Sprawozdanie
Metody Numeryczne 2, laboratorium 3

Grzegorz Rozdzialik (D4, grupa lab. 2)

8 listopada 2016

1 Zadanie

Temat **3**, zadanie **33**:

Obliczanie całek

$$\iint_D f(x,y)\,dx\,dy$$

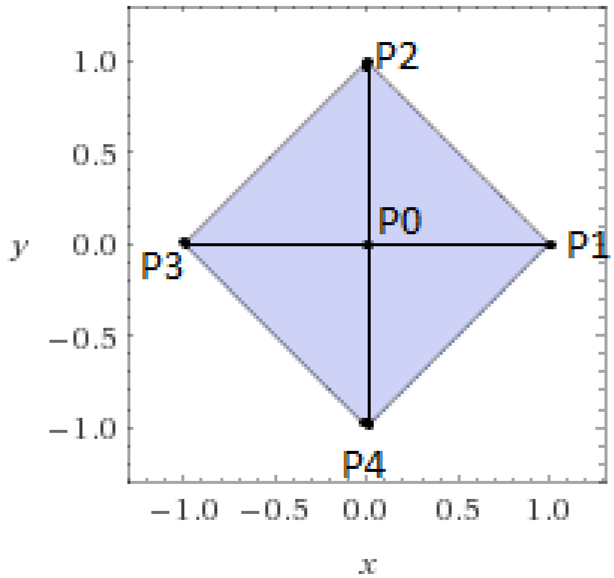
na obszarze

$$D = \{(x,y) \in \mathbb{R}^2 : |x| + |y| \leqslant 1\}$$

przez podział D na $4n^2$ trójkątów przystających, i zastosowanie na każdym z nich kwadratury rzędu 4-go.

Obszar D został przedstawiony na rysunku nr 1, jest to romb o środku $P_0 = (0,0)$ i wierzchołkach

$$P_1 = (1,0)$$
$$P_2 = (0,1)$$
$$P_3 = (-1,0)$$
$$P_4 = (0,-1)$$

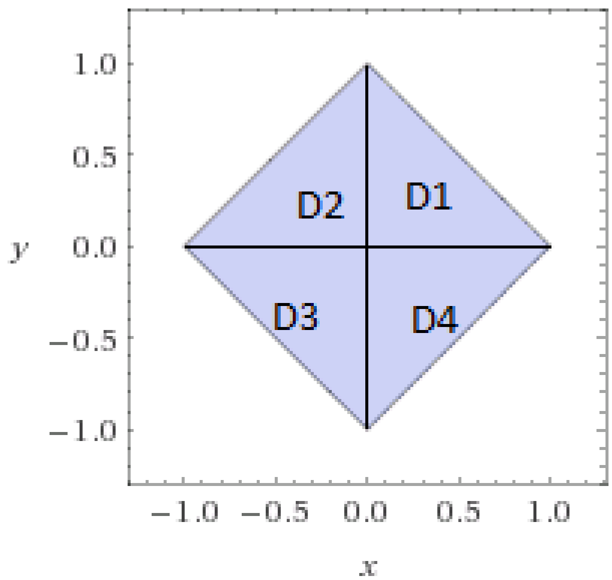


Rysunek 1: Obszar D

2 Opis metody

2.1 Podział rombu na $4n^2$ trójkątów

W celu podzielenia obszaru D na $4n^2$ trójkątów użyto podział na 4 ćwiartki D_1, D_2, D_3, D_4 na podstawie osi układu współrzędnych. Podział ten został przedstawiony na rysunku nr 2.

