Computational Physics III: Report 2
LU decomposition

Due on Mai 2$^{\text{nd}}$, 2019

Viktor Crettenand

Contents

Problem 1

Implementation of the LU Decomposition The goal of this exercise is to design a function to decompose a square matrix into two matrices, one having only indices on the diagonal and over the diagonal and the other only on the diagonal and below the diagonal. When multiplied in the order LU they must yield the original matrix that was decomposed. This function along with backward and forward substitution was used to solve the following system of linear equations:

$$\begin{cases} x_1 + \ x_2 + x_3 + \ x_4 = 13 \\ 2x_1 + 3x_2 \qquad - \ x_4 = -1 \\ -3x_1 + 4x_2 + x_3 + 2x_4 = 10 \\ x_1 + 2x_2 - x_3 + \ x_4 = -1 \end{cases} \tag{1}$$

The found solution is

$$\vec{x} = (2, 0, 6, 5) \tag{2}$$

The comparison between your and MatLabs implementation of the LU decomposition On figure 1 a measure of the difference between the lower matrix obtain with Matalb's LU decomposition function and the one obtained by the function that was implemented (without pivoting) can be seen as a function of the size of the matrix being decomposed. To obtain a measure of the difference between the two matrices the maximum index difference was computed and an average of this value was taken over 10 random matrices of same size. The comparison can be seen on figure 1 and figure 2 for the comparison only for a few smaller matrices. The difference is of the order 10^{15} for only a few value. In these cases division by values close to 0 must occur. Looking at figure 2 we see that in between these peaks the difference between the L and U matrices is small: of the order 10^{-16}.

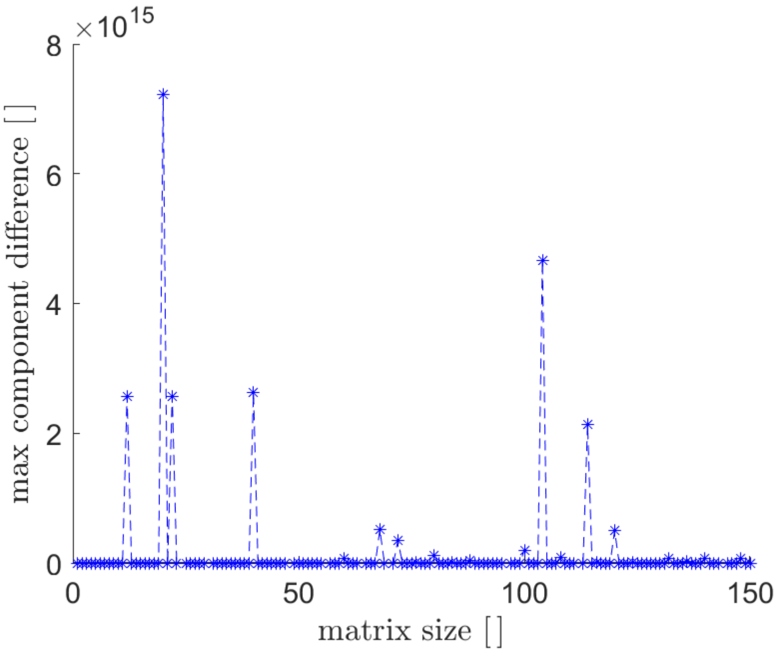


Figure 1: Comparison between the results of Matlab's LU decomposition and the implemented LU decomposition without pivoting

Problem 1 continued on next page. . .

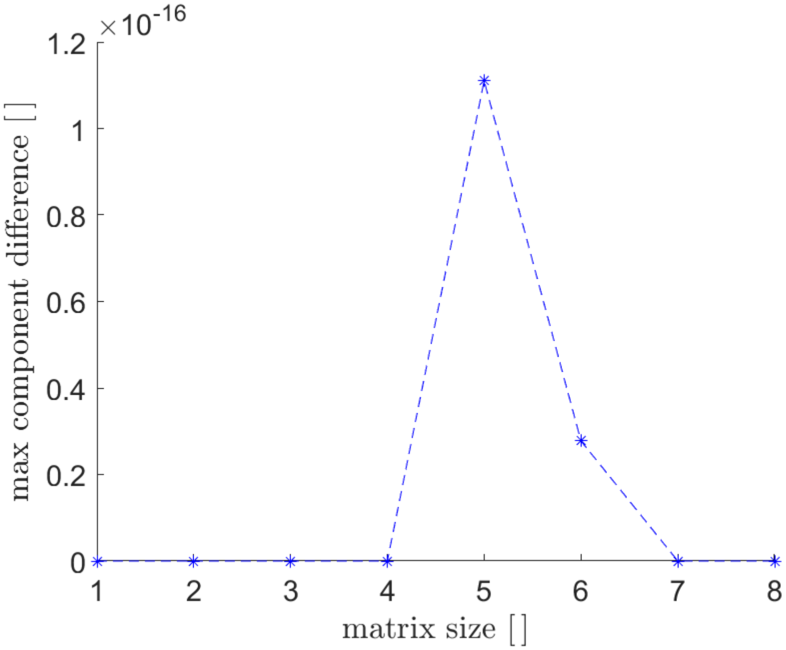


Figure 2: Comparison between the results of Matlab's LU decomposition and the implemented LU decomposition without pivoting for small values

3) Test your implementation on the following matrix: Is your result meaningful? Comment on the origin of the observed problem (if any). The implementation of the classical *LU* algorithm was tested on the matrix:

$$\begin{pmatrix} 0 & -7 & 0 \\ -3 & 2 & 6 \\ 5 & -1 & 5 \end{pmatrix} \tag{3}$$

The obtained result is not meaningful because it has infinity and "NaN" values in some components which means there were divisions by zero. The error is algorithmic and is due to the instability of the algorithm and singularities. As can be seen below, at the beginning of the *LU* algorithm without pivoting, U is set equal to A and on the seventh line, a division by U_{11} is performed. Since $A_1 1$ is equal to zero this division is ill defined and causes a singularity.

Listing 1: Implementation of the *LU* algorithm without pivoting

```
1  function [L,U]=lu_withoutpiv(A)
2  n=length(A);
3  L=eye(n);
4  U=A;
5  for k=1:n-1
6      for i=k+1:n
7          L(i,k)=U(i,k)/U(k,k);
8          for j=k:n
9              U(i,j)=U(i,j)-L(i,k)*U(k,j);
10         end
11     end
12 end
13 end
```

To correct this ill behaviour of the *LU* algorithm the lines of the matrix can be pivoted in such a way as to always divide by finite numbers. *LU* algorithm with pivoting was coded which implements this principle.

The LU decomposition of this same matrix with this improved algorithm gives L and U matrices without infinity or "NaN" values (not a number).

