Szymon Ociepka

Data wykonania:
27.03.2024

CAŁKOWANIE NUMERYCZNE

metoda Prostokątów, Trapezów, Simpsona i MC

Metody numeryczne

1. Wprowadzenie teoretyczne

1.1 metoda Prostokątów

Całkowanie numeryczne metodą prostokątów to jedna z najprostszych technik numerycznego przybliżania wartości całek oznaczonych. Polega na podziale obszaru pod krzywą na wiele małych prostokątów, a następnie sumowaniu ich powierzchni. Mając przedział [a, b] dzielimy go na n równych części. Każda część ma szerokość $dx=\frac{b-a}{n}$. Na końcu zsumujemy powierzchnie wszystkich prostokątów, aby uzyskać przybliżenie całki:

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n-1} f(x_i)\cdot dx \tag{1}$$

Warto zauważyć, że metoda prostokątów jest dość prosta, ale nie zawsze daje precyzyjne wyniki, zwłaszcza dla funkcji, które znacznie się zmieniają na danym przedziale. Dlatego w praktyce często stosuje się bardziej zaawansowane metody, takie jak metoda trapezów czy metoda Simpsona, o których w dalszej części sprawozdania.

1.2 metoda Trapezów

Całkowanie numeryczne metodą trapezów to technika przybliżania wartości całek oznaczonych, która polega na podziale obszaru pod krzywą na wiele małych trapezów, a następnie sumowaniu ich powierzchni. Również jak w metodzie prostokątów dzielimy przedział [a, b] na n równych części. Każda część ma szerokość $dx=\frac{b-a}{n}$. Na końcu zsumujemy powierzchnie wszystkich trapezów, aby uzyskać przybliżenie całki:

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n-1} \frac{f(x_i)+f(x_{i+1})}{2}\cdot dx \tag{2}$$

Metoda trapezów jest bardziej precyzyjna niż metoda prostokątów, zwłaszcza dla funkcji, które są w przybliżeniu liniowe na danym przedziale. Jednak dla funkcji, które znacznie się zmieniają, mogą być potrzebne bardziej zaawansowane metody, takie jak metoda Simpsona.

1.3 metoda Simpsona

Całkowanie numeryczne metodą Simpsona to technika przybliżania wartości całek oznaczonych, która polega na przybliżeniu obszaru pod krzywą za pomocą parabolicznych “kawałków”. W tej metodzie funkcja podcałkowa jest przybliżana parabolą rozpiętą na dwóch końcach przedziału całkowania oraz jego środku.

$$\int_a^b f(x)dx \approx I = [f(a) + 4f(\frac{a+b}{2}) + f(b)]\frac{h}{3} \quad (3)$$

W celu uzyskania większej dokładności istnieje złożona metoda Simpsona, w której obliczany obszar dzielimy na mniejsze części i na nich stosujemy metodę Simpsona. Dla przedziału od x_1 do x_5 oraz szerokości między tymi punktami równej h:

$$I_1 = [f(x_1) + 4f(x_2) + f(x_3)]\frac{h}{3} \quad (4)$$

$$I_2 = [f(x_3) + 4f(x_4) + f(x_5)]\frac{h}{3} \quad (5)$$

1.4 metoda Monte Carlo

Metoda Monte Carlo polega na wylosowaniu n punktów z przedziału całkowania a następnie obliczenie średniej wartości funkcji w tym przedziale.

$$f_{średnie} = \frac{f(x_1)}{n} + \frac{f(x_2)}{n} + ... + \frac{f(x_n)}{n} \quad (6)$$

$$\int_a^b f(x)dx \approx I = f_{średnie} \cdot |x_k - x_p| \quad (7)$$

2. Opis implementacji numerycznej

2.1 metoda Prostokątów

Funkcja oblicza całkę oznaczoną metodą prostokątów. Przyjmuje jako parametry rozmiar przedziałów, dla których całka ma zostać oszacowana jak i zakres, dla którego ma zostać policzona całka.

```
void prostokaty(int n, double xp, double xk) {
    double dx = (xk - xp) / n;;

    double* fx = new double[n];

    double temp = dx;
    double wynik = 0;

    for (int i = 0; i < n; i++) {
        fx[i] = pow((xp + temp), 3) + 2;
        wynik += fx[i];
        temp += dx;
    }
    wynik *= dx;

    cout << "Metoda 1 = " << wynik << endl;
}
```

Rysunek 1: Fragment kodu (implementacja metody prostokątów)

Funkcja sumuje wartości obliczonej funkcji f, po czym mnoży sumę przez rozmiar przedziałów dx.