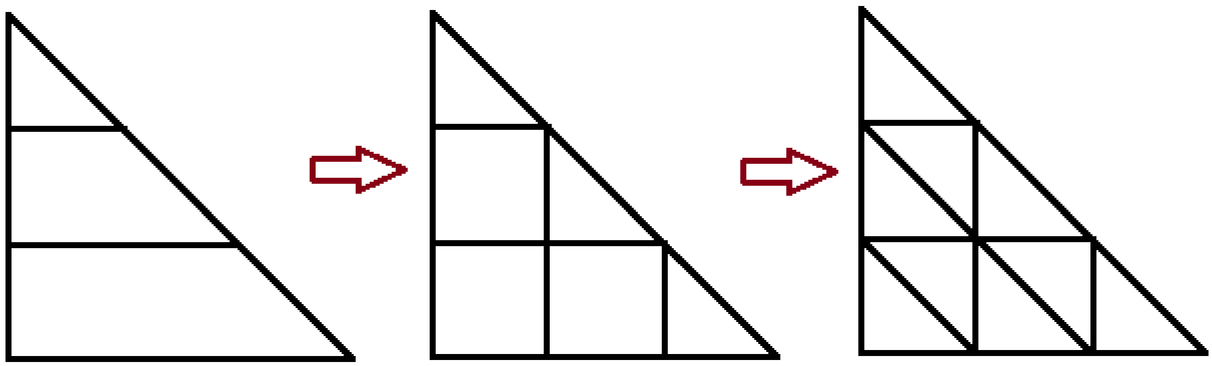


Rysunek 2: Podział D na ćwiartki

Następnie każdą z ćwiartek podzielono na n^2 trójkątów według następującej reguły:

1. Każdy bok podzielono na n równych części (punkty podziału nazwijmy węzłami).
2. Węzły leżące na dwóch różnych bokach trójkąta i równoodległe od trzeciego z boków łączymy prostymi. Proste te będą wtedy równoległe do jednego z boków.

Przykładowy schemat podziału ćwiartki D_1 z $n = 3$ (D_1 podzielono na 9 trójkątów przystających) został umieszczony na rysunku nr 3.



Rysunek 3: Podział D_1 na 9 ($= 3^2$) trójkątów

Wszystkie n^2 trójkątów po podziale mają takie samo pole, równe $p = \frac{P}{n^2}$, gdzie $P = |D_1| = |D_2| = |D_3| = |D_4|$.

Na każdym z tych trójkątów obliczamy wartość kwadratury 4-go rzędu:

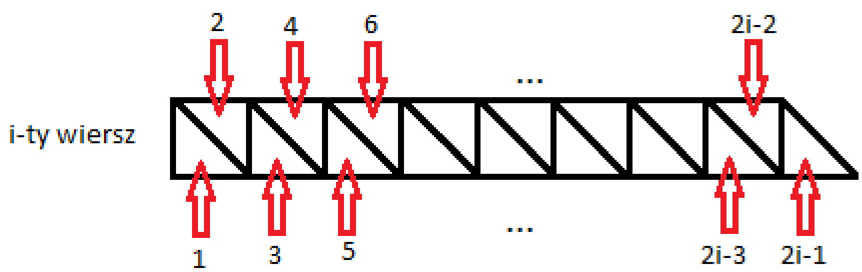
$$
\begin{aligned}
S(f) = & \frac{p}{60}\Bigg[27f(P_{012}) + 3\Big(f(P_{01}) + f(P_{02}) + f(P_{12})\Big) \\
& + 8\Big(f(P_0) + f(P_1) + f(P_2)\Big)\Bigg]
\end{aligned} \tag{1}
$$

gdzie f jest zadaną funkcją podcałkową, p zdefiniowane jak poprzednio, P_0, P_1, P_2 są wierzchołkami trójkąta po podziale, $P_{i,j} = \frac{P_i+P_j}{2}$ są środkami boków trójkąta, $P_{012} = \frac{P_0+P_1+P_2}{3}$ jest środkiem ciężkości trójkąta.

2.2 Wyznaczanie współrzędnych wierzchołków trójkątów po podziale

Aby zastosować kwadraturę (1) należy znać współrzędne wierzchołków. Zauważmy, że trójkąty po podziale można pogrupować w wiersze tak, że w pierwszym wierszu znajduje się jeden trójkąt, a w każdym kolejnym są o 2 więcej. Ostatni wiersz (n-ty) posiada $2n - 1$ trójkątów.

W i-tym wierszu jest $2i - 1$ trójkątów. Podział i-tego wiersza wraz z numerami kolejnych trójkątów został pokazany na rysunku nr 4.