Problem 2

Barycenter The goal of this exercise is to solve a system of linear equations to determine the weight of a set of celestial bodies as well as their barycenter. The system of equation obtained using the information in formulation of the problem is:

$$\begin{cases} 9m_\beta & = 12 \cdot 6 \\ \quad 6m_\alpha & = 12\cdot \\ \qquad 3m_\gamma + 10m_\epsilon & = 14 \cdot 4 \\ \qquad 5m_\gamma + 12m_\epsilon & = 14 \cdot 6 \\ \qquad\qquad 3m_\delta + 7m_\zeta & = 32 \cdot \frac{25}{4} \\ \quad 6m_\alpha \qquad + 4m_\delta & = 32 \cdot \frac{25}{4} \end{cases} \tag{4}$$

The result of the system of equations is:

$$m_\alpha = 4.0000\mathrm{M}_{♁} = 2.389 \cdot 10^{25}\mathrm{kg} \quad m_\beta = 8.0000\mathrm{M}_{♁} 4.778 \cdot 10^{25}\mathrm{kg} \quad m_\gamma = 12.0000\mathrm{M}_{♁} 7.166 \cdot 10^{25}\mathrm{kg}$$
$$m_\delta = 24.0000\mathrm{M}_{♁} 1.4333 \cdot 10^{26}\mathrm{kg} \quad m_\epsilon = 2.0000\mathrm{M}_{♁} 1.194 \cdot 10^{25}\mathrm{kg} \quad m_\zeta = 4.0000\mathrm{M}_{♁} 2.389 \cdot 10^{25}\mathrm{kg}$$

The barycenter is:

$$R = \begin{pmatrix} 4.2222 \\ 3.7778 \end{pmatrix} \text{light minutes} = \begin{pmatrix} 7.5947 \\ 6.7953 \end{pmatrix} \cdot 10^7 \quad \mathrm{km} \tag{5}$$

Problem 3

Lotka-Voterra equations The Lotka-Volterra equations, also known as the predator-prey equations are a system of first order non linear differential equations. The goal of this exercise is to determine the population of each species (the value of each variable) at equilibrium. To achieve that we use the fact that at equilibrium $\frac{dx_i}{dt} = 0$ and simply solve the system of equations thus obtained. Another goal of this exercise is to understand which variable corresponds to a prey and which corresponds to a predator by looking at the system of equations.

If only a single type of species exists, the differential equation is:

$$\frac{dx}{dt} = rx - Ax^2$$

The term rx can be interpreted as simple reproduction and the term $-Ax^2$ can be interpreted as competition among the species for food and other resources. Depending on the initial conditions the two possible equilibrium populations are:

$$x = 0 \quad \text{individuals}$$
$$x = \frac{r}{A} \quad \text{individuals}$$

In the case of four interacting populations we have the following equations describing the dynamics of the populations:

$$\frac{\mathrm{d}x_1}{\mathrm{d}t} = r_1 x_1 - x_1 \sum_i A_{1i} x_i \tag{6}$$
$$\frac{\mathrm{d}x_1}{\mathrm{d}t} = r_2 x_2 - x_1 \sum_i A_{2i} x_i \tag{7}$$
$$... \tag{8}$$
$$\frac{\mathrm{d}x_N}{\mathrm{d}t} = r_N x_1 - x_N \sum_i A_{Ni} x_i \tag{9}$$

Let's take the case of:

$$A = \begin{pmatrix} 0 & 10 & 50 & 0 \\ -1 & 3 & 10 & 0 \\ -2 & 10 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{10}$$

The matrix A represents the interaction between the different species. Observing the values of the coefficient one can infer that the first species: x_1 is a prey, the second and third species: x_2 and x_3 are predators and the last one: x_4 is neither a pray nor a predator. This is justified by looking at matrix A and noticing that the first species has a positive interaction coefficient with the second and third whereas the second and third have negative interaction coefficient with the first. Looking at equation 6, one can see that a negative coefficient in matrix A means that the term $-A_{ij}x_j$ will be positive thus increasing the time derivative of a species i : $\frac{dx_i}{dt}$. Additionally looking at vector $\vec{r}$ one will notice that the second and third components are zero. This means that species two and three are species that can not grow unless there is a species with which they have a negative interaction coefficient. The fourth species doesn't depend on the others because it has zero interaction coefficient with all other species than itself.

Problem 3 continued on next page...

The equilibrium population of the four species are given by:

$$\begin{aligned}
x_1 &= 1.2500\,\text{individuals}\\
x_2 &= 0.2500\,\text{individuals}\\
x_3 &= 0.0500\,\text{individuals}\\
x_4 &= 1.0000\,\text{individuals}
\end{aligned}$$

Problem 4

Conditioning In this exercise the conditioning of matrices is explored. A linear equation system of the form $A\vec{x} = b$ is solved with a matrix A that is ill-conditioned. The condition number of A is $5.1998 \cdot 10^3$. Condition numbers much greater than one are a sign that the matrix is ill-conditioned. An ill-conditioned matrix is a matrix for which a small relative different in b: $\left\|\frac{\delta b}{b}\right\|$ lead to a large relative difference in x: $\left\|\frac{\delta x}{x}\right\|$. That is exactly what happens in this case. The two systems of equation differ only in the vector b and are given by:

$$\begin{cases} x_1 \quad + x_3 = 2 \\ 1.001x_1 + x_2 = 1 \\ - x_2 + x_3 = 1 \end{cases} \tag{11}$$

$$\begin{cases} x_1 \quad + x_3 = 2.001 \\ 1.001x_1 + x_2 = 1 \\ - x_2 + x_3 = 1 \end{cases} \tag{12}$$

The solution to 11 is:

$$x = \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} \tag{13}$$

And the solution to 12 is:

$$x = \begin{pmatrix} -1.0000 \\ 2.0010 \\ 3.0010 \end{pmatrix} \tag{14}$$

As expected the relative difference between the two solution is large even though the relative difference $\left\|\frac{\delta b}{b}\right\|$ is small.

Problem 5

Power methods Three different power method functions were implemented : the classical power method, the inverse power method with shift and the Rayleigh quotient algorithm.

