PHYS 410: Project 1

Jack Hong
30935134

October 2, 2016

The three main MATLAB code files are **single_well.m**, **double_wells.m**, and **n_square_wells.m**. They are all MATLAB functions that can be run simply by clicking the run button or calling the script on the command line. All user parameters are located at the top of the file so that there is no confusion with function parameters when running the code.

The other important functions are **newton_root.m**, used to find the roots, **get_guesses.m**, used to find the initial guesses to pass to Newton's method, and **get_n_wells_func.m**, used to calculate the function $f(E)$ using the propagator method. Everything else are just small helper functions.

1 Exercise 2.16: Bound States of the Single Well Potential using the Propagator Method

Exercise 2.16: *Verify that this propagator approach leads to exactly the same solution for the square well as obtained earlier.*

The code files for this part of the project are:

- single_well.m (the main script file)
- calculate_hbar2.m
- newton_root.m
- compare_methods.m

To run the code, simply call *single_well* from the MATLAB command window. The function will return the bound energies of the well, as well as plot $f(E)$. You can vary the parameters of the well by changing the variables assigned at the top of the code.

The function $f(E)$ and $f'(E)$ were both derived by hand for this part to show understanding of the propagator method. They are defined on lines 37 and 40 in single_well.m. As per the textbook, $f(E)$ is defined as $f(E) = \Psi'(c) + \beta\Psi(c)$ where $\Psi'(x)$ and $\Psi(x)$ are obtained by applying the $P_{allowed}$ matrix to the column vector $(\Psi(b), \Psi'(b))^T = (1, \beta)^T$. ($\Psi(b)$ is arbitrarily set to 1 for the computation.) The roots of $f(E)$ are the energies of the bound states.

$$\begin{aligned}\begin{bmatrix}\psi(x)\\ \psi'(x)\end{bmatrix} &= \begin{bmatrix}cos[\alpha(x-b)] & \frac{1}{\alpha}sin[\alpha(x-b)]\\ -\alpha sin[\alpha(x-b)] & cos[\alpha(x-b)]\end{bmatrix}\begin{bmatrix}\psi(b)=1\\ \psi'(b)=\beta\end{bmatrix}\\ &= \begin{bmatrix}cos[\alpha(x-b)] + \frac{\beta}{\alpha}sin[\alpha(x-b)]\\ -\alpha sin[\alpha(x-b)] + \beta cos[\alpha(x-b)]\end{bmatrix}\end{aligned}$$

$$\begin{aligned}f(E) &= \Psi'(c) + \beta\Psi(c)\\ &= -\alpha sin[\alpha(c-b)] + \beta cos[\alpha(c-b)] + \beta cos[\alpha(c-b)] + \frac{\beta^2}{\alpha}sin[\alpha(c-b)]\\ &= -\alpha sin[\alpha w] + 2\beta cos[\alpha w] + \frac{\beta^2}{\alpha}sin[\alpha w] \quad \text{(let } w = c-b\text{, w is the width)}\\ f'(E) &= -\alpha' sin(\alpha w) - \alpha\alpha' w cos(\alpha w) + 2\beta' cos(\alpha w) - 2\beta\alpha' w sin(\alpha w) + \frac{2\beta'\alpha - \beta^2\alpha'}{\alpha^2}sin(\alpha w) + \frac{\beta^2}{\alpha}\alpha' w cos(\alpha w)\end{aligned}$$

To show that the results match the previous (tutorial 2) results numerically, the function is coded to use the same conditions as in tutorial 2. They are shown in table 1.

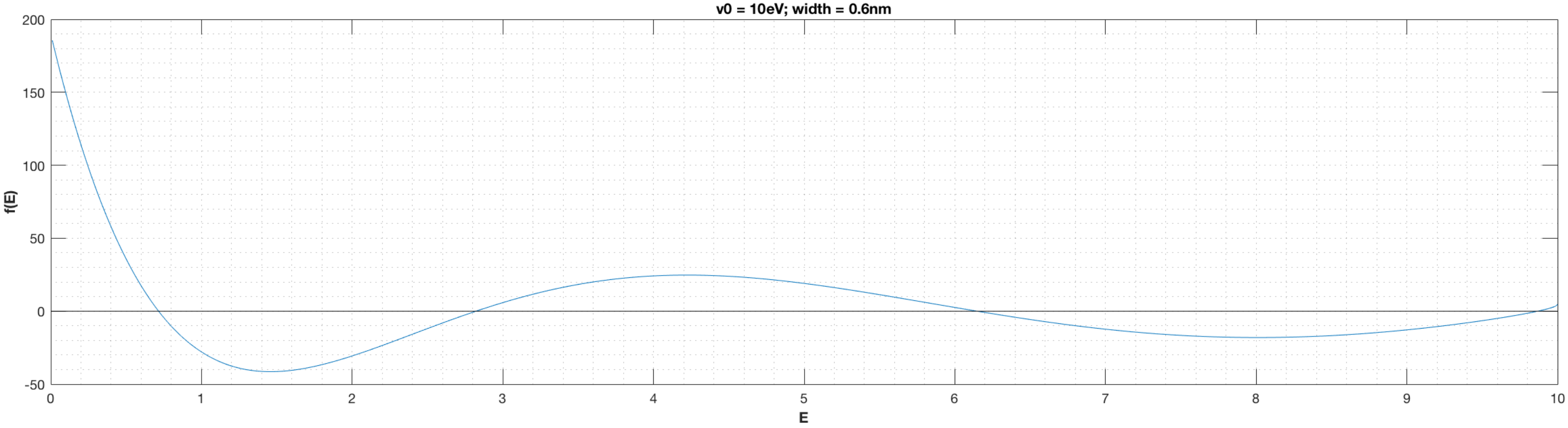
- The width of the well is 0.6 nm
- The height of the well is 10 eV
- The mass of the particle is $1m_e$

As in tutorial 2, there are 4 roots, and so 4 bound states found. The energies of the bound states are outputted by the script as a 1x4 array. These values were verified to match (within the tolerance for Newton's method) the values obtained by using the code from tutorial 2 by using the script compare_single_well_methods.m.