Rysunek 4: Podział i-tego wiersza

Niech P_0, P_1, P_2 będą współrzędnymi wierzchołków trójkąta przed podziałem zdefiniowanym analogicznie jak na rysunku 1. Podział na wiersze rozpoczynamy od P_2, zatem trójkąt po podziale, którego jednym z wierzchołków będzie P_2 znajdzie się w pierwszym wierszu.

Zdefiniujmy następujące zmienne:

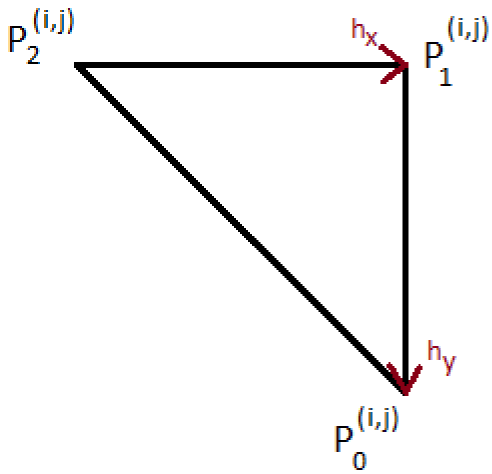
$$h_x = \frac{P_1 - P_0}{n}$$
$$h_y = \frac{P_0 - P_2}{n}$$

Wtedy h_x będzie wektorem, o jaki należy się przesunąć, aby uzyskać współrzędne kolejnej kolumny, natomiast h_y będzie wektorem, o jaki należy się przesunąć, aby uzyskać współrzędne kolejnego wiersza. Zakładam, że każda kolumna oprócz ostatniej w danym wierszu posiada 2 trójkąty.

Zatem jeżeli trójkąt ma nieparzysty indeks, to znajduje się bliżej kolejnego wiersza (”na dole”), a te z indeksem parzystym są bliżej poprzedniego wiersza (”na górze”), jak na rysunku 4.

Aby wyznaczyć współrzędne trójkąta o indeksach (i, j), gdzie $i \in \{1, 2, \ldots, n\}$, $j \in \{1, 2, \ldots, 2i-1\}$, w oparciu o współrzędne wierzchołka P_2 należy rozważyć dwa przypadki:

I $2 \mid j$

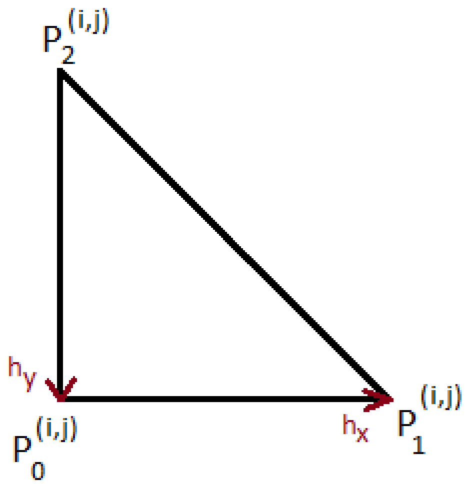


Rysunek 5: Trójkąt po podziale, gdy j - parzyste

Wtedy

$$P_2^{(i,j)} = P_2 + (i-1)h_y + \frac{j-2}{2}h_x$$
$$P_1^{(i,j)} = P_2^{(i,j)} + h_x$$
$$P_0^{(i,j)} = P_1^{(i,j)} + h_y$$

II $2 \nmid j$



Rysunek 6: Trójkąt po podziale, gdy j - nieparzyste

Wtedy

$$
\begin{aligned}
P_2^{(i,j)} &= P_2 + (i-1)h_y + \frac{j-1}{2}h_x \\
P_0^{(i,j)} &= P_2^{(i,j)} + h_y \\
P_1^{(i,j)} &= P_0^{(i,j)} + h_x
\end{aligned}
$$

gdzie $P_k^{(i,j)}$ oznacza współrzędne k-tego wierzchołka trójkąta o indeksach (i,j) po podziale (analogicznie do rysunków w podpunktach).

3 Implementacja metody

Metoda zaimplementowana jest na podstawie czterech funkcji oraz jednego skryptu pozwalającego na łatwe jej wykorzystanie i porównanie z funkcją *integral2* z MATLABa:

- $[h_x, h_y, P] = computeDivisionProperties(P_0, P_1, P_2, n)$

  Funkcja oblicza własności podziału trójkąta o wierzchołkach P_0, P_1, P_2 na n^2 trójkątów przystających (zgodnie z powyższym opisem metody). Zwraca wektory h_x, h_y oraz pole trójkąta P po podziale.