Listing 2: Implementation of the classic power method which returning the eigenvalue having the largest absolute value

```
function[vec,val] = eig_power(A)
vec=1/sqrt(length(A)).*ones(length(A),1);
val=vec'*A*vec;
err=1;
while err>1e-11
    tempvec=vec;
    tempval=val;
    vec=A*tempvec;
    vec=vec./norm(vec);
    val=vec'*A*vec;
    err=abs(val-tempval);
end

end
```

Listing 3: Implementation of the inverse power method which returns the closest eigenvalue to a given target

```
function[vec,val] = eig_ipower(A,target)
vec=1/sqrt(length(A)).*ones(length(A),1);
val=vec'*A*vec;
err=1;
while err>1e-14
    tempvec=vec;
    tempval=val;
    vec=(A-eye(length(A))*target)\tempvec;
    vec=vec./norm(vec);
    val=vec'*A*vec;
    err=abs(val-tempval);
end

end
```

Listing 4: Implementation of the Rayleigh quotient algorithm which return the closest eigenvalue to a given target

```
function[vec,val] = eig_rq(A,target)
vec=1/sqrt(length(A)).*ones(length(A),1);
val=vec'*A*vec;
err=1;
while err>1e-11
    tempvec=vec;
    tempval=val;
    vec=(A-eye(length(A))*tempval-eye(length(A))*target)\tempvec;
    vec=vec./norm(vec);
    val=vec'*A*vec;
    err=abs(tempval-val);
end
```

Problem 5 continued on next page. . .

```

end
```

Problem 6

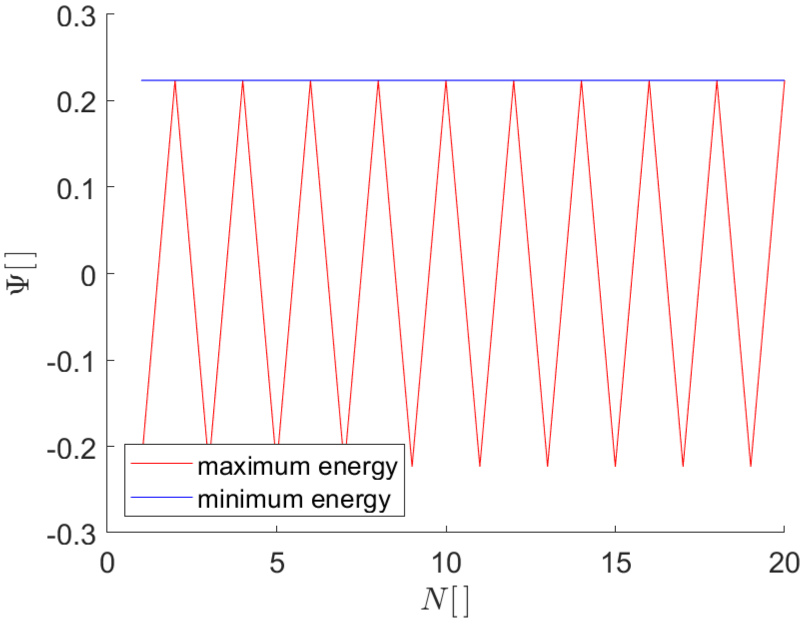
Tight-binding model for a loop of atoms In this exercise we consider a loop of N identical atoms and apply the tight binding approximation to solve Schrdinger's equation. We assume a hoping integral γ of $-1\,\mathrm{eV}$. In the case $N = 20$ the smallest eigen-energy found is $-2\,\mathrm{eV}$ and the biggest is $2\,\mathrm{eV}$. The plots of the wave functions of lowest and highest energy can be seen on figure 3 and the plots of the probability density functions can be seen on figure 4.

We notice that the highest energy wave function oscillates whereas the ground state is constant. This is in analogy with the wave functions of a particle in a box. For a particle in a box, the fundamental state is a bump shaped wave function and the following greater energy wave function have 2 and 3 oscillation and so on.

If we consider the kinetic part of a Hamiltonian in the Schrodinger equation, we notice that the energy depends on the second derivative of the wave function. A constant wave function has a vanishing second derivative which means it has no kinetic contribution to its energy.

If we think in terms of hopping integrals γ, we notice that the energy is lowered if the wave function has the same sign on two neighbouring atoms. It makes sense that the energy is highest when each site has the opposite sign of its neighbours.

This analysis was repeated for the case $N = 21$. The lowest energy eigenstate remains $-2\,\mathrm{eV}$ but the highest energy eigenstate becomes $1.977\,\mathrm{eV}$. This is explained by the fact that since the number of sites is now odd, a wave function changing sign at each site isn't possible anymore. Because of this there will be at least on hopping integral contributing negatively to the integral. The plots of the wave functions of lowest and highest energy for this case can be seen on figure 5 and the plots of the probability density functions can be seen on figure 6.



6

Figure 3: Wave function of the maximum and minimum energy eigenstates of a loop of 20 atoms

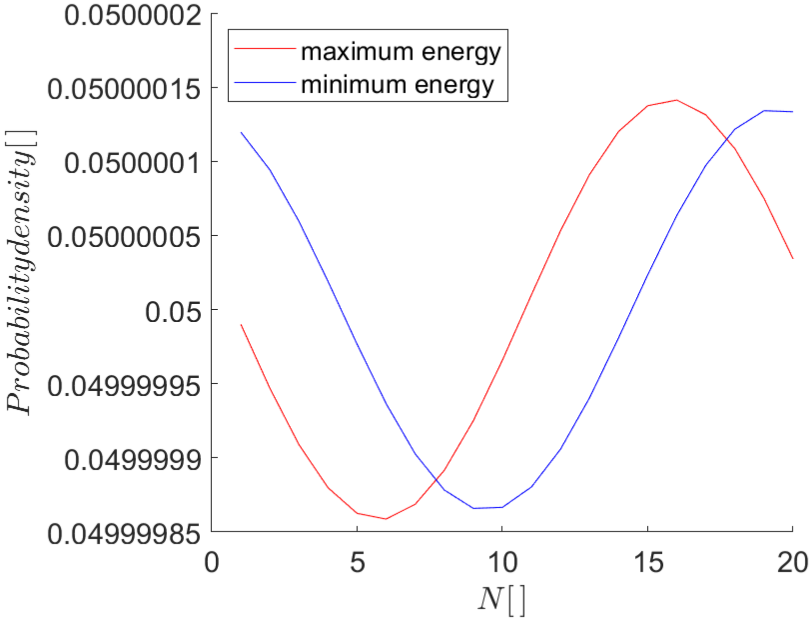


Figure 4: Probability density function of the maximum and minimum energy eigenstates of a loop of 20 atoms

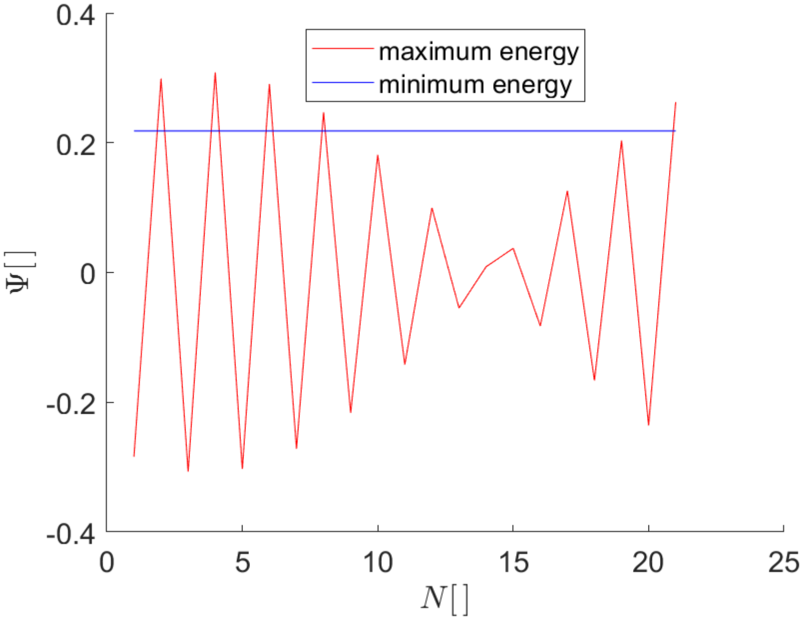


Figure 5: Wave function of the maximum and minimum energy eigenstates of a loop of 21 atoms