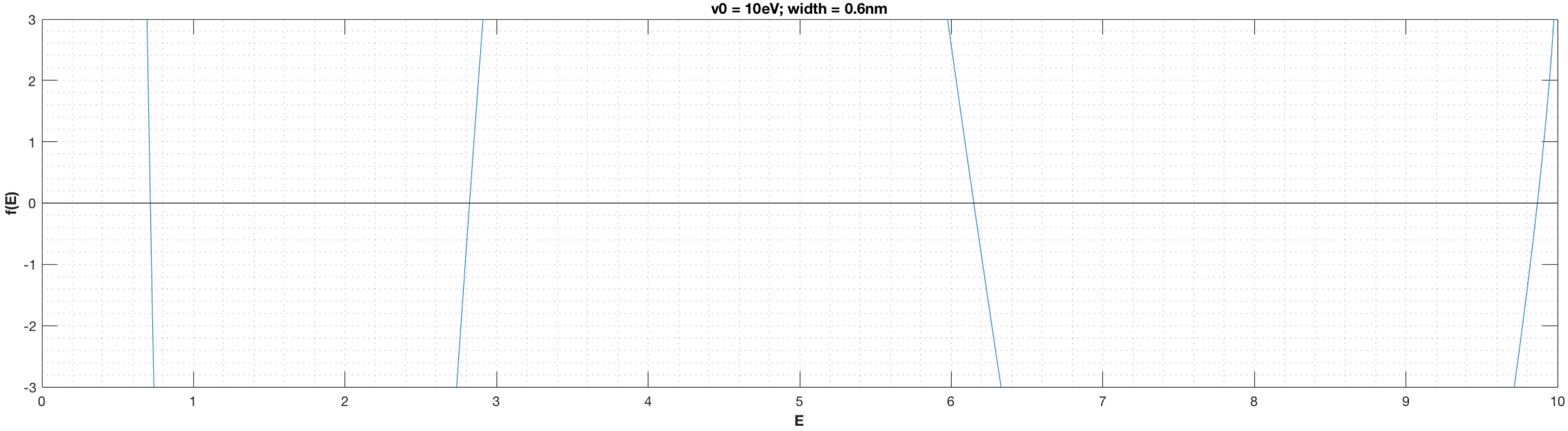
Bound state energies,Propagator (eV)	**Bound state energies, Tutorial 2 (eV)**
0.715452608397591	0.715452608371538
2.821392291788203	2.821392291594718
6.148645004380378	6.148645004137442
9.867204411132326	9.867204411848185

Table 1: Bound state energies of a finite single potential well with depth 10eV, and width 0.6nm. Both the propagator and the previous method produced exactly the same results, within the tolerance used for Newton's method (1×10^{-8}).

f(E) = $\psi'(c) + \beta\psi(c)$
Roots are bound state energies for a single square well
v0 = 10eV; width = 0.6nm



ZOOMED IN OF ABOVE: f(E) = $\psi'(c) + \beta\psi(c)$
Roots are bound state energies for a single square well
v0 = 10eV; width = 0.6nm



2 Exercise 2.18: Bound States of the Double Well Potential Using the Propagator Method

Exercise 2.18: *Solve for all energies of the double well, with $V_0 = 10eV$ and the well width, W, being* 0.6*nm as before. (Our earlier a was half the well width.) Let the separation between the wells, S, be* 0.2*nm. Then vary the distance between the wells, and plot the lowest two energies of the system versus separation distance.*

The code files for this part of the project are:

- double_wells.m (the main script file)
- find_roots.m
- get_guesses.m
- get_n_wells_func.m
- newton_root.m
- calculate_hbar2.m

To run the code, call *double_wells* from the MATLAB command window. You can vary the parameters of the well by changing the variables assigned at the top of the code.

This time, $\Psi(x)$ and $\Psi'(x)$ was calculated using MATLAB matrix multiplication to propagate the initial vector, $(\Psi(b), \Psi'(b))^T = (1, \beta)^T$ through two wells and a forbidden region of distance s.

$$\begin{bmatrix} \psi(x) \\ \psi'(x) \end{bmatrix} = P_{allowed}(E, width) * P_{forbidden}(E, sep) * P_{allowed}(E, width) * \begin{bmatrix} 1 \\ \beta \end{bmatrix} \tag{1}$$

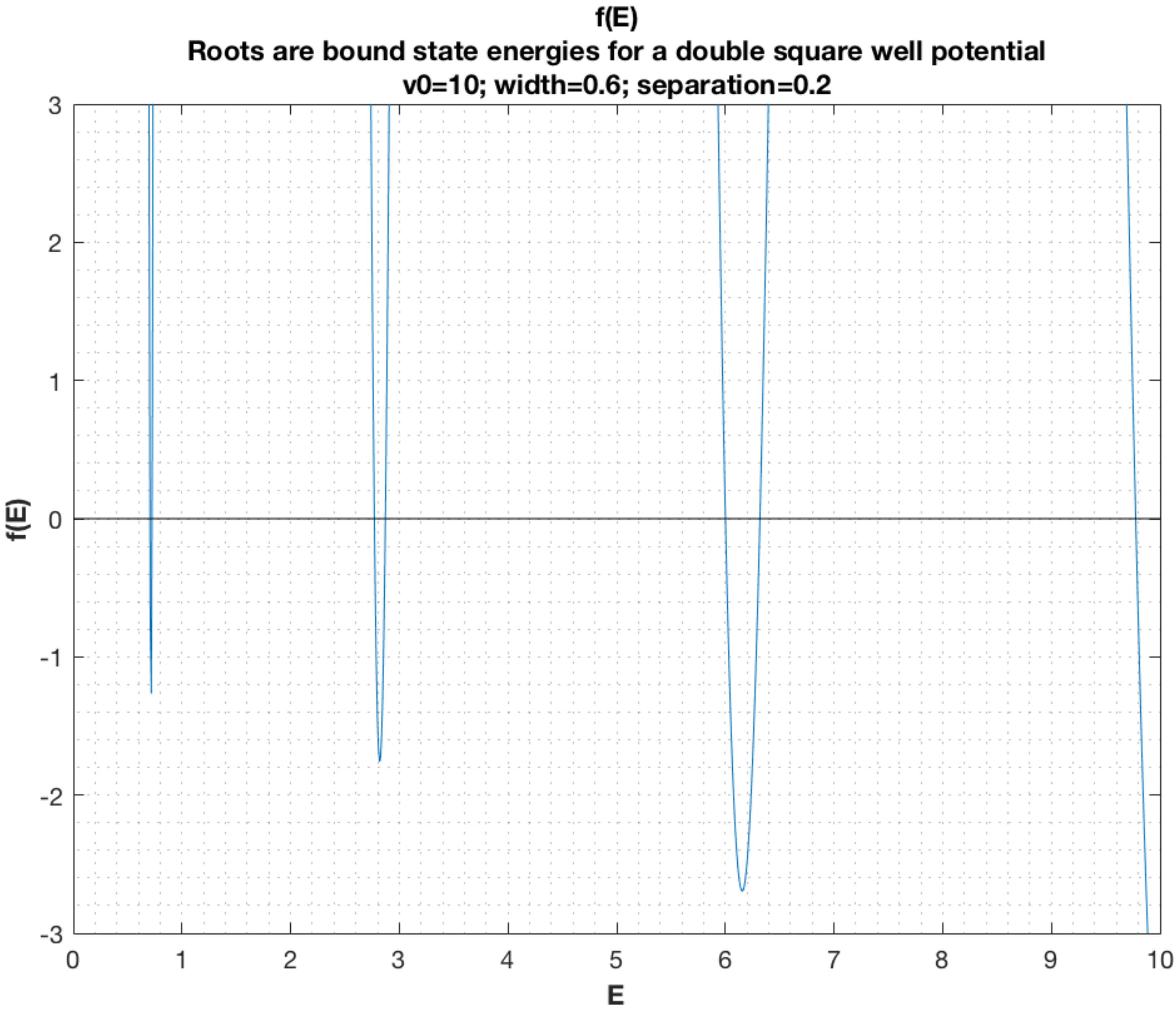
Where $P_{allowed}$ and $P_{forbidden}$ are as defined in the book. $f(E)$ is the same as before: $f(E) = \Psi'(c) + \beta\Psi(c)$. The code for generating the function $f(E)$ for a given double square well potential is found in get_n_wells_func.m.

