

# Sprawozdanie

## Metody numeryczne Laboratorium 4.

*Diagonalizacja macierzy operatora energii w 2D*

01.04.2020 r.

Aleksandra Rolka

Celem 4. laboratorium było znalezienie numerycznego rozwiązania niezależnego od czasu równania Schrödingera w dwóch wymiarach.

Często przy tworzeniu modeli matematycznych wykorzystywanych do symulacji zjawisk fizycznych czy zachowania się układu, zachodzi potrzeba rozwiązania tzw. problemu własnego. Matematycznie równanie Schrödingera niezależne od czasu ma właśnie postać tzw. równania własnego energii.

Otrzymane rozwiązania nazywa się wartościami własnymi  $E$  oraz funkcjami własnymi  $\psi$  operatora Hamiltona.

## 1. Wstęp teoretyczny

Wektor  $x$  jest **wektorem własnym** macierzy  $A$  jeśli istnieje taka liczba  $\lambda$ , że

$$Ax = \lambda x, \quad (1)$$

$\lambda$  - wartość własna macierzy  $A$ .

Liczba  $\lambda$  jest **wartością własną** macierzy  $A$  wtedy i tylko wtedy, gdy jest pierwiastkiem wielomianu charakterystycznego  $\det(A - \lambda I)$  macierzy  $A$  ( $I$ -macierz jednostkowa). Wynika to z następującego przekształcenia:

$$\begin{aligned} Ax - \lambda x &= 0 \\ (A - I\lambda)x &= 0 \\ \det(A - I\lambda) &= 0 \end{aligned} \quad (2)$$

Wyznaczając wartości własne i wektory własne macierzy można wykorzystać **metodę Householdera**, która redukuje macierz do postaci trójdzielnej.

## 2. Zadanie do wykonania

### 2.1 Opis problemu

Zadanie: znaleźć numeryczne rozwiązanie niezależne od czasu równania Schrödingera

$$H \psi(t) = E \psi(t) \quad (3)$$

w dwóch wymiarach,

gdzie

$H$  – operator Hamiltona (operator energii całkowitej układu, w skrócie nazywany hamiltonianem)  
 $\psi(t)$  – wektor stanu układu, funkcja falowa

Postać operatora energii :

$$H = -\frac{\hbar^2}{2m} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right), \quad (4)$$

gdzie

$\hbar$  – stała Plancka

$m$  – masa

Przy wprowadzeniu siatki węzłów:  $x_i = \Delta \cdot i, i = 1, 2, \dots, n_x$  oraz  $y_j = \Delta \cdot j, j = 1, 2, \dots, n_y$ , dyskretyzuje się równanie własne na tej siatce zastępując drugie pochodne ilorazami różnicowymi  $\psi(x, y) = \psi(x_i, y_j) = \psi_{i,j}$  :

$$H\psi = E\psi \Rightarrow -\frac{\hbar^2}{2m} \left( \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{\Delta^2} + \frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{\Delta^2} \right) = E\psi_{i,j}. \quad (5)$$

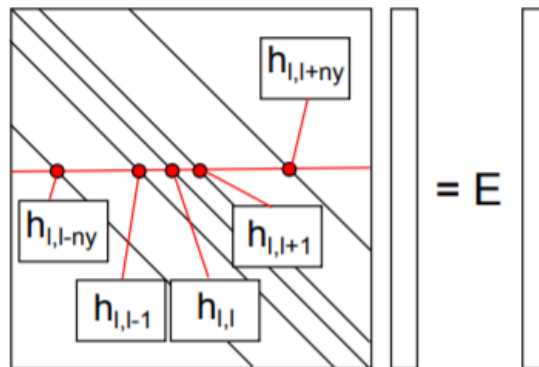
Dzięki dokonaniu reindeksacji:  $l = j + (i - 1) \cdot n_y, l = 1, 2, \dots, n, n = n_x \cdot n_y$  oraz wprowadzeniu współczynnika  $t = -\frac{\hbar^2}{2m\Delta^2}$ , równanie przyjmuje prostszą postać:

$$H\psi = t(\psi_{l-n_y} + \psi_{l-1} - 4\psi_l + \psi_{l+1} + \psi_{l+n_y}). \quad (6)$$