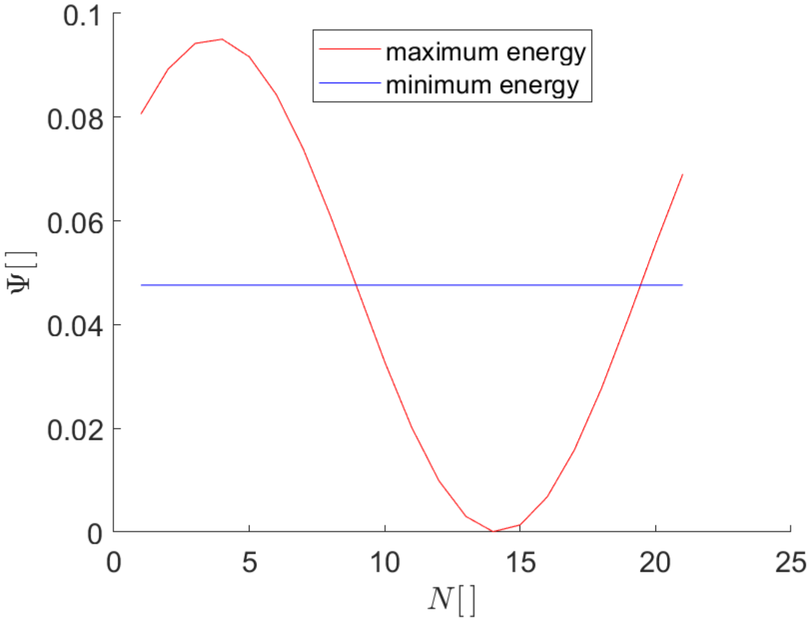
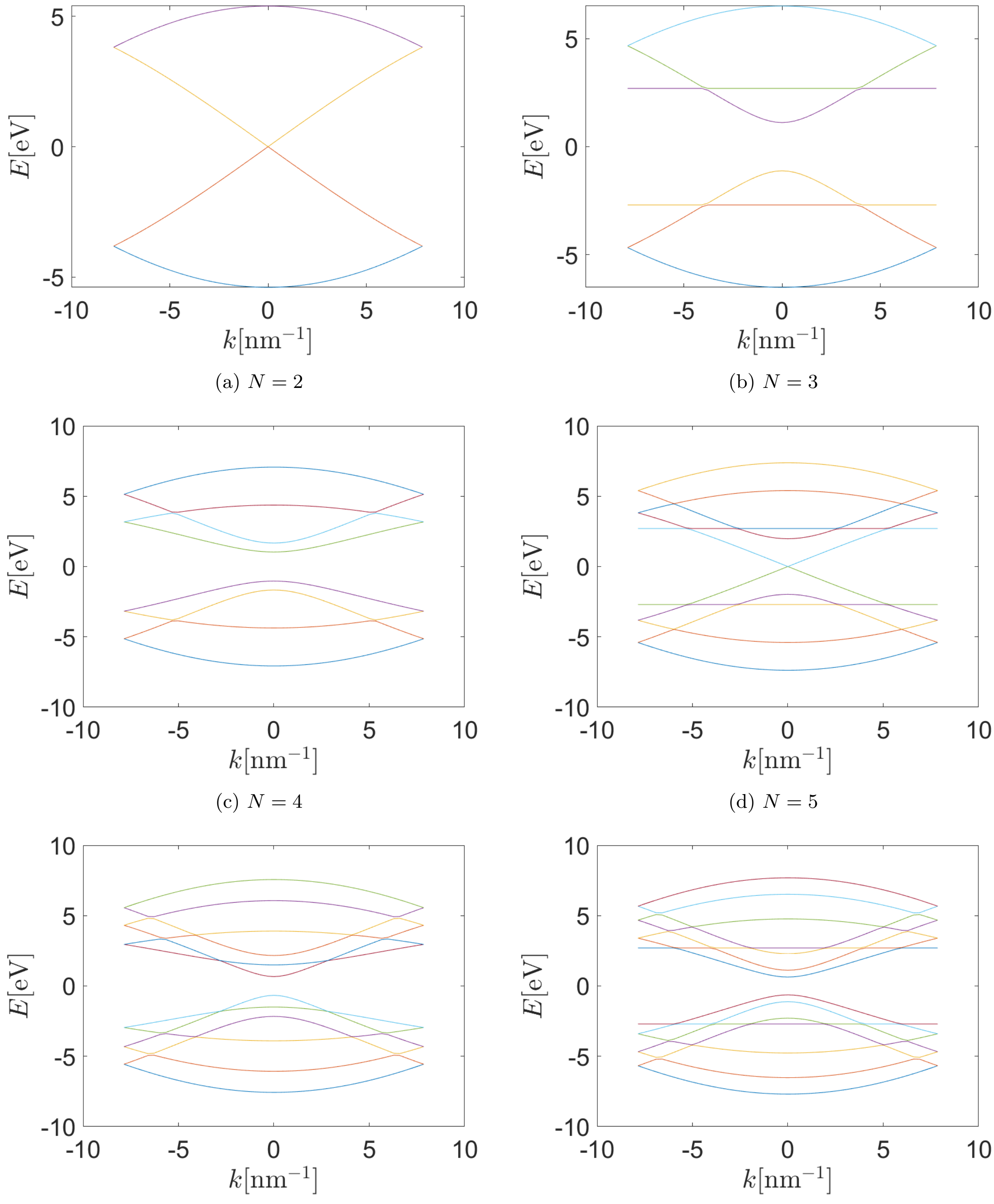


Figure 6: Probability density function of the maximum and minimum energy eigenstates of a loop of 21 atoms

Problem 7

Band structures of one-dimensional periodic graphene nanoribbons In this exercise one dimensional nanoribbons of graphene are considered. Cases with different values of the defining parameter $N[]$ are considered where N defines the width of the nanoribbon. The band structure of the nanoribbons for $N = 2, ..., 7$ can be seen on figure 7.