2.2 metoda Trapezów

Funkcja oblicza całkę oznaczoną metodą trapezów. Identycznie jak w przypadku metody prostokątów, funkcja ta przyjmuje jako parametry rozmiar przedziałów, dla których całka ma zostać oszacowana jak i zakres, dla którego ma zostać policzona całka.

```
void trapezy(int n, double xp, double xk) {
    double dx = (xk - xp) / n;;


    double temp = 0;
    double wynik = 0;

    double a, b;

    for (int i = 0; i < n; i++) {
        a = pow((xp + temp), 3) + 2;
        b = pow((xp + temp + dx), 3) + 2;

        wynik += ((a + b) / 2) * dx;
        temp += dx;
    }

    cout << "Metoda 2 = " << wynik << endl;
}
```

Rysunek 2: Fragment kodu (implementacja metody trapezów)

Funkcja sumuje obliczone wartości odpowiednich punktów funkcji, następnie dzieli przez 2 po czym mnoży przez rozmiar przedziałów.

2.3 metoda Simpsona

Funkcja oblicza całkę oznaczoną metodą Simpsona. Identycznie jak w przypadku poprzednich metod, funkcja ta przyjmuje jako parametry rozmiar przedziałów, dla których całka ma zostać oszacowana jak i zakres, dla którego ma zostać policzona całka.

```
void simpson(int n, double xp, double xk) {
    double temp = 0;
    double wynik = 0;
    double h = (xk - xp) / n;;

    double a, b, c;

    for (int i = 0; i < n - 2; i++) {
        a = pow((xp + temp), 3) + 2;
        b = pow((xp + temp + h), 3) + 2;
        c = pow((xp + temp + 2 * h), 3) + 2;

        temp += 2 * h;
        wynik += ((a + (4 * b) + c) * (h / 3));

    }
    cout << "Metoda 3 = " << wynik << endl;

}
```

Rysunek 3: Fragment kodu (implementacja metody Simpsona)

Funkcja oblicza wartości funkcji w trzech punktach dla danego rozmiaru, a następnie oblicza całkę paraboli dla danego obszaru.

2.4 metoda Monte Carlo

Funkcja oblicza całkę oznaczoną metodą Monte Carlo. Identycznie jak w przypadku poprzednich metod, funkcja ta przyjmuje jako parametry rozmiar przedziałów, dla których całka ma zostać oszacowana jak i zakres, dla którego ma zostać policzona całka.

```
void monteCarlo(int n, double xp, double xk) {
    double temp = 0;
    double wynik = 0;

    double* fx = new double[n];
    double a;
    fx[0] = NULL;

    for (int i = 0; i < n; i++) {
        fx[i] = fRand(xp, xk);
        cout << "punkt nr " << i << " = " << fx[i] << "\n";
        a = pow((fx[i]), 3) + 2;
        wynik += a / n;
    }
    wynik = wynik * fabs(xk - xp);
    cout << "Metoda 4 = " << wynik << endl;

}
```

Rysunek 4: Fragment kodu (implementacja metody Monte Carlo)

Funkcja losuje n punktów za pomocą funkcji fRand, która przyjmuje zakres dla którego liczona jest całka.

```
double fRand(double fMin, double fMax) {
    double f = (double)rand() / RAND_MAX;
    return fMin + f * (fMax - fMin);
}
```

Rysunek 5: Fragment kodu (funkcja losująca liczby typu double)

Po wylosowaniu liczb, funkcja oblicza ich średnią a następnie mnoży ją przez wartość bezwzględną różnicy punktu końca zakresu z punktem początku badanego zakresu.

3. Testy kodu numerycznego

3.1 metoda Prostokątów

TEST I: $\int_{1}^{4} f(x^3+2)\,dx\,, dx=1$

```
dx = 1
Metoda 1 = 105
```

Rysunek 3.1.1: wynik pierwszego testu dla dx = 1

TEST II: $\int_{1}^{4} f(x^3+2)\,dx\,, dx=0.1$

```
dx = 0.1
Metoda 1 = 72.9375
```

Rysunek 3.1.2: wynik drugiego testu dla dx = 0.1

TEST III: $\int_{1}^{4} f(x^3+2)\,dx\,, dx=0.01$

```
dx = 0.01
Metoda 1 = 70.0654
```

Rysunek 3.1.3: wynik trzeciego testu dla dx = 0.01

3.2 metoda Trapezów

TEST I: $\int_{1}^{4} f(x^3+2)\,dx\,, dx=1$

```
dx = 1
Metoda 2 = 73.5
```

Rysunek 3.2.1: wynik pierwszego testu dla dx = 1

TEST II: $\int_1^4 f(x^3+2)\,dx\,,dx=0.1$

```
dx = 0.1
Metoda 2 = 69.7875
```

Rysunek 3.2.2: wynik drugiego testu dla dx = 0.1

TEST III: $\int_1^4 f(x^3+2)\,dx\,,dx=0.01$

```
dx = 0.01
Metoda 2 = 69.7504
```