- $[P_0^{(i,j)}, P_1^{(i,j)}, P_2(i,j)] = computeSingleTriangleCoordinates(P_2, h_x, h_y, i, j)$

  Funkcja oblicza współrzędne trójkąta o indeksie (i, j) po podziale na podstawie współrzędnej P_2 trójkąta przed podziałem oraz wektorów h_x, h_j.

- $S = integrateSingleTriangle(f, P_0, P_1, P_2, P)$

  Funkcja przybliża wartość całki z funkcji podcałkowej f na trójkącie o polu P i wierzchołkach P_0, P_1, P_2, używając do tego kwadratury (1).

- $S = numericalInterpolationTriangle(f, P_0, P_1, P_2, n)$

  Funkcja przybliża wartość całki z funkcji podcałkowej f na trójkącie o wierzchołkach P_0, P_1, P_2, dzieląc go na n^2 trójkątów przystających.

  Funkcja używa $computeDivisionProperties$ do uzyskania własności podziału (h_x, h_y, P), a następnie wykonuje przybliżenie całki na każdym z trójkątów po podziale używając funkcji $computeSingleTriangleCoordinates$ do uzyskania współrzędnych tego trójkąta, oraz $integrateSingleTriangle$ do uzyskania wartości kwadratury.

- $integrateDiamond$

  Skrypt pozwalający określić funkcję podcałkową f oraz parametr n określający liczbę podziałów. Wykonuje zadanie - przybliża całkę na obszarze D (rysunek 1).

  Skrypt dodatkowo podaje informacje o szybkości działania metody oraz porównuje ją z funkcją $integral2$ dostępną w MATLABie.

Funkcji można używać do przybliżania całki na dowolnym trójkącie, zatem nie musi być to romb podzielony na 4 trójkąty. Wystarczy zmodyfikować skrypt $integrateDiamond$.

Nie jest to jednak tematem zadania, więc postanowiono zostawić to jako zadanie dla ciekawego czytelnika.

4 Poprawność metody

Kwadratura (1) jest rzędu 4, zatem dla wielomianów stopnia co najwyżej 3 metoda daje poprawne wyniki, nawet z $n = 1$. Dla wielomianów stopnia wyższego przybliżenie nie jest dokładne, ale dokładność wzrasta wraz ze wzrostem parametru n.

5 Przykłady

W przykładach najpierw została obliczona prawdziwa wartość całki, a następnie porównane w tabeli wyniki metody z zadania oraz funkcji *integral2*.

Przykład 1 Całka z wielomianu stopnia 3.

$$\iint_D f(x,y)\,dx\,dy = \iint_D x^3\,dx\,dy = 0$$

$n = 1$

	metoda z zadania	funkcja *integral2*
przybliżenie całki	0	1.47451×10^{-17}
czas obliczania	2.6 ms	4.13 ms

Metoda z zadania jest dokładna (i powinna być, gdyż rząd kwadratury jest równy 4, a funkcja podcałkowa jest wielomianem stopnia 3), a także szybsza.

Przykład 2 Całka z wielomianu stopnia 4.

$$\iint_D f(x,y)\,dx\,dy = \iint_D x^4\,dx\,dy = \frac{2}{15} \approx 0.133333$$

$n = 1$

	metoda z zadania	funkcja *integral2*
przybliżenie całki	0.144444	0.133333
czas obliczania	6.19 ms	4.7 ms

W tym przypadku metoda z zadania okazała się gorsza od funkcji dostępnej w MATLABie, zarówno pod względem dokładności, jak i szybkości działania.

Przykład 3 Zwiększenie parametru n w stosunku do poprzedniego przykładu.

$$\iint_D f(x,y)\,dx\,dy = \iint_D x^4\,dx\,dy = \frac{2}{15} \approx 0.133333$$

$n = 10$

	metoda z zadania	funkcja *integral2*
przybliżenie całki	0.133334	0.133333
czas obliczania	16.34 ms	6.06 ms

Przy zwiększeniu ilości podziałów metoda osiąga lepsza dokładność, jednakże zmniejsza się jej szybkość. W tym przypadku nadal jest gorsza od funkcji *integral2*.