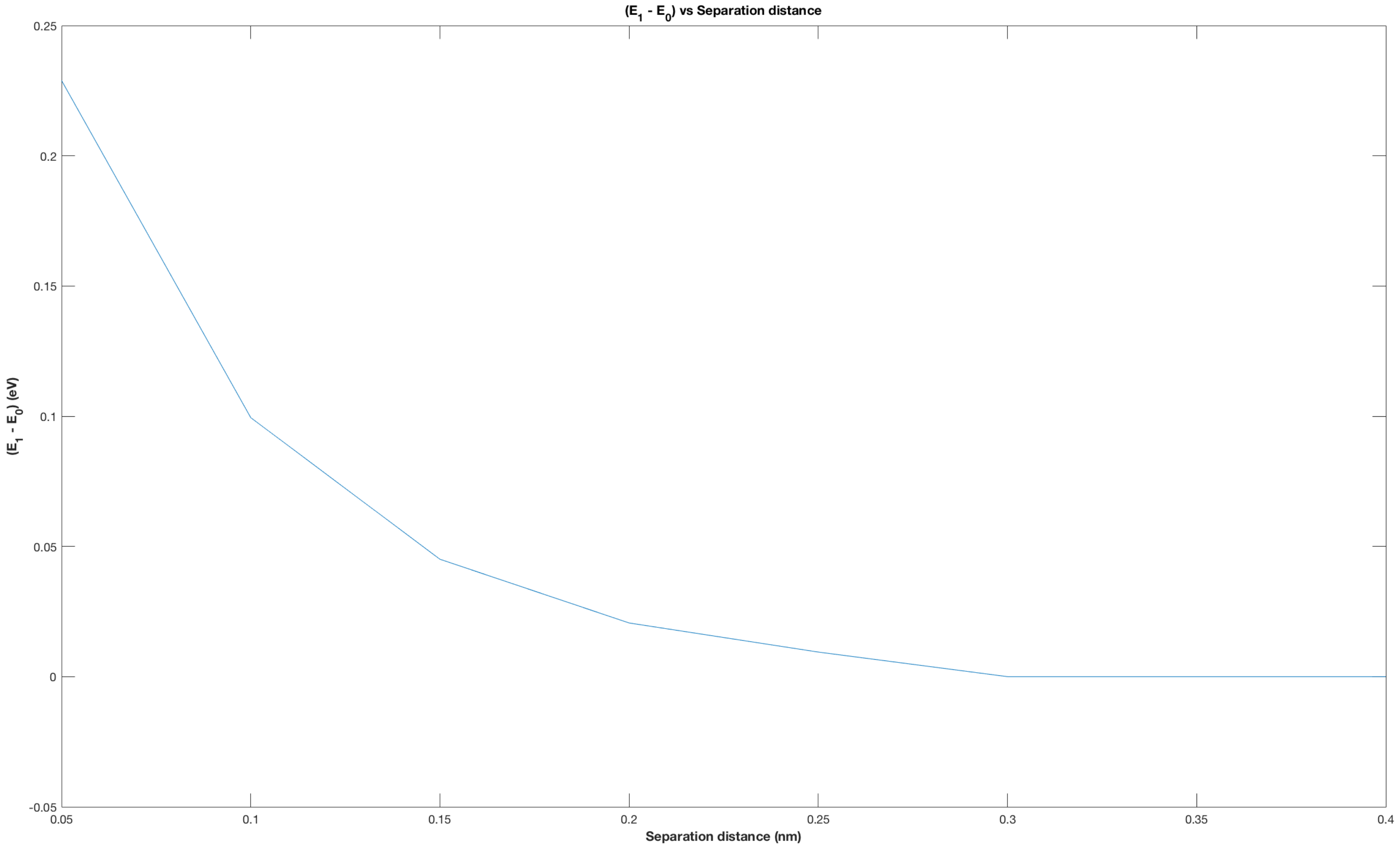
The derivative is also now calculated using a simple numerical method in the function get_f_prime.m. This function just evaluates $f(E)$ along many points, take the difference between adjacent points using the built in $diff$ function and divides the differences by the increment, h.

The 7 bound state energies for double wells of depth 10eV, width 0.6nm, and separation 0.2nm are listed in table 2.