Zapisując operator  $H$  jako macierz kwadratową  $n \times n$  to jedyne elementy niezerowe w wierszu mają postać:

$$H_{l,l\pm n_y} = H_{l,l\pm 1} = t, \quad H_{l,l} = -4t, \quad (7)$$

więc macierz  $H$  jest pięcioprzekątniowa jak pokazuje rysunek poniżej (rys.1) :



Rys 1. Postać macierzy operatora energii dla problemu własnego  $H\psi = E\psi$  w 2D.

Następnym krokiem jest diagonalizacja tej macierzy.

Przyjęte parametry:

- $n = n_x * n_y$ ,
- $n_x = 20$ ,
- $n_y = 20$ ,
- $m = 10$ ,
- $t = -0.021$

W programie rozwiązującym dane zagadnienie korzystano z biblioteki *Numerical Recipes (NR)*.

Do rozwiązania problemu stworzono macierze  $H_{n \times n}$ ,  $Y_{n \times n}$ ,  $X_{n \times n}$  oraz wektory  $d$  i  $e$ .

Uwzględniono również fakt, że węzły znajdujące się na brzegach rozpatrywanego obszaru mają mniej sąsiadów niż węzły położone wewnątrz, dlatego elementy macierzy  $H$  wypełniono według poniższego algorytmu:

```
for ( i = 1; i <= n_x; i++){
    for ( j = 1; j <= n_y; j++){
        l = j + (i - 1) * n_y;
        for ( k = 1; k <= n; k++)
            H[l][k] = 0.;
        if(i > 1)
            H[l][l - n_y] = t;    //dla i=1 nie ma sąsiada z lewej strony
        if(i < n_x)
            H[l][l + n_y] = t;    //dla i=n_x nie ma sąsiada z prawej strony
        H[l][l] = -4 * t;
        if(j > 1)
            H[l][l - 1] = t;    //dla j=1 nie ma sąsiada poniżej siatki
        if(j < n_y)
            H[l][l + 1] = t;    //dla j=n_y nie ma sąsiada powyżej siatki
    }
}
```

Następnie macierz  $H$  przekształcono do postaci trójdagonalnej, przy użyciu procedury *tred2* z biblioteki numerycznej :

$$tred2(H, n, d, e), \quad (8)$$

gdzie

- $H_{n \times n}$  - macierz układu,
- $n$  - ilość wierszy/kolumn w  $H$ ,
- $d$  - wektor  $n$ - elementowy
- $e$  - wektor  $n$ - elementowy

Procedura ta zwraca macierz trójdziagonalną ( $T$ ) zapisaną w postaci wektorów  $d$  i  $e$ , gdzie wektor  $d$  jest diagonalą, a wektor  $e$  pierwszą poddiagonalą  $T$ .

Kolejnym krokiem jest diagonalizacja macierzy  $T$  :

$$T \cdot y_k = \lambda_k \cdot y_k, \quad (9)$$

przy pomocy procedury *tqli* :

$$tqli(d, e, n, Y), \quad (10)$$

gdzie  $Y$  jest macierzą jednostkową wymiaru  $n \times n$ , którą procedura *tqli* przekształca i w jej kolumnach zwróci wektory własne macierzy  $T$ , a wartości własne zapisane zostają w wektorze  $d$ .

