Solving the 1-D Schrodinger equation using finite differences

Considering the one-dimensional, time-independent Schrodinger equation for a particle in a potential $V(x)$,

$$-\frac{\hbar^2}{2m}\frac{d^2\psi}{dx^2} + V(x)\psi = E\psi, \tag{1}$$

we can define two potential wells for which to find eigenstates,

$$V_1(x) = 50\,\mathrm{eV\,nm^{-2}}x^2 + 2500\,\mathrm{eV\,nm^{-4}}x^4, \text{ and,} \tag{2}$$

$$V_1(x) = 50\,\mathrm{eV\,nm^{-2}}x^2 + 2500\,\mathrm{eV\,nm^{-4}}x^4 + 1500\,\mathrm{eV\,nm^{-3}}x^3. \tag{3}$$

We can also rewrite the equation for simplicity as,

$$-\frac{d^2\psi}{dx^2} + \frac{2m}{\hbar^2}V(x)\psi = \frac{2m}{\hbar^2}E\psi, \tag{4}$$

where $\hbar^2 = 0.076199682\,\mathrm{eV}\,m_e\,\mathrm{nm}^2$ in our system of units, and the mass is chosen to be $m = m_e$. We can define a modified energy term $E_m = E(2m/\hbar^2)$, and a modified potential $V_m(x_i)$, such that,

$$V_m(x_i) = \frac{2m}{\hbar^2}V(x_i). \tag{5}$$

The finite differences method can then be used to determine the energy eigenvalues and eigenstates of this system, such that $D\psi = E_m\psi$ for N discretized points (x_i, ψ_i) between boundary regions x_0 and x_1, where $\psi(x_0) = \psi(x_1) = 0$ by construction. In particular, the finite differences matrix D is an $N \times N$ sparse matrix, and can be expressed as,

$$D = \begin{pmatrix} \frac{2}{\hbar^2} + V_m(x_1) & -\frac{1}{\hbar^2} & 0 & 0 & \dots \\ -\frac{1}{\hbar^2} & \frac{2}{\hbar^2} + V_m(x_2) & -\frac{1}{\hbar^2} & 0 & \dots \\ 0 & -\frac{1}{\hbar^2} & \frac{2}{\hbar^2} + V_m(x_3) & -\frac{1}{\hbar^2} & \dots \\ \vdots & \vdots & \vdots & \ddots & \end{pmatrix}, \tag{6}$$

Note that, in this definition, the i terms in $\{x_i\}$ represent the i^{th} point on the grid, while x_0 and x_1 are general expressions for the leftmost and rightmost endpoints of the range.

This matrix can be diagonalized to obtain the energy eigenvalues of the system, with the corresponding eigenvectors representing the eigenfunctions evaluated over the space $\{x_i\}$. The initial parameters for x_0, x_1, and N were chosen to ensure accuracy in the energy eigenvalues to the fifth decimal place, and were taken to be $[x_0, x_1] = [-2.0, 2.0]$ and $N = 25000$ – and, by definition, imply a step size of $h = 1.6 \times 10^{-4}$. The resultant energy eigenvalues for the two potential wells are found in Table 1. It is clear that there is good agreement with the shooting method results used in the previous laboratory exercise. The eigenfunctions corresponding to each energy eigenvalue are given in Figure 1 for both potential wells.

In order to test whether the found solutions were robust against stricter choices of x_0, x_1, and N, these input parameters were modified by:

1. Decreasing the step size h by increasing N over the same range $[x_0, x_1]$
2. Increasing the range $[x_0, x_1]$, keeping the step size constant by increasing N proportionately

To this end, option 1.) was tested by extended by setting $N = 31250$, which reduces the step size h by 20% [1]. Likewise, option 2.) was tested by changing the range of the solution to $[-2.5, 2.5]$, but maintaining the same step size by increasing N to 31250. It was found in each case that the changes to the energy eigenvalues were below the fifth decimal place, indicating that the nominal choices of x_0, x_1, and N were sufficient to determine the energy eigenvalues to the required precision of 10^{-5}.

[1] Smaller step sizes were computationally impossible as they required more memory and swap space than was available on my system.

n	E_1 (eV)	E_2 (eV)
1	2.04746	-19.03340
2	6.90335	-8.96473
3	12.95470	-1.09948

Table 1: Energies of the first three eigenstates in potentials $V_1(x)$ and $V_2(x)$.