Number	**Energy (eV)**
E_0	0.704880450332507
E_1	0.725492451539013
E_2	2.770146706209319
E_3	2.871760461975545
E_4	6.000818419886097
E_5	6.318131357882863
E_6	9.775956822047366

Table 2: The permitted bound state energies of a finite double potential well with depths 10eV, widths 0.6nm, and separation 0.2nm.


f(E)
Roots are bound state energies for a double square well potential
v0=10; width=0.6; separation=0.2
f(E)
3
2
1
0
-1
-2
-3
0
1
2
3
4
5
6
7
8
9
10
E


$(E_1 - E_0)$ vs Separation distance
$(E_1 - E_0)$ (eV)
0.25
0.2
0.15
0.1
0.05
0
-0.05
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
Separation distance (nm)

Observation: Figure 2 generated by the function shows a plot of the $E_1 - E_0$ vs s, the separation distance between the two wells. It clearly shows the inverse relationship between ΔE and s: as s increases, ΔE decreases.

3 Exercise 2.19: Bound States of the N-Square Well Potential Using the Propagator Method

Exercise 2.19: Modify your code to treat this many well problem. Use your program to investigate the energy structure of the model crystal with increasing N. What can you say about the limit as N $->$ infinity?

The code files for this part are:

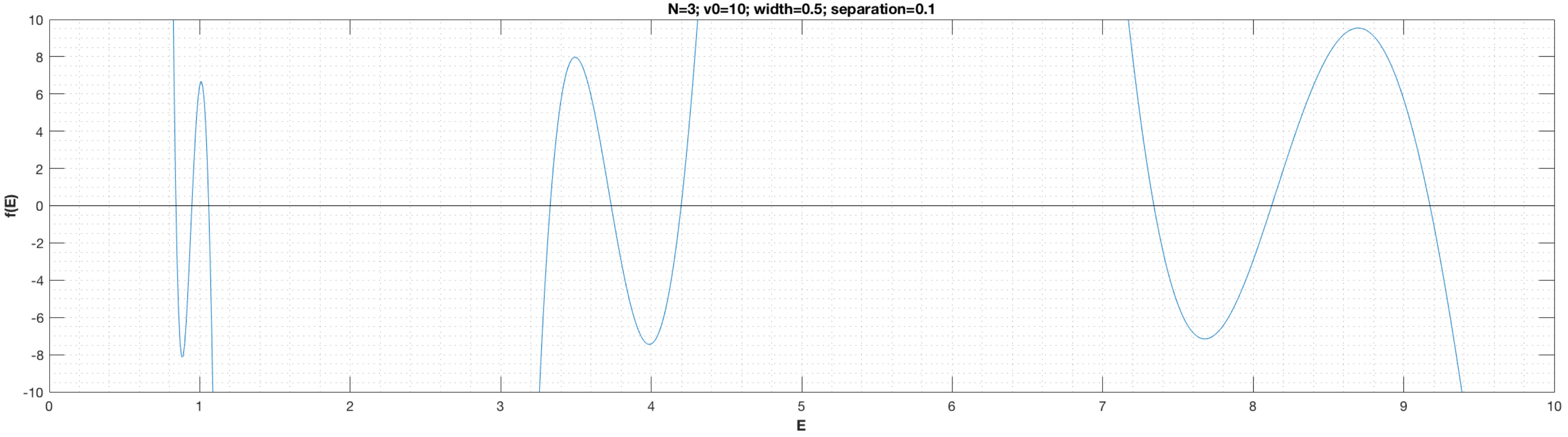
- n_square_wells.m
- find_roots.m
- get_guesses.m
- get_n_wells_func.m
- newton_root.m
- calculate_hbar2.m

To run the code, call *n_square_wells* from the MATLAB command window. You can vary the parameters of the well by changing the variables assigned at the top of the code.

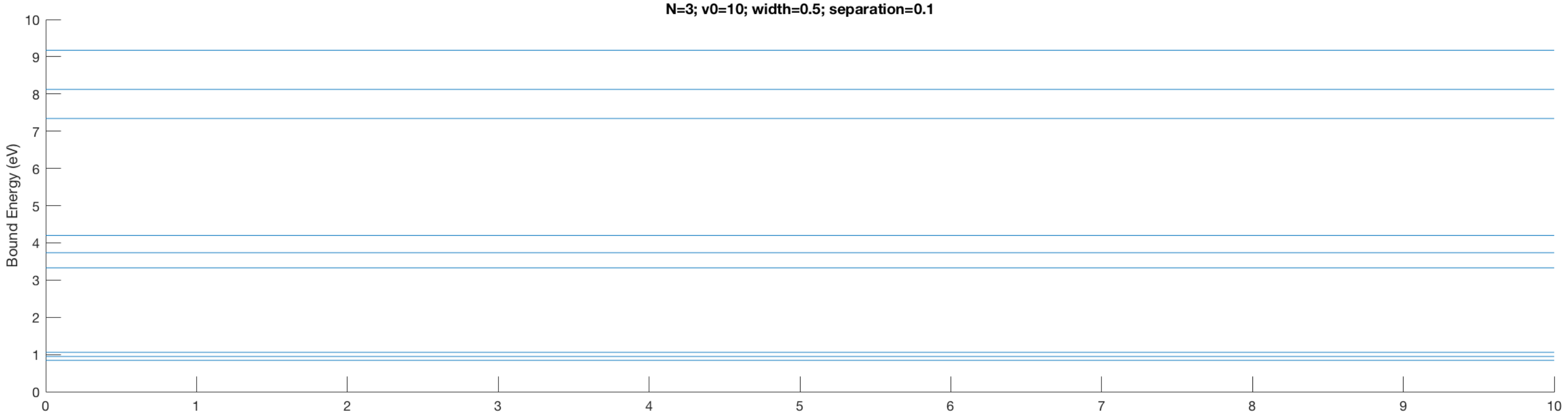
The computation is almost exactly the same as for the double square well. I use MATLAB to matrix multiply $[1;\beta]$ by $P_{allowed}$ and $P_{forbidden}$ (from the left) N number of times and use Newton's method to find the roots of the function $f(E) = \Psi'(c) + \beta\Psi(c)$. Again, the code is found in get_n_wells_func.m.

Observation: From the plots below, you can clearly see evidence for band structures forming for the allowed energies; there appear to be 3 separate groups of closely packed energy levels.