Następnie trzeba odtworzyć wektory własne pierwotnego problemu. Aby przekształcić wszystkie wektory wykonano mnożenie dwóch macierzy:

$$X = P \cdot Y, \quad (11)$$

gdzie

$Y$  – macierz, w kolumnach której zapisane są wektory  $y_k$ ,

$P$  – macierz przekształcenia (otrzymana z procedury *tred2*)

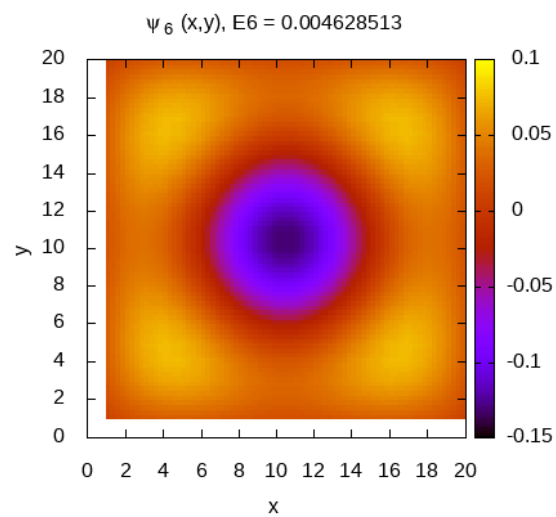
