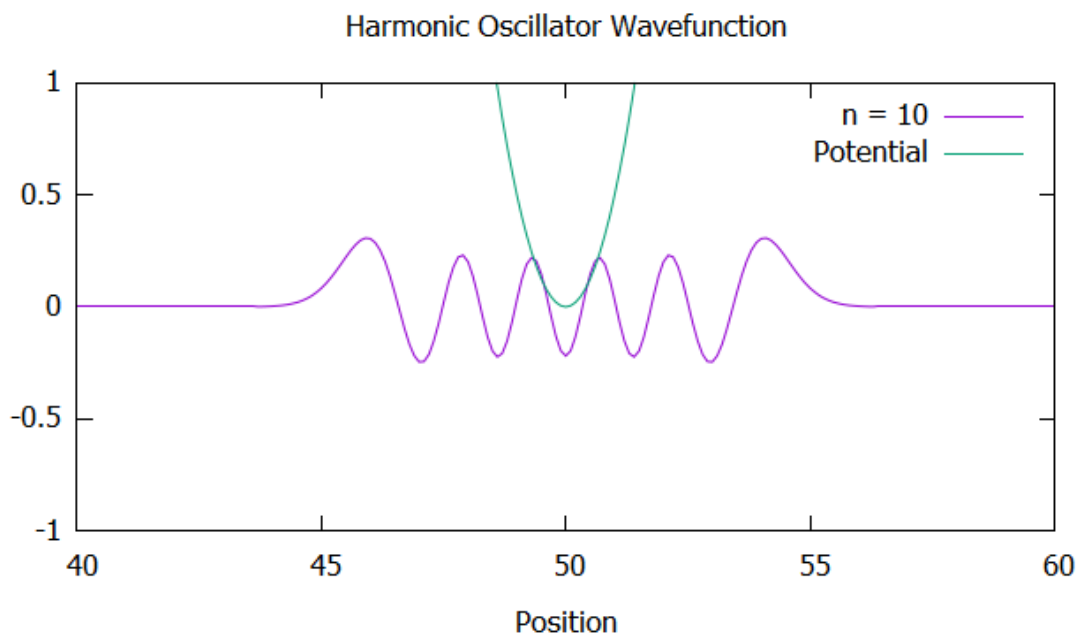


Figure 2

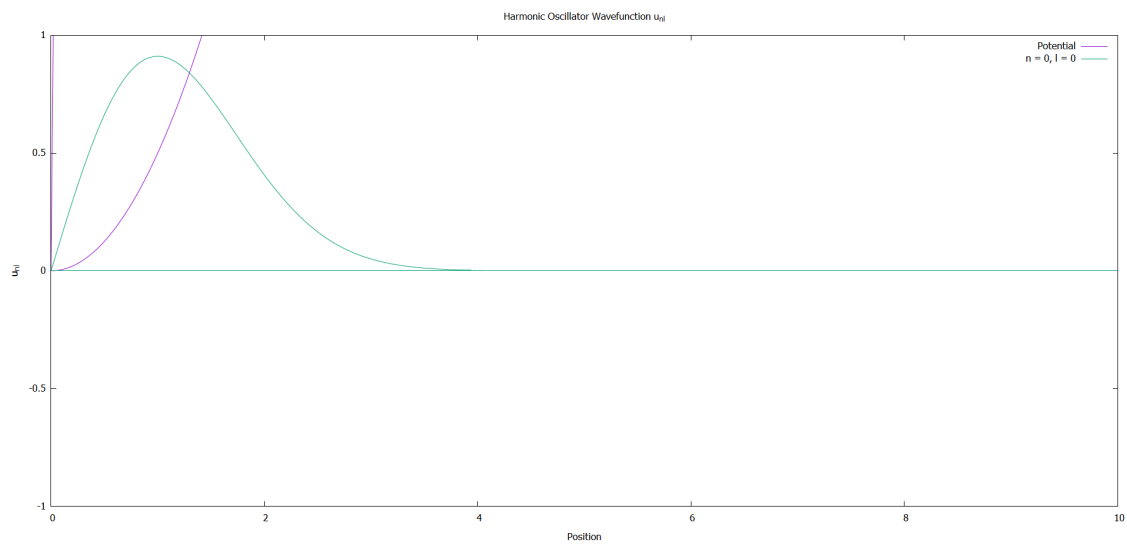


Figure 3

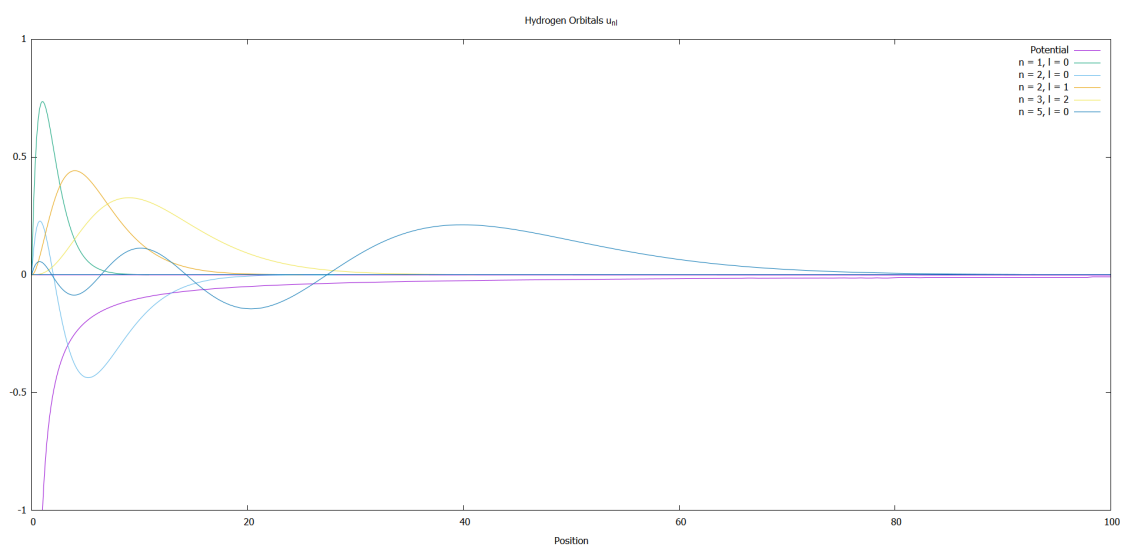


Figure 4