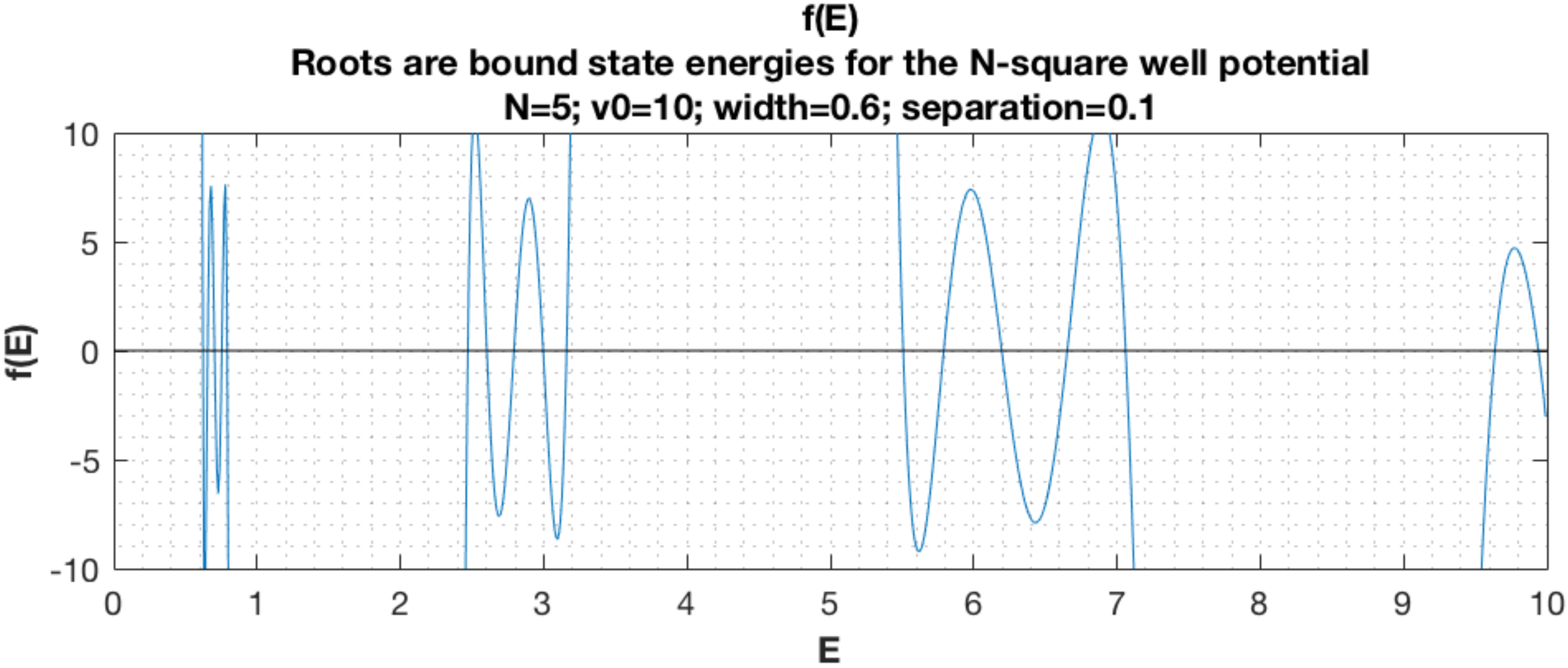
Also, you can see that as N increases, the splitting energy decreases. This is especially evident when you plot the energy levels for the 20 square well. In addition, the number of nearby energy levels = N / energy band structure as N -¿ infinity.