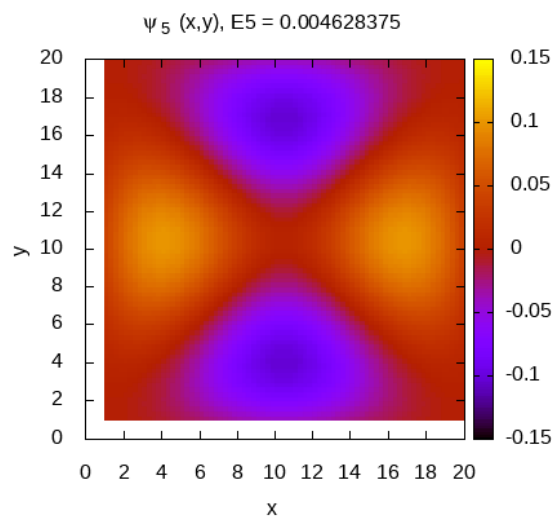
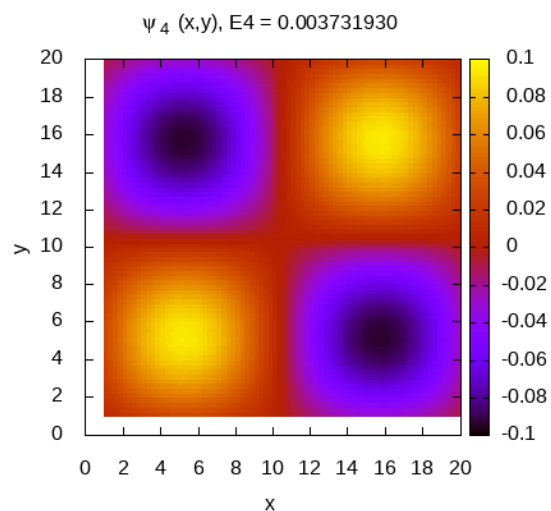
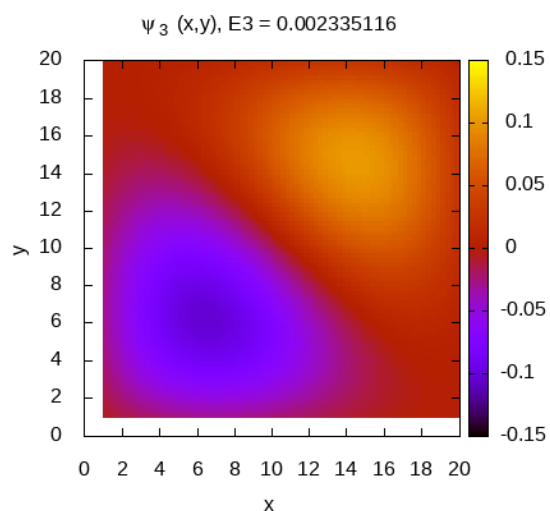
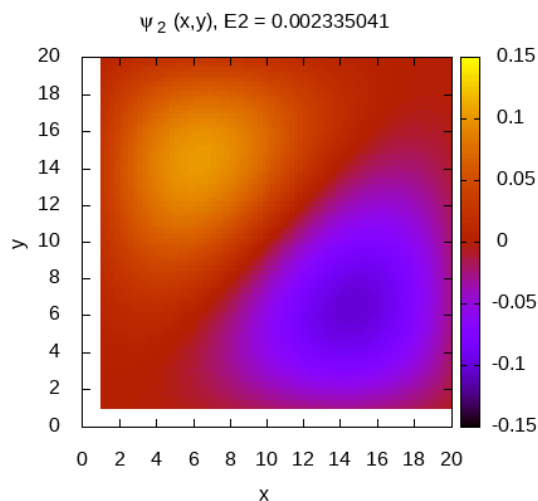
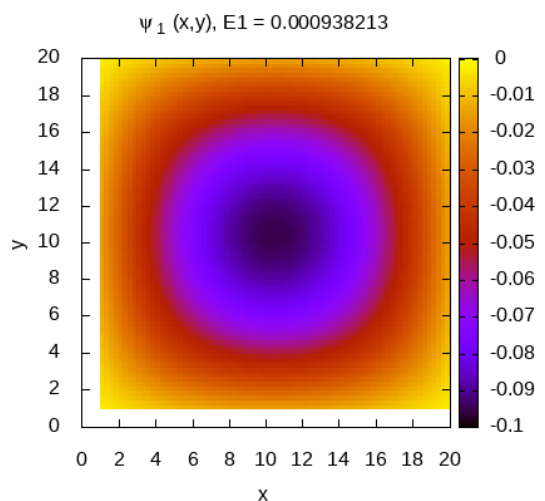
Procedura *tred2* zwróciła nieposortowane wektory własne i wartości własne, dlatego dokonano sortowania energii oraz indeksów wektorów według algorytmu (tablica *indx*):