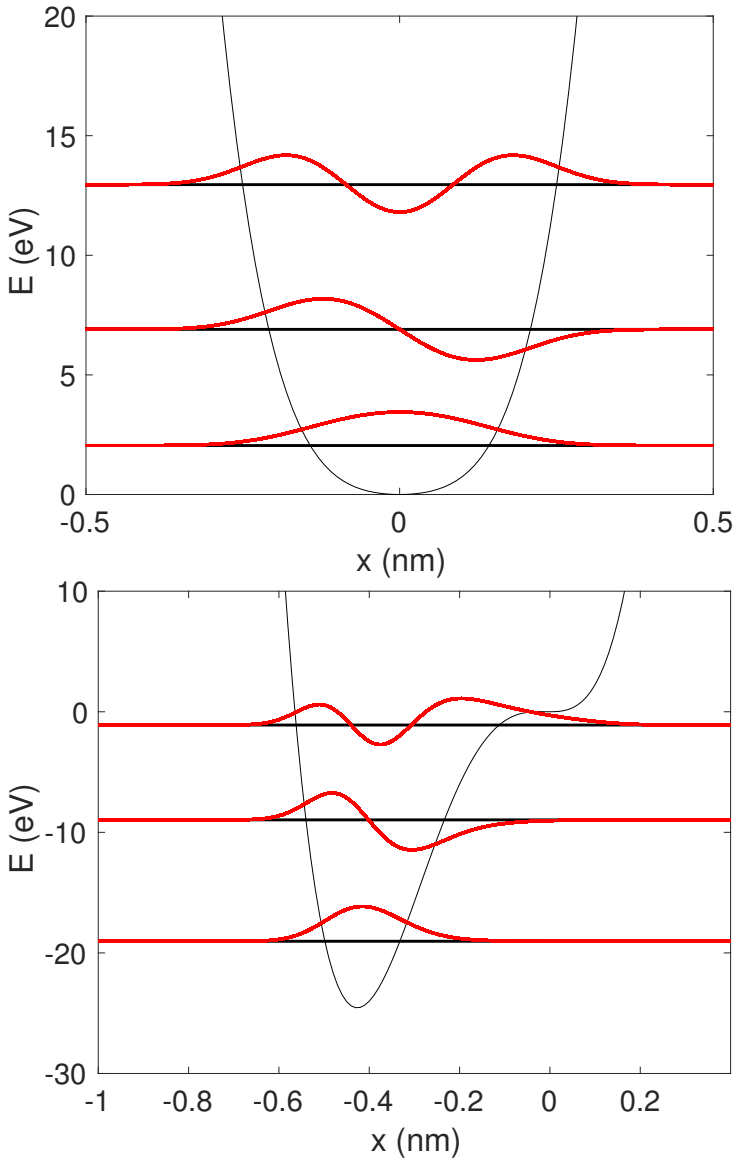


Figure 1: The first three energy eigenfunctions for $V_1(x)$ and $V_2(x)$, computed using the finite difference method.

Finite differences for non-linear ODE

The differential equation we're interested in solving is,

$$\sqrt{r}n'' = n^{3/2}. \tag{7}$$

If we discretize r into a series of points separated by step size h, then for (r_i, n_i), we have an approximate definition of the central second derivative at the i^{th} point,

$$n_i'' \approx \frac{n_{i+1} - 2n_i + n_{i-1}}{h^2}. \tag{8}$$

Substituting this expression into the ODE, we obtain,

$$\sqrt{r_i}\left(\frac{n_{i+1} - 2n_i + n_{i-1}}{h^2}\right) = n_i^{3/2}$$

$$\therefore \frac{\sqrt{r_i}}{h^2}(n_{i+1} + n_{i-1}) - 2\frac{\sqrt{r_i}}{h^2}n_i = n_i^{3/2}$$

$$\therefore \sqrt{r_i}\left(\frac{n_{i+1} + n_{i-1}}{h^2}\right) = n_i\left(\sqrt{n_i} + 2\frac{\sqrt{r_i}}{h^2}\right), \text{ or,}$$

$$n_i = \frac{\sqrt{r_i}\left(\frac{n_{i+1}+n_{i-1}}{h^2}\right)}{\sqrt{n_i} + 2\sqrt{r_i}/h^2}. \tag{9}$$