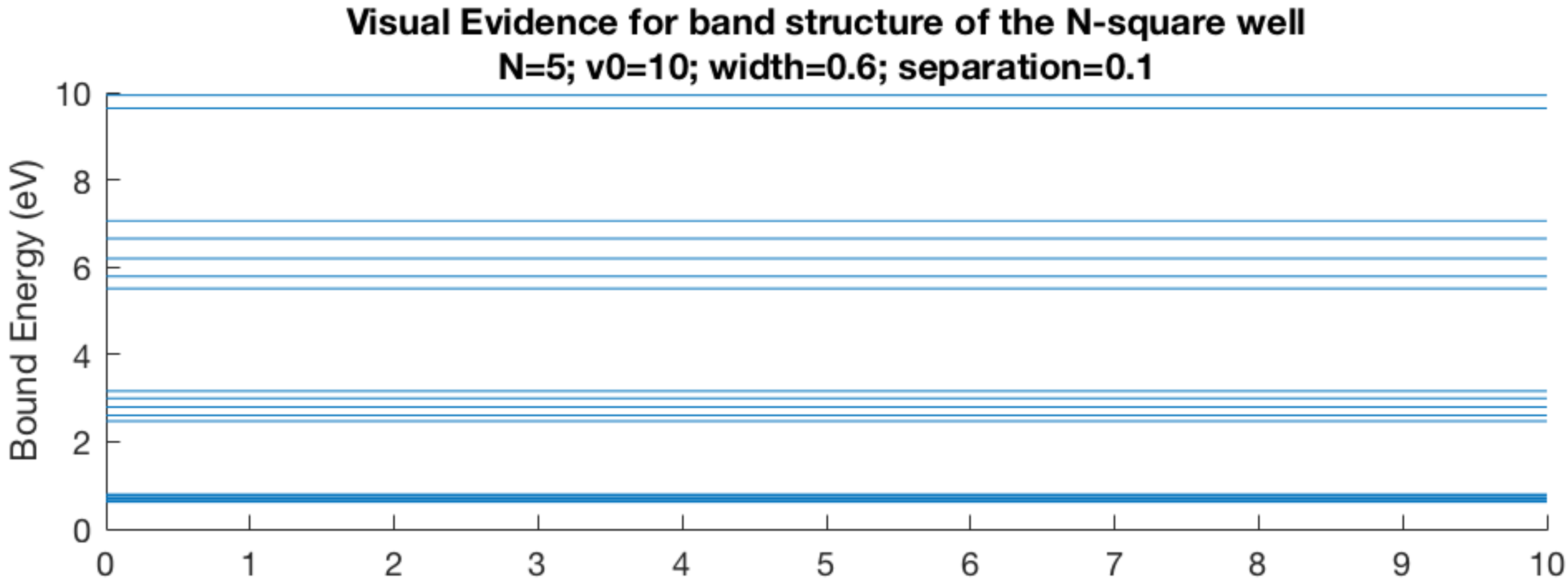


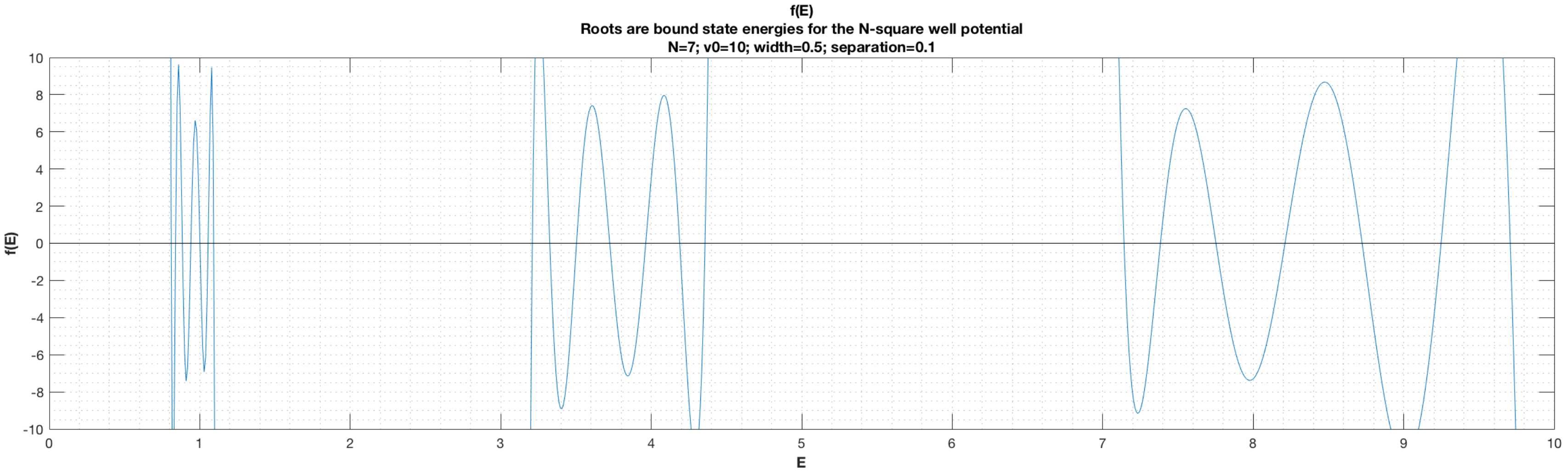
f(E)
Roots are bound state energies for the N-square well potential
N=3; v0=10; width=0.5; separation=0.1



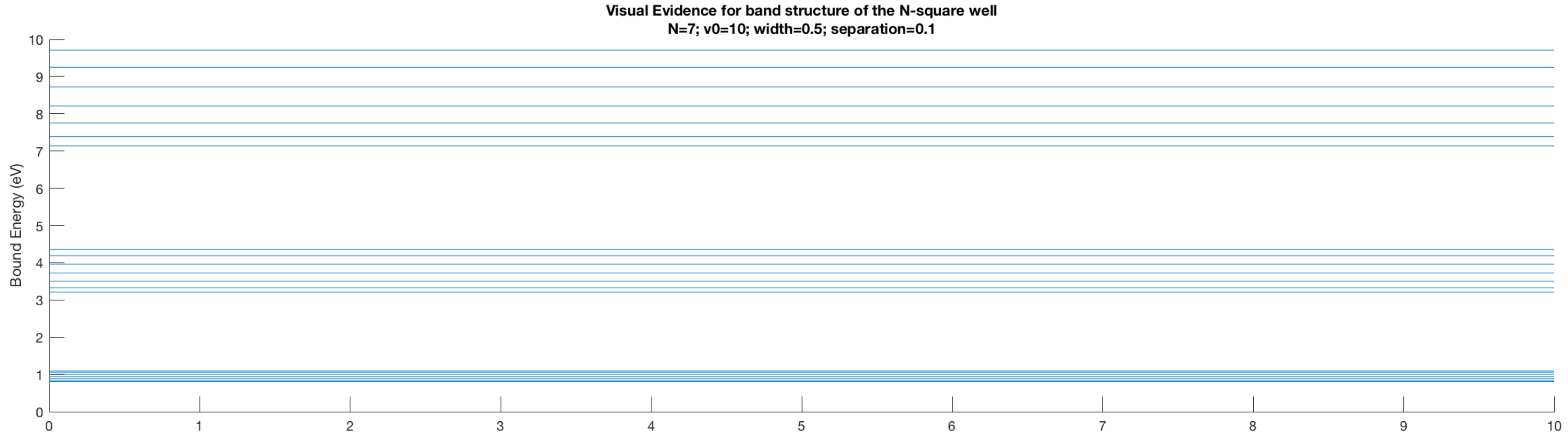
Visual Evidence for band structure of the N-square well
N=3; v0=10; width=0.5; separation=0.1


f(E)
Roots are bound state energies for the N-square well potential
N=5; v0=10; width=0.6; separation=0.1
f(E)
10
5
0
-5
-10
0
1
2
3
4
5
6
7
8
9
10
E