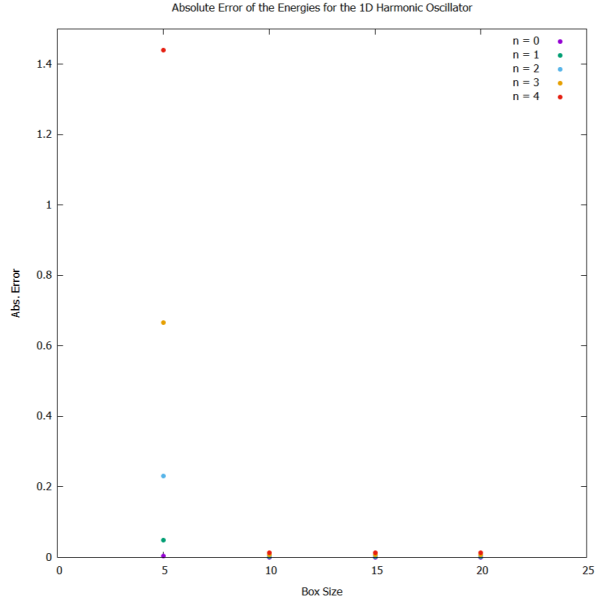


Figure 5: Absolute difference between the computed energies and the theoretical energies of the 1D harmonic oscillator for various box sizes.