To use the finite differences method to solve this equation, we can express the j^{th} estimate of a given point (r_i, n_i) as $n_i^{(j)}$, so,

$$n_i^{(j)} = \frac{\sqrt{r_i}\left(\frac{n_{i+1}^{(j-1)}+n_{i-1}^{(j-1)}}{h^2}\right)}{\sqrt{n_i^{(j-1)}} + 2\sqrt{r_i}/h^2}.$$

In particular, the Gauss-Seidel method replaces $n_{i-1}^{(j-1)}$ with $n_{i-1}^{(j)}$, so,

$$n_i^{(j)} = \frac{\sqrt{r_i}\left(\frac{n_{i+1}^{(j-1)}+n_{i-1}^{(j)}}{h^2}\right)}{\sqrt{n_i^{(j-1)}} + 2\sqrt{r_i}/h^2}. \tag{10}$$

Furthermore, the Gauss-Seidel method can be used in tandem with over-relaxation for the j^{th} iteration of the solution, where we define the "over-relaxed" iteration as,

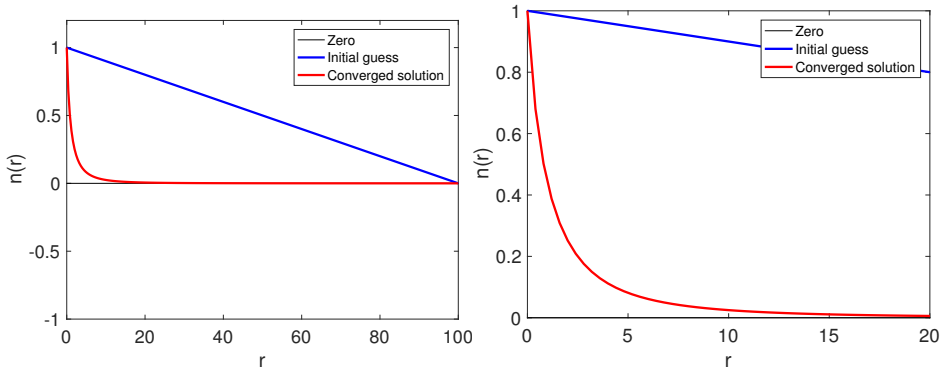
$$n_i^{(j)} \rightarrow \; n_i^{(j)} + \alpha \times \left(n_i^{(j)} - n_i^{(j-1)} \right). \tag{11}$$

where α is the relaxation parameter.

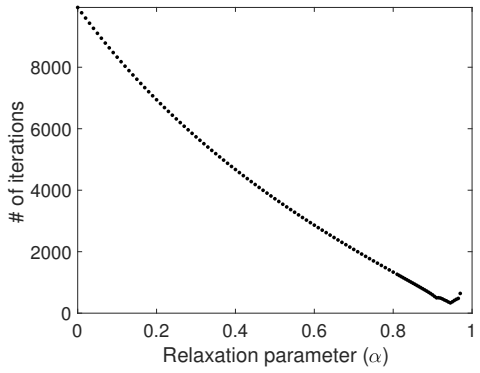
The boundary conditions used for this system were $n(0) = 1$ and $n(100) \approx n(\infty) = 0$. The grid along r was defined using $N = 250$ nodes, and the tolerance required for each point was 10^{-5}, such that the relative error at each solution iteration was defined as,

```
rel_err = max(abs(n-n_old)./abs(n)).
```

The converged, estimated solution for the differential equation, along with the initial guess, are given in Figure 2 for a relaxation parameter of $\alpha = 0.5$. Furthermore, a distribution of the number of iterations required for convergence against α is also given in Figure 2. It is evident that the use of over-relaxation improves the efficiency of the algorithm up to roughly $\alpha = 0.95$, at which point the efficiency starts decreasing as the finite differences solution fluctuates around the "true" solution for longer and longer periods before convergence.



(a) n vs. r **(b)** n vs. r (zoomed)



(c) N_{iter} vs. α

Figure 2: (Top) Distributions of n vs. r before and after the Gauss-Seidel method is applied for $\alpha = 0.5$. (Bottom) Distribution of the # of iterations required for convergence, versus the relaxation parameter, α.