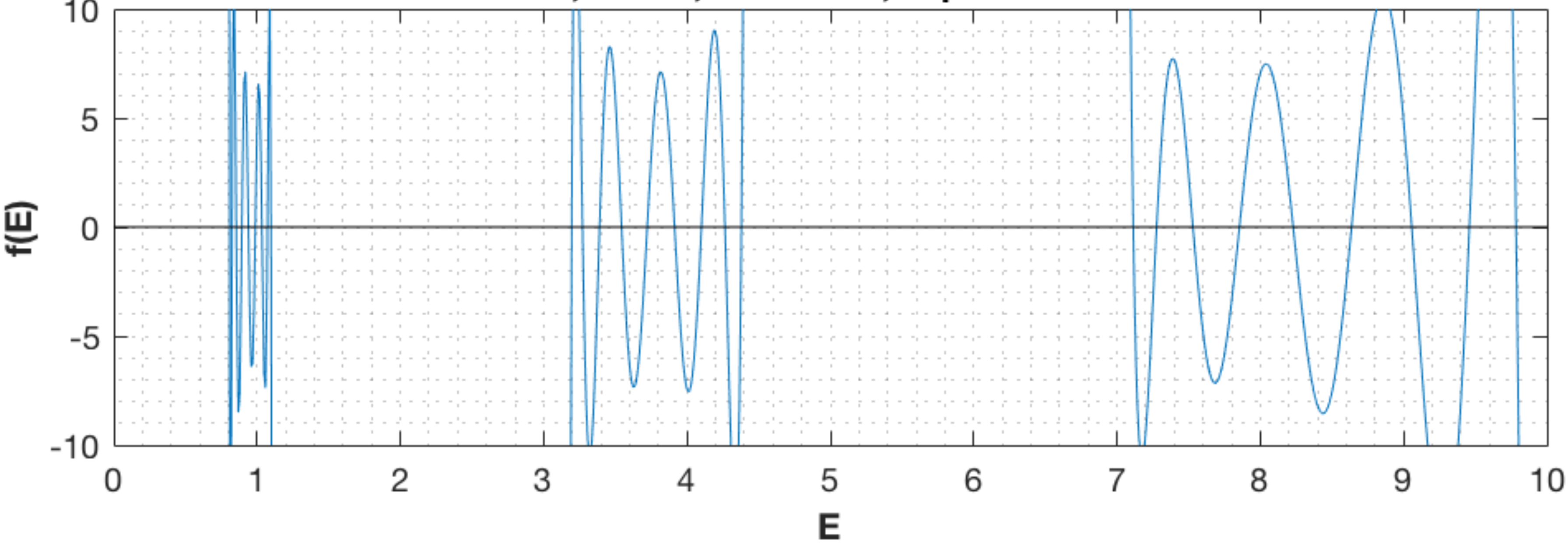

Visual Evidence for band structure of the N-square well
N=5; v0=10; width=0.6; separation=0.1
Bound Energy (eV)
10
8
6
4
2
0
0
1
2
3
4
5
6
7
8
9
10



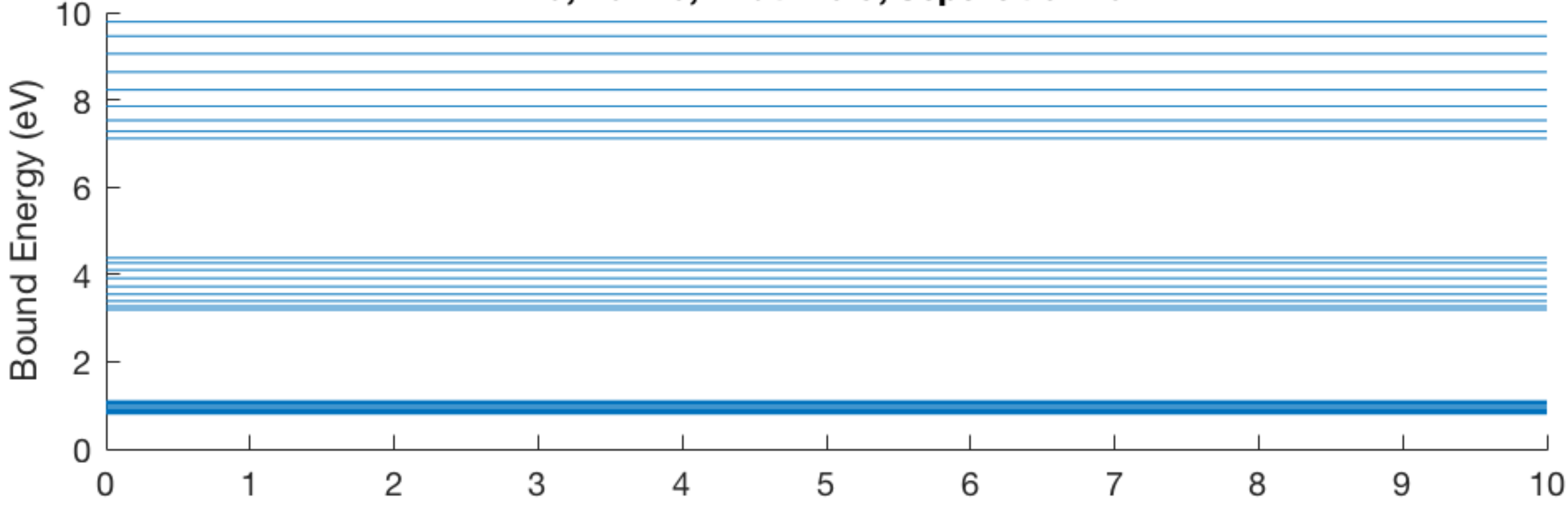
f(E)
Roots are bound state energies for the N-square well potential
N=7; v0=10; width=0.5; separation=0.1



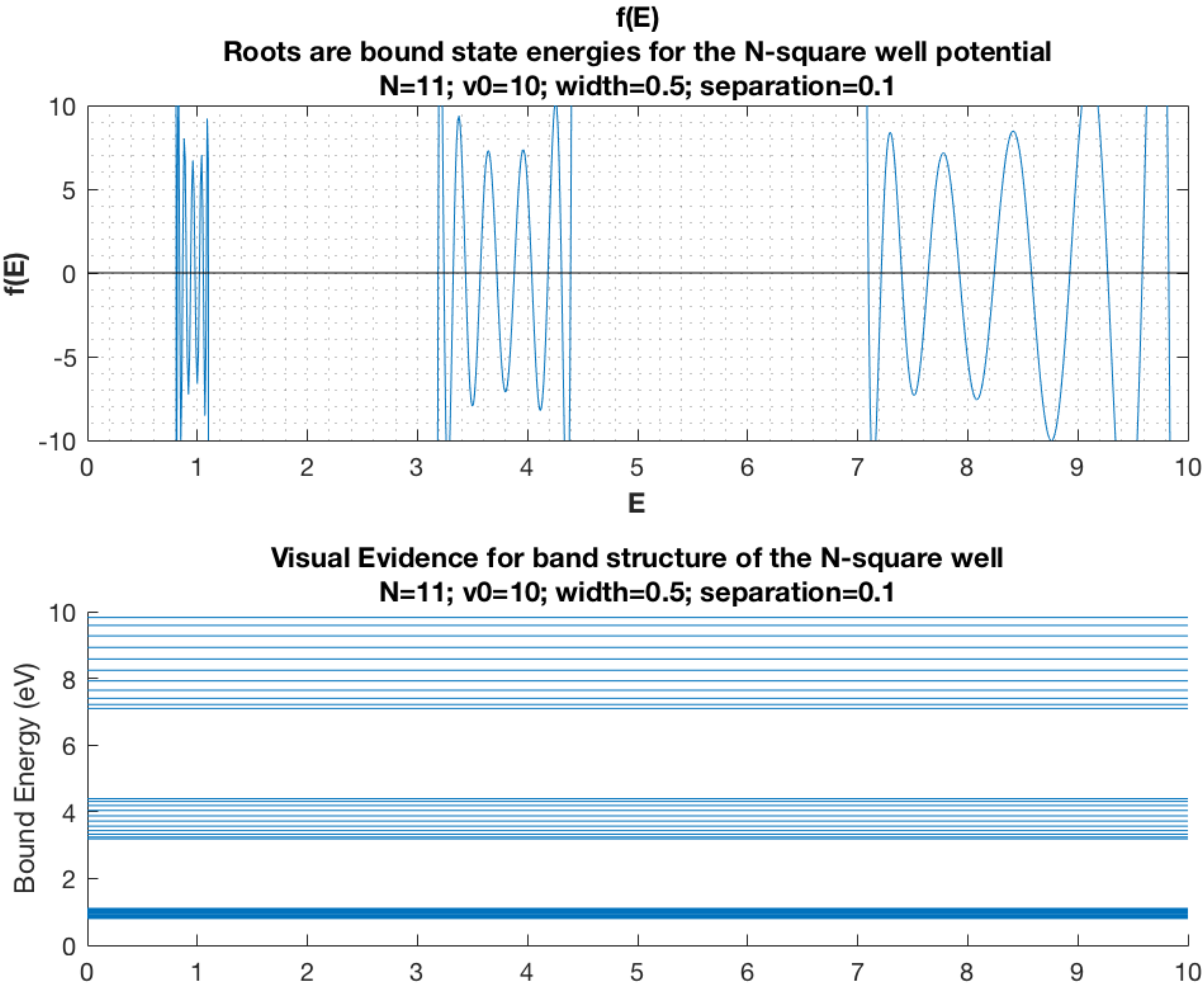
Visual Evidence for band structure of the N-square well
N=7; v0=10; width=0.5; separation=0.1



