Rozwiązywanie równania Poissona w 1D przy użyciu baz funkcyjnych - metody: kolokacji,najmniejszych kwadratów, Galerkina

Tomasz Chwiej

24 listopada 2017

1 Wstęp

Chcemy znaleźć rozwiązanie RRZ:

$$Lu = f \quad \Longrightarrow \quad \frac{d^2u}{dx^2} + \frac{2}{x}\frac{du}{dx} = -x^2 \cdot sin(x) + 6 \cdot x \cdot cos(x) + 6 \cdot sin(x) \tag{1}$$

w przedziale $x \in [0, 2\pi]$. Dla warunków brzegowowych: $u(0) = u(2\pi) = 0$ jego rozwiązaniem (dokładnym) jest funkcja:

$$u_d(x) = x^2 \cdot sin(x) \tag{2}$$

Jeśli zdefiniujemy pewną bazę funkcyjną $\{v_i\}$ to rozwiązanie (numeryczne) możemy wyrazić jako:

$$u_n(x) = \sum_{i=1}^{N} c_i v_i(x) \tag{3}$$

Niezależnie od tego jaką metodę zastosujemy: kolokacji, najmniejszych kwadratów czy Galerkina, rozwiązanie problemu sprowadza się zawsze do rozwiązania układu równań:

$$Ac = b \tag{4}$$

Elementy macierzy A ($a_{k,i}$) oraz wektora b liczymy następująco:

- w metodzie kolokacji

$$a_{k,i} = (Lv_i)(x_k) \tag{5}$$
$$b_k = f(x_k) \tag{6}$$

  gdzie: x_k położenie k-tego węzła

- w metodzie najmniejszych kwadratów

$$a_{k,i} = (Lv_k, Lv_i) = \int_0^{2\pi} Lv_k \cdot Lv_i \, dx \tag{7}$$
$$b_k = (f, Lv_k) = \int_0^{2\pi} f \cdot Lv_k \, dx \tag{8}$$

- w metodzie Galerkina

$$a_{k,i} = (v_k, Lv_i) = \int_0^{2\pi} v_k \cdot Lv_i \, dx \tag{9}$$
$$b_k = (f, v_k) = \int_0^{2\pi} f \cdot v_k \, dx \tag{10}$$

Łatwo zauważyć, że rozwiązanie dla każdej z powyższych metod można uzyskać wykorzystując ten sam algorytm, zmiejniając tylko sposób wyznaczania elementów $a_{k,i}$ oraz b_k.

2 Zadania do wykonania

1. Definiujemy bazę I w postaci: $v_i(x) = sin(i \cdot x/2), \quad i = 1, 2, \ldots, N$, która spełnia warunki brzegowe.

2. Rozwiązać równanie różniczkowe metodą kolokacji, dla $N = 6, 7, 8, 9, 10$, oraz węzłów rozłożonych zgodnie z wzorem $x_k = k \cdot \Delta x$, gdzie: $\Delta x = 2\pi/(N+1)$ oraz $k = 1, 2, \ldots, N$. Na jednym rysunku narysować różnicę rozwiązania dokładnego i numerycznego $\Delta u(x) = u_d(x) - u_n(x)$ dla wszystkich wartości N. (50 pkt.)

3. Rozwiązać równanie różniczkowe metodą najmniejszych kwadratów, dla $N = 6, 7, 8, 9, 10$. Na jednym rysunku narysować różnicę rozwiązania dokładnego i numerycznego $\Delta u(x) = u_d(x) - u_n(x)$ dla wszystkich wartości N. (20 pkt.)

4. Rozwiązać równanie różniczkowe metodą Galerkina, dla $N = 6, 7, 8, 9, 10$. Na jednym rysunku narysować różnicę rozwiązania dokładnego i numerycznego $\Delta u(x) = u_d(x) - u_n(x)$ dla wszystkich wartości N. (20 pkt.)

5. Powtórzyć obliczenia dla trzech metod, korzystając z bazy II: $v_i(x) = x \cdot (x - 2\pi) \cdot (x - \pi)^{i-1}, \quad i = 1, 2, \ldots, N$ (10 pkt.)

3 Uwagi

1. Pochodne proszę liczyć stosując ilorazy różnicowe:

$$\frac{du}{dx} = \frac{u(x + \Delta x) - u(x - \Delta x)}{2\Delta x} \tag{11}$$

$$\frac{d^2u}{dx^2} = \frac{u(x + \Delta x) - 2u(x) + u(x - \Delta x)}{\Delta x^2} \tag{12}$$

   z krokiem $\Delta x = 0.001$

2. Całki proszę liczyć numerycznie stosując kwadraturę Gaussa-Legendre'a:

$$\int_{-\pi}^{\pi} g(x)dx = \sum_{i=1}^{n} w_i \cdot g(x_i) \tag{13}$$

   gdzie: x_i oraz w_i to położenia węzłów i współczynniki kwadratury, które można wygenerować przy użyciu procedur z Numerical Recipes lub GSL. W obliczeniach zastosować $n = 40$.

   Przykład całkowania przy użyciu GSL-a w C:

```
#include <gsl/gsl_math.h>
#include<gsl/gsl_integration.h>
 //calkowanie Lvk_Lvi
double Lvk_Lvi(int k,int i){
        double pi=4.*atan(1.0);
        double calka,a,b,xi,wi;
        double dx=0.0001;
        int j;
        size_t n=40;
        a=-pi;
        b=pi;
        gsl_integration_glfixed_table *tab= gsl_integration_glfixed_table_alloc(n);

        calka=0.;
        for(j=0;j<n;j++){
                gsl_integration_glfixed_point(a,b,j,&xi,&wi,tab);
                calka=calka+wi*Lvi(k,xi,dx)*Lvi(i,xi,dx);
        }
        gsl_integration_glfixed_table_free(tab);
        return calka;
}
```

3. Aby łatwiej porównać wyniki dla różnych baz, można wszystkie wykresy umieścić na jednym rysunku np. w Gnuplocie wykorzystując otoczenie **multiplot:**

```
set term postscript color enhanced solid
set size square
set size 1.,1.

set out 'wyniki.eps'

set multiplot layout 2,3 rowsfirst

  set title 'baza 1-kolokacja'
  plot 'zad1.dat' u 1:7 w l t 'N=6'

  set title 'baza 1-least squares'
  plot 'zad2.dat' u 1:7 w l t 'N=6'

  set title 'baza 1-galerkin'
  plot 'zad3.dat' u 1:7 w l t 'N=6'



  set title 'baza 2-kolokacja'
  plot 'zad1_2.dat' u 1:7 w l t 'N=6'

  set title 'baza 2-least squares'
  plot 'zad2_2.dat' u 1:7 w l t 'N=6'

  set title 'baza 2-galerkin'
  plot 'zad3_2.dat' u 1:7 w l t 'N=6'

unset multiplot
```