Diagonalizacja macierzy operatora energii w 2D

Tomasz Chwiej

22 marca 2017

1 Wprowadzenie

Naszym celem jest znalezienie numerycznego rozwiązania niezależnego od czasu równanie Schrödingera

$$H\,\psi = E\,\psi \tag{1}$$

w dwóch wymiarach. Postać operatora energii jest następująca

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

W tym celu wprowadzamy siatkę węzłów: $x_i = \Delta \cdot i$, $i = 1, 2, \ldots, n_x$ oraz $y_j = \Delta \cdot j$, $j = 1, 2, \ldots, n_y$. Następnie dyskretyzujemy równanie własne na siatce zastępując drugie pochodne ilorazami różnicowymi:

$$\psi(x, y) = \psi(x_i, y_j) = \psi_{i,j} \tag{3}$$

$$H\psi = E\psi \Longrightarrow \quad -\frac{\hbar^2}{2m^*}\left(\frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{\Delta^2} + \frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{\Delta^2}\right) = E\psi_{i,j} \tag{4}$$

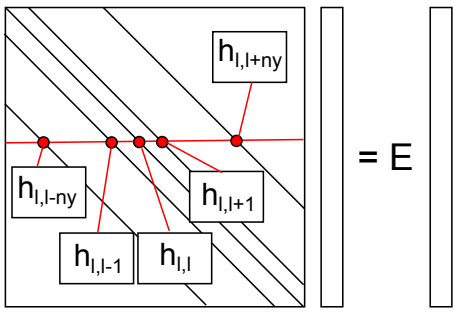
Dokonujemy teraz reindeksacji: $l = j + (i-1)\cdot n_y$, $l = 1, 2, \ldots, n$, $n = n_x \cdot n_y$ oraz wprowadzamy współczynnik $t = -\frac{\hbar^2}{2m\Delta^2}$, dzięki czemu równanie przyjmuje prostszą postać:

$$H\psi = t\,(\psi_{l-ny} + \psi_{l-1} - 4\psi_l + \psi_{l+1} + \psi_{l+ny}) \tag{5}$$

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

$$H_{l,l\pm n_y} = H_{l,l\pm 1} = t, \quad H_{l,l} = -4t \tag{6}$$

więc macierz H jest pięcioprzekątniowa jak na rysunku poniżej (rys.1)

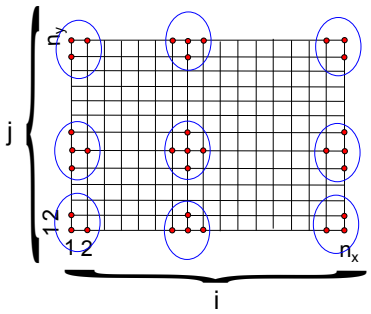


Rysunek 1: Postać macierzy operatora energii dla problemu własnego $H\psi = E\psi$ w 2D.

Naszym celem jest jej diagonalizacja.

2 Zadania do wykonania:

1. Przyjmujemy następujące parametry: $n = n_x * n_y$, $n_x = 20$, $n_y = 20$, $m = 10$, $t = -0.021$.
2. Tworzymy macierze: $H_{n\times n}$, $Y_{n\times n}$, $X_{n\times n}$ oraz wektory n-elementowe $\boldsymbol{d}$ i $\boldsymbol{e}$.
3. Aby wypełnić elementy H musimy uwzględnić fakt że węzły znajdujące się na brzegach rozpatrywanego obszaru mają mniej sąsiadów niż węzły położone wewnątrz (rysunek 2)



Rysunek 2: Dwuwymiarowa siatka na której szukamy rozwiązania.

dlatego elementy macierzy wypełniamy stosując poniższy kod:

```
for(i=1;i<=nx;i++){
      for(j=1;j<=ny;j++){

                  l=j+(i-1)*ny;

                  for(k=1;k<=n;k++)H[l][k]=0.;

                  if(i>1)   H[l][l-ny]=t;   //dla i=1 nie ma sasiada z lewej strony
                  if(i<nx)  H[l][l+ny]=t;   //dla i=nx nie ma sasiada z prawej strony
                            H[l][l]=-4*t;
                  if(j>1)   H[l][l-1]=t;    //dla j=1 nie ma sasiada ponizej siatki
                  if(j<ny)  H[l][l+1]=t;    //dla j=ny nie ma sasiada powyzej siatki
    }
}
```