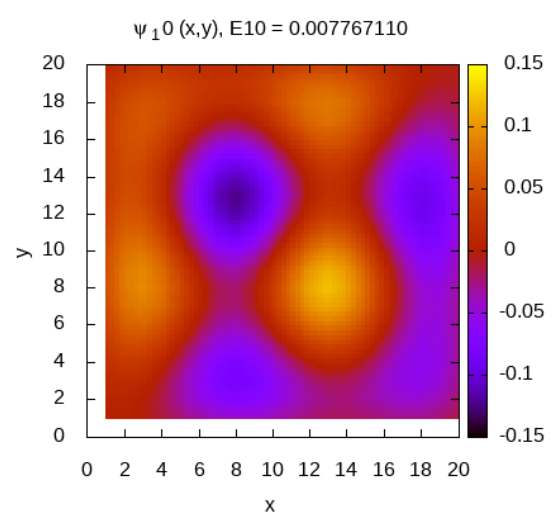
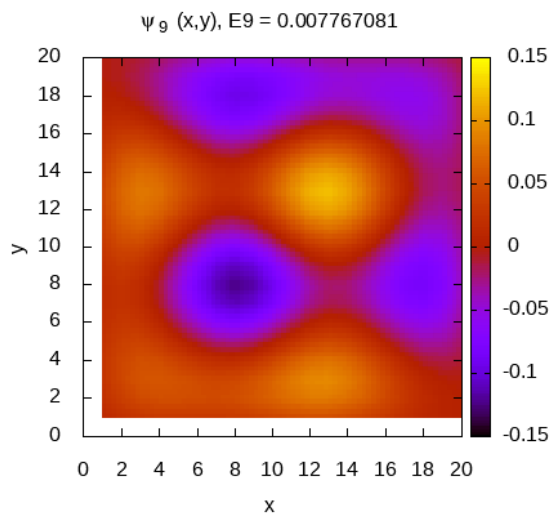
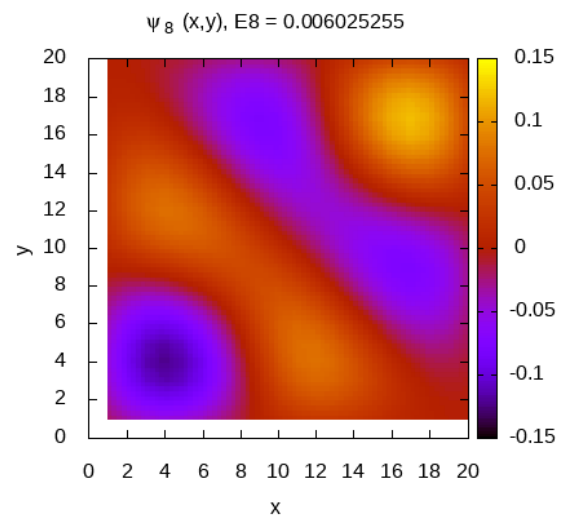
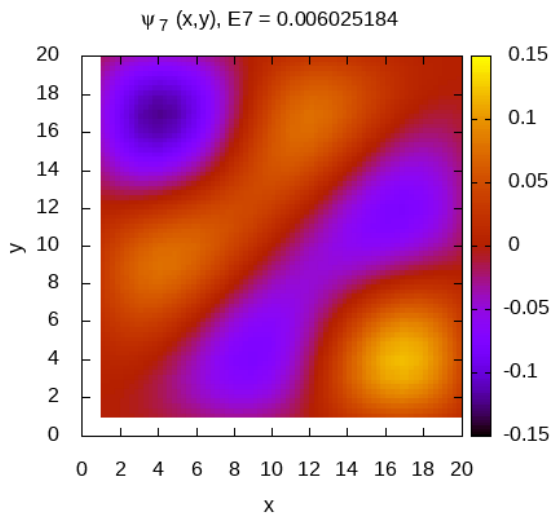
```
for ( l = 1; l <= n; l++)
    indx[l] = l;                // inicjalizacja
for ( l = 1; l <= n - 1; l++){
    for ( k = n; k >= l + 1; k--){
        e1 = d[k-1];
        e2 = d[k];
        l1 = indx[k - 1];
        l2 = indx[k];
        if (e2 < e1){            //wymienienie energii i indeksów wektorów miejscami
            d[k] = e1;
            d[k - 1] = e2;
            indx[k] = l1;
            indx[k-1] = l2;
        }
    }
}
```

przez co wartości własne zostały posortowane od najmniejszej do największej w tablicy  $d$  i odpowiadają im wektory własne, których indeksy zostały wpisane do kolejnych komórek tablicy *indx*.

## 2.2 Wyniki

Otrzymane wyniki programu zapisane zostały w pliku *dane.dat*, a następnie za pomocą stworzonego skryptu Gnuplot'a wygenerowano wykresy kolejnych 10 rozwiązań w postaci funkcji falowych 2D:





Powyższe rysunki: Wektory własne macierzy  $H$  odpowiadające dziesięciu najniższym wartościom własnym (funkcje falowe hamiltonianu dla cząstki w dwuwymiarowym kwadratowym pudle potencjału). W podpisach nad wykresami zamieszczono wartości własne (energje poszczególnych stanów).

### 3. Wnioski

Niektóre (zaokrąglone) wartości własne pojawiają się dwukrotnie, jednak odpowiadające im wektory własne nie są identyczne. W większości przypadków wykresy funkcji o podobnych wartościach własnych wyglądają bardzo podobnie, ale różnią się orientacją, np. rysunek jednej funkcji jest odbiciem symetrycznym drugiej o takiej samej wartości własnej.