(a) $N = 2$

(b) $N = 3$

(c) $N = 4$

(d) $N = 5$

(e) $N = 6$

(f) $N = 7$

Figure 7: Band structure of graphene nanoribbons for different defining parameter $N[]$

In this particular system the N lowest bands are populated by electrons. A plot of the band gap between the N^{th} and the $N+1^{\mathrm{th}}$ band in function of N can be seen on figure 8. There are vanishing band gaps for systems with $N = 2 + m * 3$ with m a positive integer at least until $N = 20$. This means that thos systems are metals. All other systems are insulators.

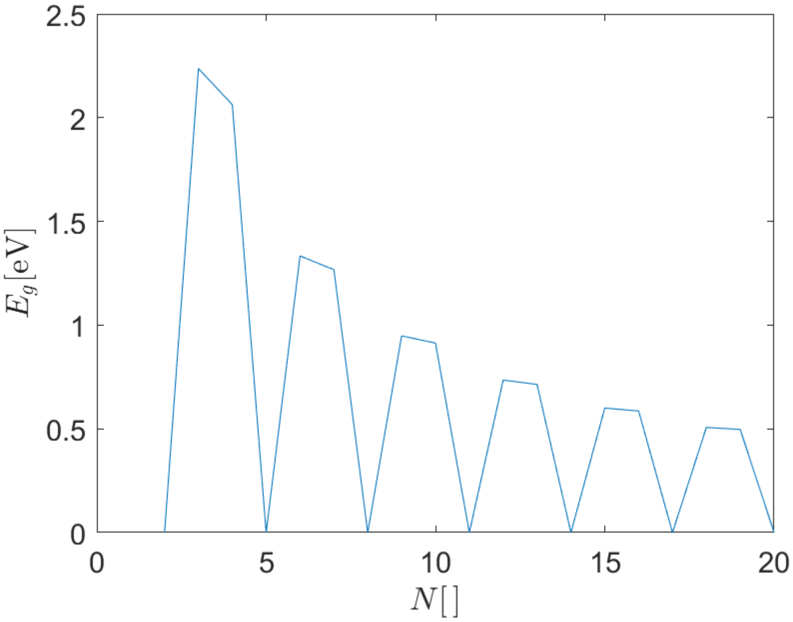


Figure 8: Energy gap E_g eV between the conduction and valence bands as a function of the defining parameter $N[\,]$

Problem 8

Jacobi method The Jacobi method is an algorithm that calculates the eigenvalues and eigenvectors of a matrix interactively. The classical and cyclic Jacobi algorithms where implemented. In the classical algorithm at each iteration the indices of the maximum off diagonal term is computed and reduced whereas the cyclic algorithm simply goes over all indices regardless their value.

These two algorithm where compared by measuring the computation time on a broad range of magnetizable matrices of different size and by measuring the number of Jacobi rotations required to perform diagonalization this can be seen on figures 9 and 10. The number of Jacobi rotation is the number of time the function reduces an off diagonal element. The computations where made with an error tolerance of $1 \cdot 10^{-2}$. This means the Jocobi algorithm stopped when the norm of the off-diagonal terms where less than $1 \cdot 10^{-2}$. As can be seen on figure 9, the time take by the cyclic Jacobi algorithm is lesser than the time taken by the classical Jacobi algorithm. This is due to the high cost of the computation of the indices of the maximum in the classical algorithm. As one would expect the cyclic algorithm makes more Jacobi rotations since in makes Jacobi rotation for all off diagonal elements whereas the classical algorithm prioritizes the big off diagonal elements and never goes over an element that's zero.

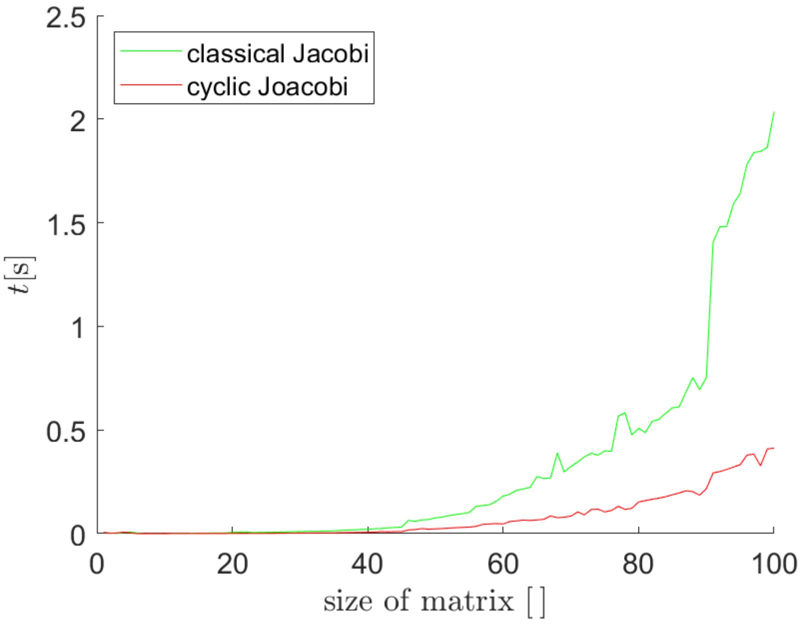


Figure 9: Comparison of time taken by classical and cyclical Jacobi algorithm to diagonalize a matrix of size N