Rysunek 3.2.3: wynik trzeciego testu dla dx = 0.01

3.3 metoda Simpsona

TEST I: $\int_1^5 f(x^3+2)\,dx\,,n=4$

```
n = 4
Metoda 3 = 164
```

Rysunek 3.3.1: wynik pierwszego testu dla n = 4

TEST II: $\int_1^5 f(x^3+2)\,dx\,,n=40$

```
n = 40
Metoda 3 = 164
```

Rysunek 3.3.2: wynik drugiego testu dla n = 40

TEST III: $\int_1^5 f(x^3+2)\,dx\,,n=400$

```
n = 400
Metoda 3 = 164
```

Rysunek 3.3.3: wynik trzeciego testu dla n=400

3.4 metoda Monte Carlo

TEST I: $\int_{1}^{5} f(x^3+2)\,dx\,, n=4$, dla 4 punktów z prezentacji z zajęć „lab5b.pdf”

```
n = 4
punkt nr 0 = 1.5
punkt nr 1 = 2.6
punkt nr 2 = 3.8
punkt nr 3 = 4.5
Metoda 4 = 174.948
```

Rysunek 3.4.1: wynik pierwszego testu dla n = 4

TEST II: $\int_{1}^{5} f(x^3+2)\,dx\,, n=40$

```
n = 40
Metoda 4 = 135.415
```

Rysunek 3.4.2: wynik drugiego testu dla n = 40

TEST III: $\int_{1}^{5} f(x^3+2)\,dx\,, n=400$

```
n = 400
Metoda 4 = 161.219
```

Rysunek 3.4.3: wynik trzeciego testu dla n = 400

4. Analiza wyników

Poniższe tabele zawierają zestawienie wyników z testów przeprowadzonych w 3 części oraz wyników z kalkulatora online.

4.1 Metoda prostokątów

Rozmiar przedziału	***Obliczona wartość***	***Wartość obliczona online***
1	105	105
0.1	72.9375	72.9375
0.01	70.0654	70.06537499999997

Widać, że wyniki są niemal identyczne.

4.2 Metoda trapezów

Rozmiar przedziału	***Obliczona wartość***	***Wartość obliczona online***
1	73.5	73.5
0.1	69.7875	69.78750000000001
0.01	69.7504	69.75037499999996

Również widać, że wyniki są niemal identyczne.

4.3 Metoda Simpsona

Liczba przedziałów	***Obliczona wartość***	***Wartość obliczona online***
4	164	164
40	164	164.00000000000003
400	164	164.00000000000006

Również widać, że wyniki są niemal identyczne.

4.4 Metoda Monte Carlo

Liczba losowań	***Obliczona wartość***	***Wartość obliczona online***
4	174.948	141.4180559719122
40	135.415	134.21788058826888
400	161.219	160.36826196648778

Wyniki się różnią ze względu iż metoda Monte Carlo bazuje na liczbach losowych, przez co przy każdym uruchomieniu programu zwracany wynik się trochę różni.

5. Wnioski

W metodzie prostokątów, obserwujemy znaczącą poprawę dokładności w miarę zmniejszania przedziałów i zwiększania liczby obliczeń. Metoda trapezów, z drugiej strony, oferuje wyższą precyzję nawet przy mniejszej liczbie przedziałów. Chociaż zwiększenie liczby przedziałów poprawia dokładność, efekt ten jest mniej wyraźny niż w przypadku metody prostokątów. Metoda Simpsona dostarczyła najbardziej precyzyjnych wyników. Metoda Monte Carlo, opierająca się na liczbach losowych, jako jedyna zwraca różne wyniki przy każdym uruchomieniu. Niemniej jednak, zwiększanie liczby losowanych punktów zwiększa prawdopodobieństwo uzyskania wyniku zbliżonego do prawdziwej wartości. Podsumowując podczas wszystkich testów, metoda Simpsona wykazała największą dokładność. Wynika to z faktu, że metoda Simpsona opiera się na interpolacji parabolicznej, a funkcja, której całkę chcieliśmy obliczyć, jest parabolą.

6. Źródła

- prezentacja z zajęć „lab 5a.pdf”
- prezentacja z zajęć „lab 5b.pdf”
- arkusz kalkulacyjny Microsoft Excel
- kalkulator internetowy www.algorytm.org/procedury-numeryczne/