4. Przekształcamy macierz H do postaci trójdiagonalnej

$$P^{-1}HP = T \tag{7}$$

przy użyciu procedury **tred2**:

$$tred2(H, n, d, e)$$

gdzie: $H_{n\times n}$ jest macierzą układu, n-ilość wierszy/kolumn w H, $\boldsymbol{d}$ i $\boldsymbol{e}$ to wektory n- elementowe. Procedura **tred2** zwraca macierz trójdiagonalną (T) zapisaną w postaci wektorów $\boldsymbol{d}$ i $\boldsymbol{e}$. Wektor $\boldsymbol{d}$ jest diagonalą, a wektor $\boldsymbol{e}$ pierwszą poddiagonalą T.
Uwaga: Na wyjściu macierz $H = P$ tzn. zostaje ona nadpisana przez macierz podobieństwa P.

5. Diagonalizujemy macierz T

$$T \cdot \boldsymbol{y}_k = \lambda_k \cdot \boldsymbol{y}_k \tag{8}$$

używając procedury **tqli**:

$$tqli(d, e, n, Y)$$

Jeśli do procedury przekażemy $Y_{n\times n} = I_{n\times n}$ to procedura zwróci w kolumnach macierzy Y wektory własne T, a wartości własne zapisane są w wektorze $\boldsymbol{d}$. Uwaga: wartości i wektory własne nie są posortowane.

6. Odtwarzamy wektory własne pierwotnego problemu (dla ułatwienia zapiszmy je jako $Hx_k = \lambda_k x_k$, gdzie k numeruje wartości i wektory własne). Poniżej wyjaśnienie:

$$
\begin{aligned}
T &= P^{-1}AP && (9)\\
Ty_k &= \lambda \boldsymbol{y}_k && (10)\\
P^{-1}AP\boldsymbol{y}_k &= \lambda \boldsymbol{y}_k \qquad P\cdot / && (11)\\
A(P\boldsymbol{y}_k) &= \lambda(P\boldsymbol{y}_k) && (12)\\
A\boldsymbol{x}_k &= \lambda \boldsymbol{x}_k && (13)\\
\boldsymbol{x}_k &= P\boldsymbol{y}_k && (14)
\end{aligned}
$$

Czyli, jeśli chcemy przekształcić wszystkie wektory to wykonujemy mnożenie dwóch macierzy:

$$X = P \cdot Y \tag{15}$$

gdzie: Y to macierz, w której kolumnach zapisane są wektory $\boldsymbol{y}_k$, a macierz P to macierz przekształcenia (którą dostajemy z **tred2**).

7. Ponieważ wektory i wartości własne nie są posortowane, dokonujemy sortowania energii oraz indeksów wektorów (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){  //wymieniamy energie i indeksy wektorów miejscami
           d[k]=e1;
           d[k-1]=e2;
           indx[k]=l1;
           indx[k-1]=l2;
         }
  }
}
```

teraz wartości własne są posortowane od najmniejszej do największej w tablicy $\boldsymbol{d}$, a odpowiadają im wektory własne których indeksy wpisane są do kolejnych komórek tablicy **indx**.

8. Zapisujemy wektory własne do pliku uwzględniając pierwotne indeksy, czyli $l \to (i, j)$

```
FILE *fp;
fp=fopen("dane.dat","w");
        for(i=1;i<=nx;i++){
                for(j=1;j<=ny;j++){
```

```
                        l=j+(i-1)*ny;
                        fprintf(fp,"%6d  %6d ",i,j);
                        for(k=1;k<=m;k++)fprintf(fp," %12.6f  ",x[l][ indx[k] ]);
                        fprintf(fp,"\n");
                }
                 fprintf(fp,"\n");
        }
fclose(fp);
```

9. Rysujemy kolejne rozwiązania w postaci map (funkcje falowe w 2D) w Gnuplocie przy użyciu skryptu:

```
set term png
set view map
set pm3d interpolate 4,4
set out 'vec_NUMER.png'
splot 'dane.dat' u 1:2:NUMER w pm3d
```

   gdzie: $NUMER = 3, 4, \ldots$ to kolumny w których wpisane są wektory własne odpowiadające najniższym energiom (wartościom własnym).

10. Do pliku proszę zapisać m wartości własnych. Ponadto proszę narysować pierwsze m rozwiązań (wektorów własnych/ funkcji falowych). Niektóre wartości własne pojawiają się dwukrotnie, czy odpowiadające im wektory (funkcje) własne są identyczne?