Podobną dokładność osiąga dopiero dla $n = 17$, jednakże wtedy jej czas działania około dziesięciokrotnie większy od MATLABowej alternatywy.

Przykład 4 Funkcja podcałkowa nie będąca wielomianem.

$$\iint_D f(x,y)\,dx\,dy = \iint_D (\sin x + \cos y)\,dx\,dy = 4 - 4\cos 1 \approx 1.83879$$

$n = 1$

	metoda z zadania	funkcja *integral2*
przybliżenie całki	1.8392	1.83879
czas obliczania	3.36 ms	2.9 ms

Po raz kolejny funkcja dostępna w MATLABie jest szybsza i dokładniejsza.

Dla $n = 4$ metoda z zadania osiąga podobną dokładność.

	metoda z zadania	funkcja *integral2*
przybliżenie całki	1.83879	1.83879
czas obliczania	4.66 ms	3.13 ms

Przykład 5 Wielomian stopnia 7.

$$\iint_D f(x,y)\,dx\,dy = \iint_D (x^7 + x^6 + y^4 + y^3)\,dx\,dy = \frac{43}{210} \approx 0.204762$$

$n = 1$

	metoda z zadania	funkcja *integral2*
przybliżenie całki	0.254012	0.204762
czas obliczania	3.59 ms	6.96 ms

W tym przypadku metoda z zadania okazała się być szybsza, ale nie dokładniejsza. Podobną dokładność osiąga dla $n = 16$:

	metoda z zadania	funkcja *integral2*
przybliżenie całki	0.204762	0.204762
czas obliczania	40.65 ms	7.05 ms

6 Wnioski

1. Czas działania wzrasta kwadratowo wraz ze wzrostem parametru n.
2. Metoda z zadania osiąga gorsze wyniki od funkcji *integral2* dostępnej w MATLABie. Jest w stanie osiągnąć podobną dokładność, jednak kosztem znacznego zwiększenia się czasu potrzebnego na obliczenia.
3. Jeżeli nie zależy nam na dokładności, można użyć metody z zadania z $n = 1$, wtedy jest bardzo prawdopodobnym, że będzie ona szybsza od funkcji *integral2*.

7 Skrypt do testowania

Skrypt *integrateDiamond* pozwala na ustalanie funkcji podcałkowej f oraz parametru n określającego ilość podziałów obszaru D. Wyświetla on także czas przybliżania całki tą metodą oraz porównuje wyniki z funkcją *integral2* dostępną w MATLABie.

Skrypt zgodnie z opisem metody dzieli obszar D na 4 trójkąty przystające, a następnie każdy z nich na n^2 trójkątów przystających i na nich stosuje kwadraturę.

Pokazuje on także jak używać funkcji stworzonych na potrzeby tej metody do przybliżania całki na dowolnym trójkącie.

```
% Parametry:
% Funkcja podcalkowa
f = @(x, y)(x.^2 + y.^2);
% Liczba okreslajaca ilosc podzialow
n = 1;

% Srodek rombu
P0 = [0 0];
% Wierzcholki rombu
P1 = [1 0];
P2 = [0 1];
P3 = [-1 0];
P4 = [0 -1];

% Kwadratura dla trojkatow w poszczegolnych cwiartkach ukladu wspolrzednych
S1 = numericalInterpolationTriangle(f, P0, P1, P2, n); % I cwiartka
S2 = numericalInterpolationTriangle(f, P0, P2, P3, n); % II cwiartka
S3 = numericalInterpolationTriangle(f, P0, P3, P4, n); % III cwiartka
S4 = numericalInterpolationTriangle(f, P0, P4, P1, n); % IV cwiartka

S = S1 + S2 + S3 + S4;
```

8 Bibliografia

1. Informacje z wykładu *Metod numerycznych 2* (wydział MiNI PW, dr Iwona Wróbel) (w szczególności wzór na kwadraturę rzędu 4 na trójkącie)