f(E)
Roots are bound state energies for the N-square well potential
N=9; v0=10; width=0.5; separation=0.1

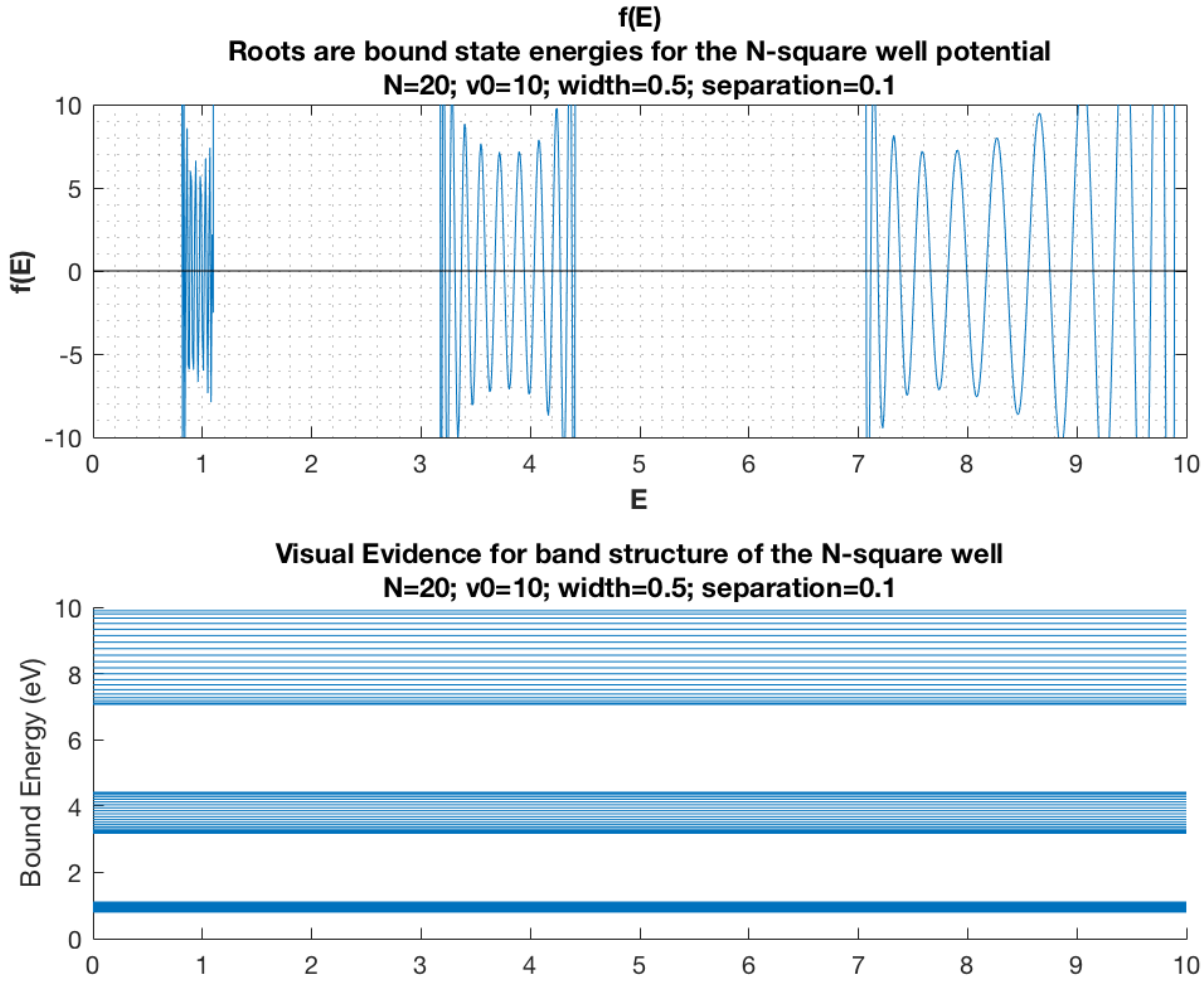


Visual Evidence for band structure of the N-square well
N=9; v0=10; width=0.5; separation=0.1



f(E)
Roots are bound state energies for the N-square well potential
N=11; v0=10; width=0.5; separation=0.1

Visual Evidence for band structure of the N-square well
N=11; v0=10; width=0.5; separation=0.1


f(E)
Roots are bound state energies for the N-square well potential
N=20; v0=10; width=0.5; separation=0.1
f(E)
10
5
0
-5
-10
0
1
2
3
4
5
6
7
8
9
10
E
Visual Evidence for band structure of the N-square well
N=20; v0=10; width=0.5; separation=0.1
Bound Energy (eV)
10
8
6
4
2
0
0
1
2
3
4
5
6
7
8
9
10