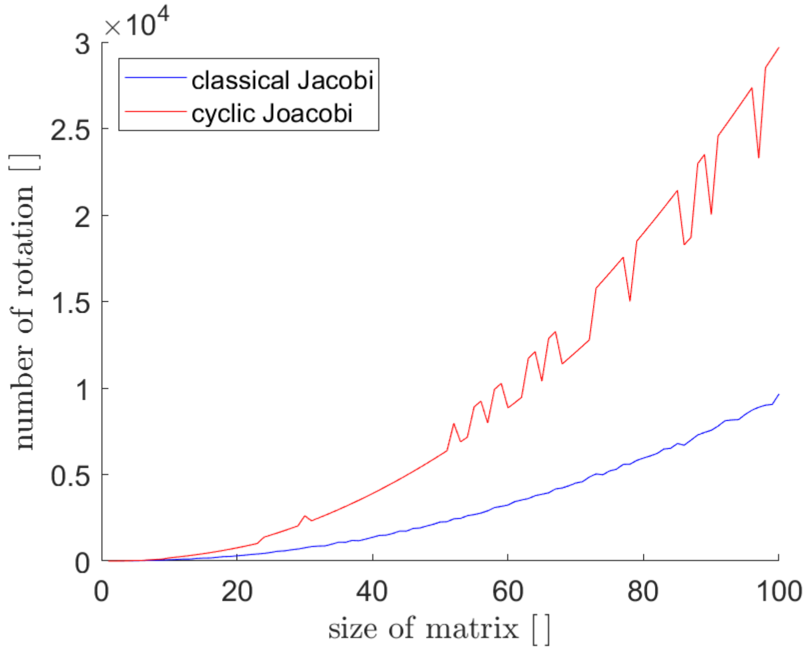


Figure 10: Number of rotations needed to diagonalize a matrix of size N

Problem 9

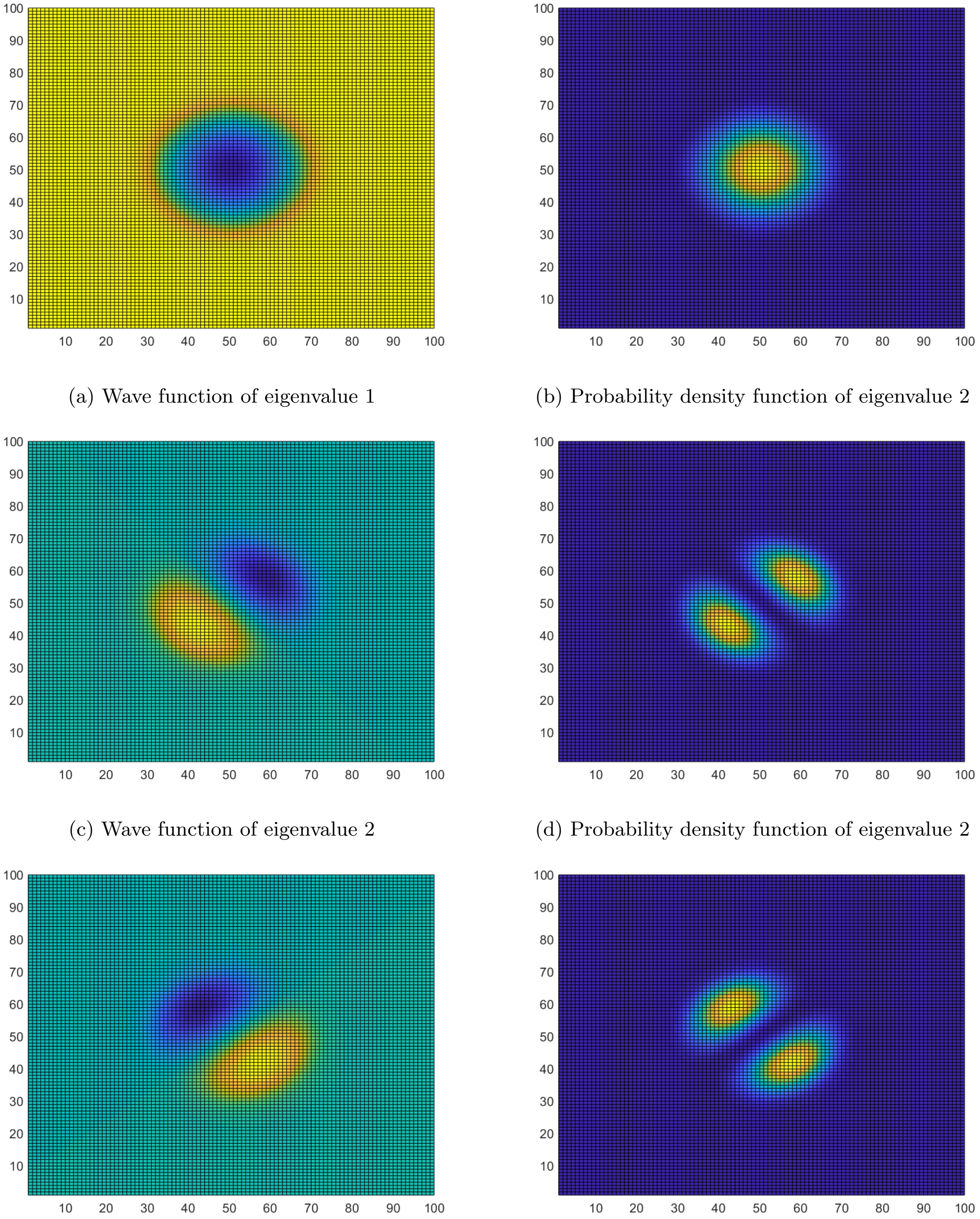
Two-dimensional quantum well In this exercise a two dimensional quantum well is considered. A square surface of $a \times a$ is discretised and Schrodinger's equation is solved on it. The potential well is given by :

$$\begin{cases} V_0 & \text{if } x^2 + \frac{y^2}{c^2} < r^2 \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

First the case $c = 1$ is considered. Ten bound states were found for this case. The three lowest eigen-energies are :

$$-1.3371\,\mathrm{eV}$$
$$-1.0900\,\mathrm{eV}$$
$$-1.0900\,\mathrm{eV}$$

The three corresponding eigenvectors can be visualized on figure 11.



(a) Wave function of eigenvalue 1

(b) Probability density function of eigenvalue 2

(c) Wave function of eigenvalue 2

(d) Probability density function of eigenvalue 2

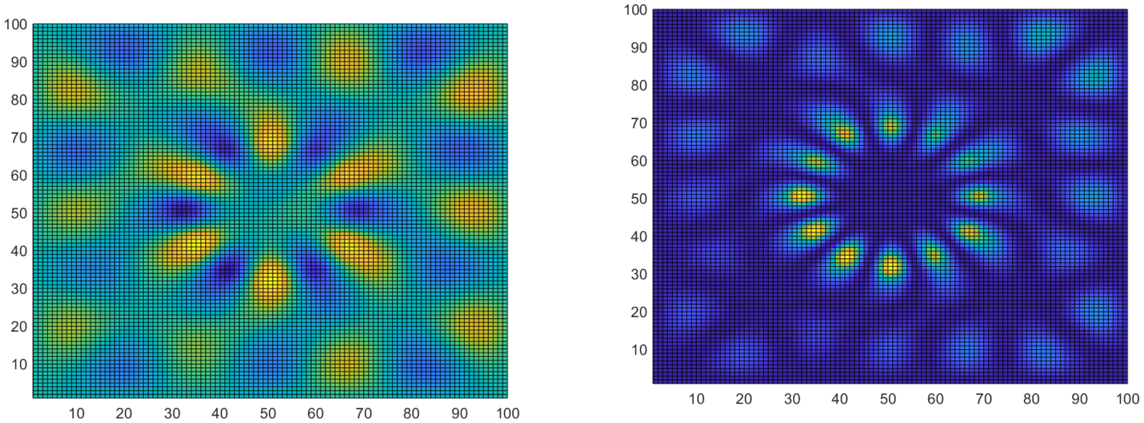
(e) Wave function of eigenvalue 3

(f) Probability density function of eigenvalue 3

Figure 11: Image of wave function and probability density function of three lowest energy eigenenergy

The eigenwaves of the three lowest energy eigenstates are localized in the well as expected. The lowest energy wave function has a higher probability density in the center of the well than on it's side where it rapidly decreases. The second and the third lowest eigen-energies have the same energy which makes sense since the problem has a radial symmetry and the two wave functions are simply rotated with respect to each other. The second and the third wave functions have a wave vector double as big as the first wave function in one of their diagonal direction.