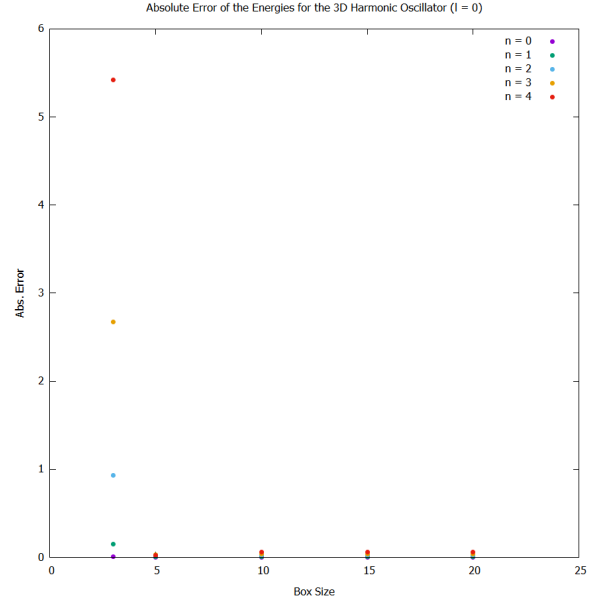


Figure 6: Absolute difference between the computed energies and the theoretical energies of the 3D harmonic oscillator s-orbitals for various box sizes.

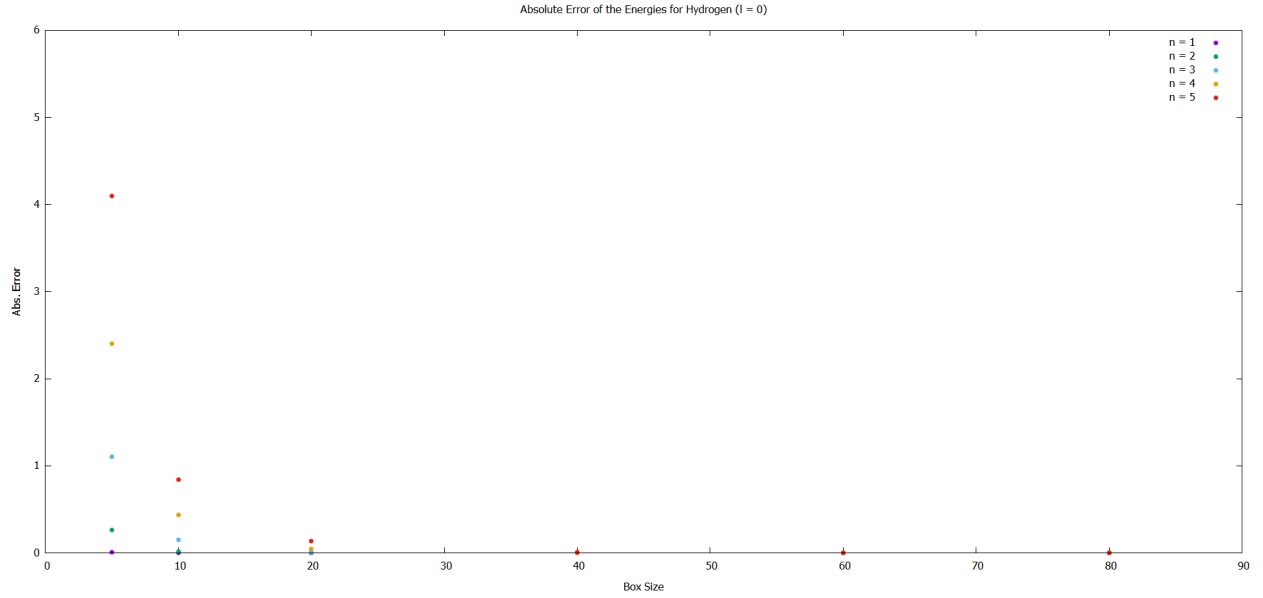


Figure 7: Absolute difference between the computed energies and the theoretical energies of the hydrogen atom s-orbitals for various box sizes.