The scattering solution having an energy of 1.0034 eV was investigated. The corresponding wave function and probability function can be seen on figure 12. The probability of finding electrons in the first, second and third energy eigenstates as well as in the eigenstate of energy 1.0034 eV is 0.9843, 0.9575, 0.9575 and 0.2700 respectively. These probabilities where calculated by integrating the normalized probability density functions over the inside surface of the potential well. As expected the scattering eigenstate has a much lower probability to be in the quantum well as the bound states. We also notice that the second and third eigenstates have the same probability to be in the well. As explained before this is due to the radial symmetry of the problem.



(a) Wave function of eigenvalue $E = 1.0178\,\text{eV}$

(b) Probability density function of eigenvalue $E = 1.0178\,\text{eV}$

Figure 12: Image of wave function and probability density function of energy $E = 1.0178\,\text{eV}$

The case where $c > 1$ was considered and the number of bound states was plotted as a function of c on figure 13. The number of bound states increases with c which makes sense because the area covered by the well also increases with the c. After some point the sides of the well go past the border of the considered squared area and the number of bound states don't increase as rapidly with c.

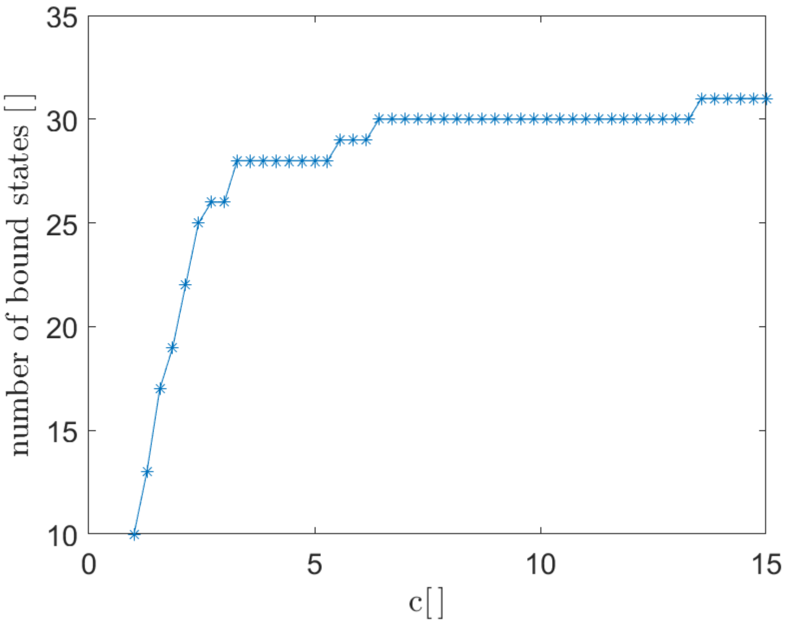


Figure 13: Number of bound states as a function of the the parameter c linked to the ellipticity of the well

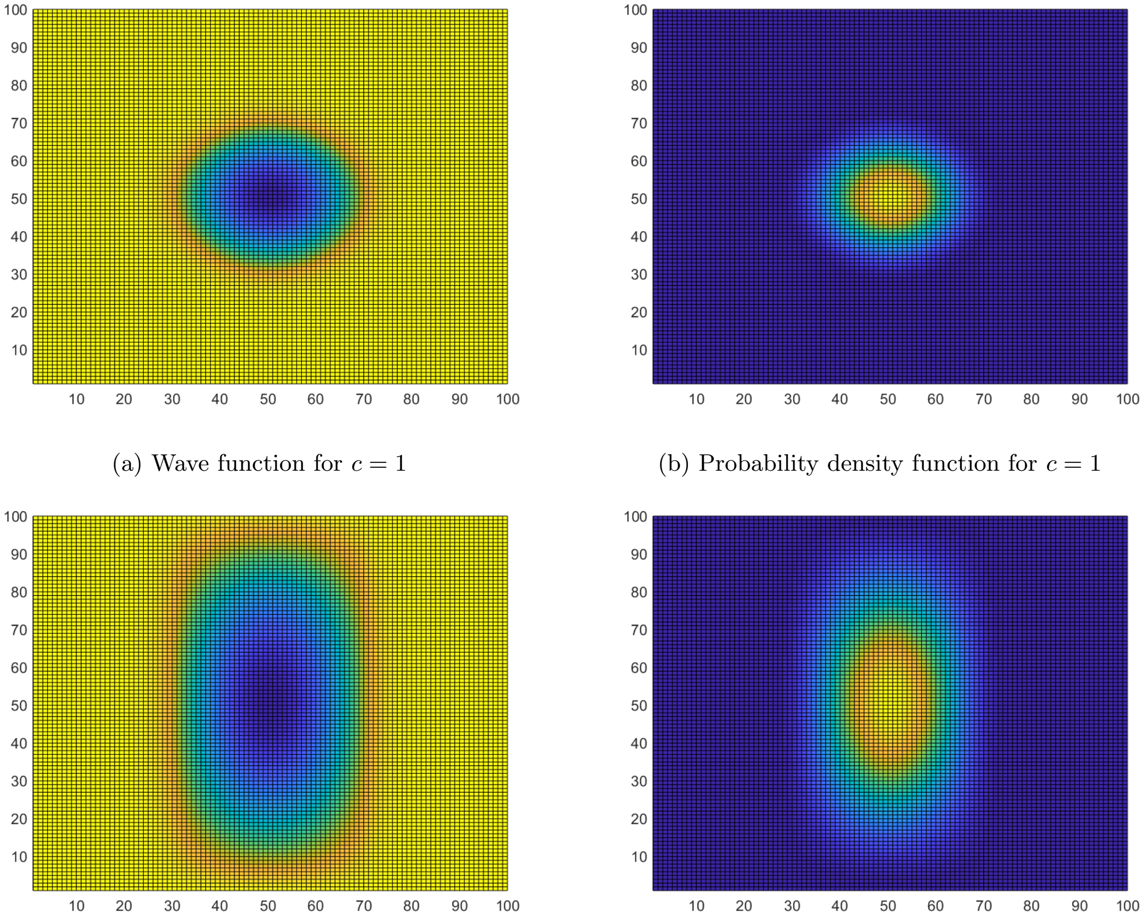
The three eigenstates of lowest energy for $c = 1$ and $c = 25$ were compared. The eigen-energies for $c = 25$ are:

- $1.4172\,\mathrm{eV}$
- $1.3721\,\mathrm{eV}$
- $1.2969\,\mathrm{eV}$

whereas the eigen-energies for $c = 1$ are:

- $1.3371\,\mathrm{eV}$
- $1.0900\,\mathrm{eV}$
- $1.0900\,\mathrm{eV}$

One can notice that the lowest eigen-energies for $c = 25$ are smaller and closer to one another in the $c = 25$ case compared to the $c = 1$ case. Next the eigenvectors of these same states are compared. The first second and third eigen functions and probability densities are compared on figures 14, 15 and 16 respectively. The first eigen function changes as expected transforming into more of an elliptical shape for $c = 1$ whereas it is perfectly circular for $c = 1$. The probability density function vanishes on the upper and lower part of the image, not because of the potential well but because of the boundary condition of the aa square box. The second eigen function for the $c = 25$ case also has two distinct spots where the probability of presence is great just like in $c = 1$ case but this time around it's orientation is different, it is along the long axis of the ellipse. The third eigen function is qualitatively different for the $c = 25$ case. It has three spots where the probability of presence is greater whereas in the $c = 1$ case the eigen function still only has two spots. It seems it is more energy efficient to increase the the wave vector along the axis of the ellipse than to increase it in the direction perpendicular to that. This is due to the fact that the potential well is much longer in one direction that in the perpendicular direction.



(a) Wave function for $c = 1$

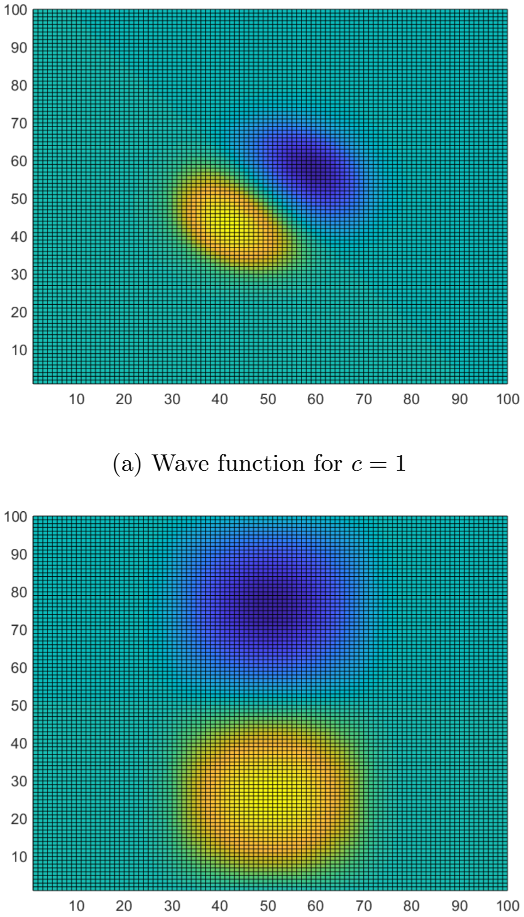
(b) Probability density function for $c = 1$

(c) Wave function for $c = 25$

(d) Probability density function for $c = 25$

Figure 14: Comparison of lowest energy wave function and probability density function for $c = 1$ and $c = 25$

Problem 9 continued on next page. . .



(a) Wave function for $c = 1$

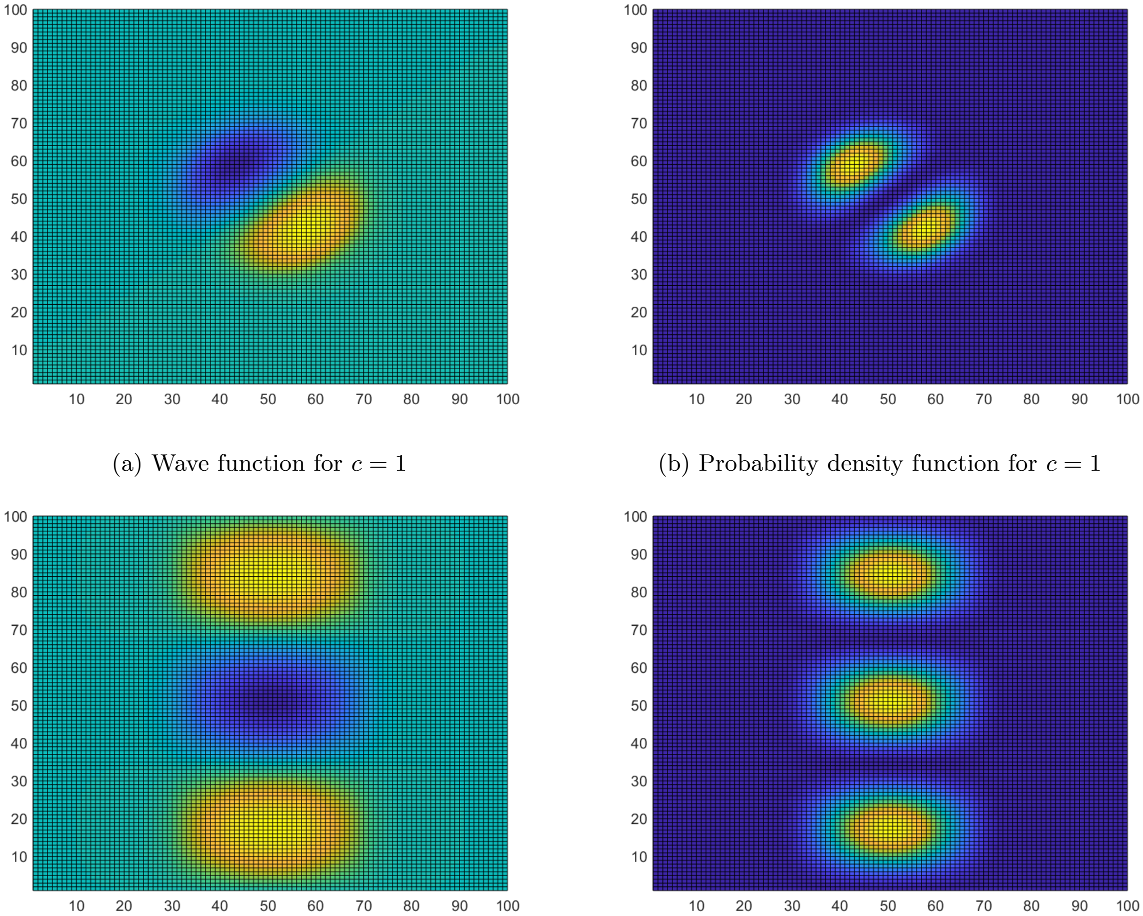
(b) Probability density function for $c = 1$

(c) Wave function for $c = 25$

(d) Probability density function for $c = 25$

Figure 15: Comparison of lowest energy wave function and probability density function for $c = 1$ and $c = 25$

Problem 9 continued on next page. . .



(a) Wave function for $c = 1$

(b) Probability density function for $c = 1$

(c) Wave function for $c = 25$

(d) Probability density function for $c = 25$

Figure 16: Comparison of lowest energy wave function and probability density function for $c = 1$ and $c = 25$