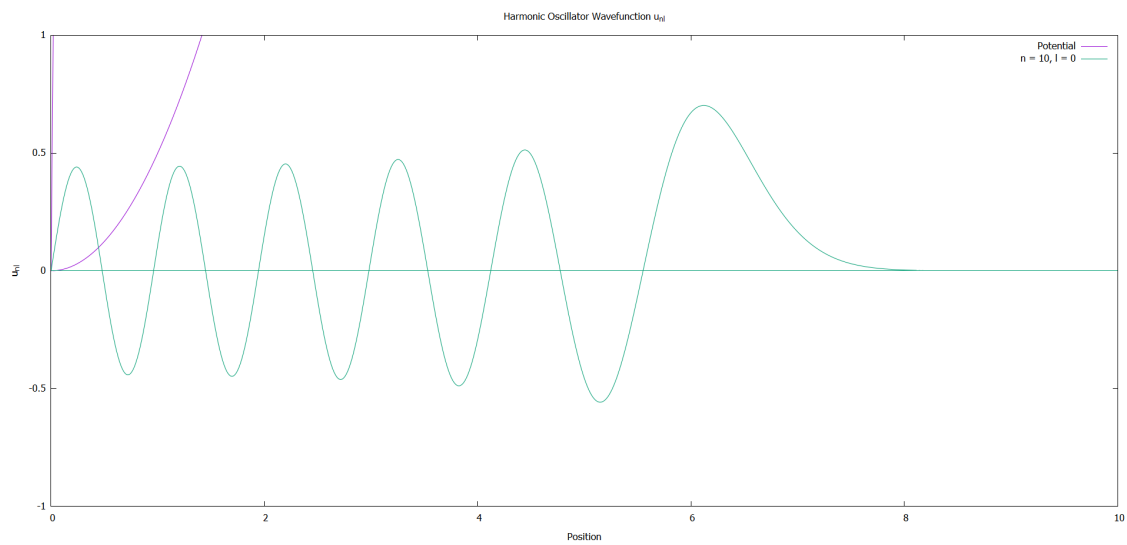


Figure 8

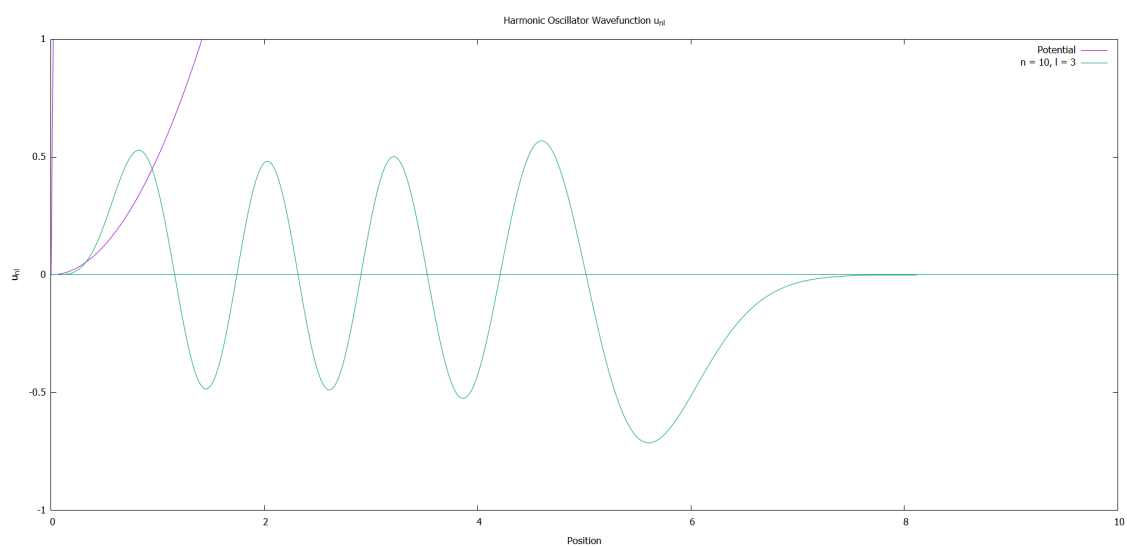


Figure 9

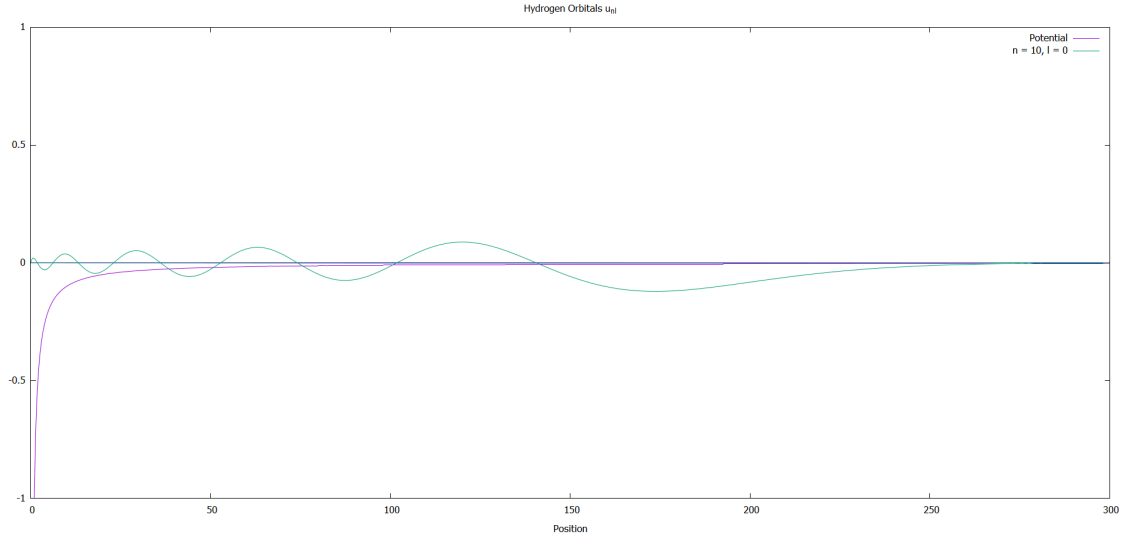


Figure 10

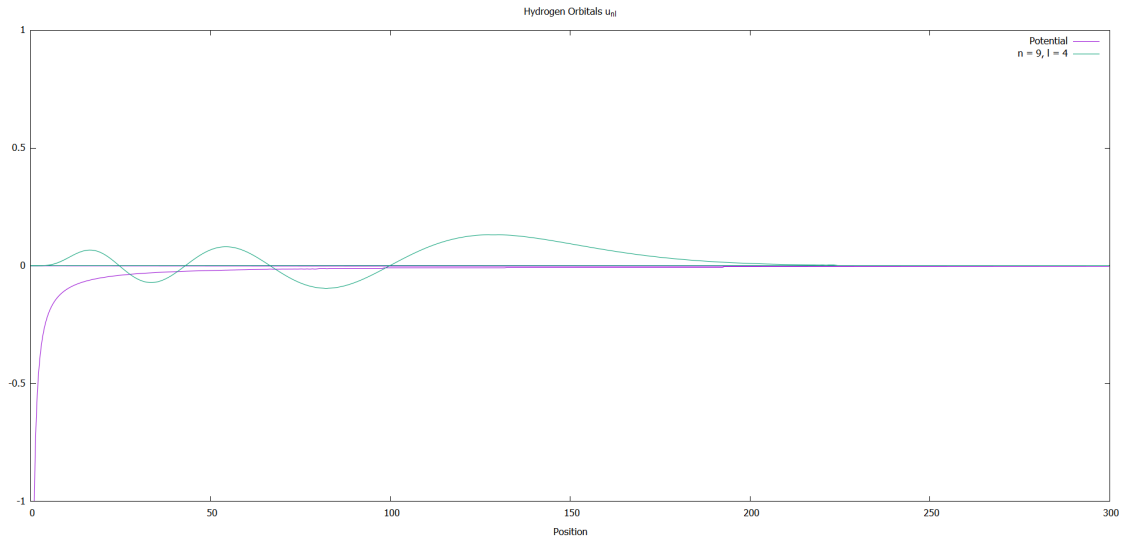


Figure 11

4 Conclusions

The programs are able to produce plots of the wavefunctions accurately, as shown in Figures 1 – 4. The eigenvalues for higher energy states are more accurate when calculations are performed in a bigger box, as shown in Figures 5 – 7. However, if only the first few energies are needed, then a smaller box may suffice. Finally, by using a variable spacial step size, we are able to produce plots of the orbitals for high n , as seen in Figures 8 – 11.