27 listopada 2019

Michał Gozdera
grupa: G1
nr indeksu: 298869

24. Interpolacja funkcjami liniowymi na prostokącie podzielonym na 2mn trójkątów przystających. Tablicowanie funkcji, przybliżenia i błędu w środkach ciężkości trójkątów. Obliczenie błędu średniokwadratowego w tych punktach.

Projekt nr 2

1 Wstęp

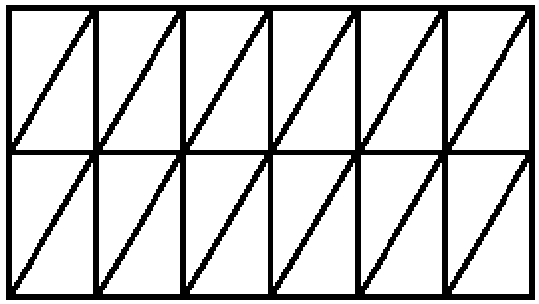
Celem projektu jest interpolacja funkcji dwóch zmiennych funkcjami liniowymi na prostokącie. Prostokąt dzielimy na trójkąty przystające i przeprowadzamy interpolację na każdym z nich. Otrzymane wyniki tablicujemy uwzględniając błąd względny oraz obliczamy błąd średniokwadratowy.

Stosując metodę interpolacji na trójkątach przyjmujemy, że dla każdego trójkąta węzły interpolacji znajdują się w jego wierzchołkach. Otrzymujemy zatem 3 węzły, co pozwala na wyznaczenie liniowej funkcji interpolującej. Mając funkcje interpolujące na każdym z trójkątów, mamy również funkcję interpolującą na zadanym prostokącie.

Zastosowanie powyższej metody do interpolacji większości funkcji dwóch zmiennych pozwala na uzyskanie dokładnego wyniku. Należy jednak pamiętać o odpowiednim dobraniu liczby trójkątów względem rodzaju funkcji oraz długości przedziałów określających obszar, na którym interpolujemy (długości boków prostokąta).

2 Opis metody

Interpolację należy zacząć od określenia obszaru, na którym interpolujemy. Niech zatem prostokąt określony będzie przez odcinki $[x\_beg, x\_end]$ oraz $[y\_beg, y\_end]$. Podzielmy najpierw prostokąt na mn mniejszych prostokątów, a następnie każdy z nich na dwa trójkąty przystające. Przyjmijmy, że dzielimy odcinek równoległy do osi OX na n części, a do osi OY - na m. Otrzymaliśmy w ten sposób podział wejściowego obszaru na $2mn$ trójkątów przystających.



Rysunek 1: Podział prostokąta na trójkąty przystające.

Aby obliczyć wartość funkcji interpolującej w punkcie o współrzędnych (xx, yy), lokalizujemy, w którym z trójkątów znajduje się ten punkt, a następnie przeprowadzamy interpolację na tym trójkącie. Najpierw określamy numer kolumny i wiersza "małego" prostokąta, w którym znajduje się punkt:

$$col = \lceil (xx - x\_beg)/hx) \rceil$$
$$row = \lceil (yy - y\_beg)/hy) \rceil$$

gdzie:

$$hx = (x\_end - x\_beg)/n$$
$$hy = (y\_end - y\_beg)/m.$$

Następnie sprawdzamy, czy szukanym trójkątem jest "dolny" czy "górny" trójkąt prostokąta znajdującego się w kolumnie nr col i wierszu nr row. Sprowadza się to do określenia czy punkt znajduje się powyżej czy poniżej przekątnej prostokąta, która ma równanie:

$$yy = (hy/hx) * (xx) + (y\_beg + row * hy) - (hy/hx) * (x\_beg + col * hx).$$

Kolejnym krokiem jest znalezienie wartości funkcji interpolującej na określonym powyżej trójkącie w punkcie (xx, yy). Korzystamy tutaj z poniższych wzorów:

$$d = (x1 - x2) * (y0 - y2) - (y1 - y2) * (x0 - x2)$$

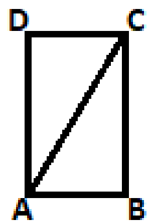
$$fi_0(x, y) = ((y2 - y1) * x + (x1 - x2) * y + x2 * y1 - y2 * x1)/d$$
$$fi_1(x, y) = ((y0 - y2) * x + (x2 - x0) * y + x0 * y2 - y0 * x2)/d$$

$$fi_2(x,y) = ((y1 - y0) * x + (x0 - x1) * y + x1 * y0 - y1 * x0)/d$$

$$v = fun(x0,y0) * fi_0(xx,yy) + fun(x1,y1) * fi_1(xx,yy) + fun(x2,y2) * fi_2(xx,yy)$$

gdzie: $(x0,y0)$, $(x1,y1)$, $(x2,y2)$ - węzły interpolacji - wierzchołki trójkąta, v - wartość funkcji interpolującej, fun - interpolowana funkcja (projekt dotyczy badania właściwości metody interpolacji, dlatego zakładamy, że znamy jawny wzór funkcji interpolowanej. W rzeczywistych zastosowaniach korzystalibyśmy oczywiście z punktowych wartości funkcji w węzłach).

Następnym etapem projektu jest wyznaczenie środków ciężkości trójkątów i zbadanie błędu interpolacji w tych punktach. Rozważmy trójkąty jak na rysunku poniżej:



Rysunek 2: Trójkąty o wierzchołkach w punktach A, B, C i D.

Korzystamy z faktu, że współrzędna x-owa środka ciężkości trójkąta ABC "jest oddalona" od współrzędnej x-owej punktu A o $\frac{2}{3}hx$, a współrzędna y-owa o $\frac{1}{3}hy$ od współrzędnej y-owej punktu A. Analogicznie, dla trójkąta ACD odległości współrzędnych środka ciężkości od punktu A to odpowiednio: $\frac{1}{3}hx$, $\frac{1}{3}hy$. W ten sposób łatwo wyznaczyć współrzędne środków ciężkości wszystkich trójkątów, dodając odpowiednie przesunięcie wzdłuż osi OX i OY.

Tablicowanie przeprowadzamy poprzez wpisanie w kolejne kolumny macierzy odpowiednio wektorów: współrzędnej x-owej środka ciężkości, współrzędnej y-owej środka ciężkości, wartości funkcji interpolowanej w tym punkcie, wartości funkcji interpolującej oraz modułu różnicy między wartością dokładną a przybliżoną podzielonego przez moduł wartości dokładnej - błędu względnego.

Ostatnim etapem jest obliczenie błędu średniokwadratowego (MSE) według wzoru:

$$MSE = \frac{\sum_{i=1}^{2mn}(w(i) - g(i))^2}{2mn}$$

gdzie: w - wektor wartości dokładnych, g - wektor wartości przybliżonych.

3 Opis programu obliczeniowego

Program składa się z pięciu funkcji (argumenty wejściowe zgodnie z wcześniej zdefinowaymi oznaczeniami):

- `[v] = interp_fun(xx, yy, x_beg, y_beg, hx, hy, fun)` - zwraca wartość funkcji interpolującej w danym punkcie

- `[x_cg, y_cg] = center_of_gravity(x_beg, hx, x_end, y_beg, hy, y_end, n, m)` - zwraca dwa wektory - współrzędnych x-owych i y-owych środków ciężkości trójkątów
- `[ret] = func(x,y)` - dowolna funkcja dwóch zmiennych, używana podczas testowania metody
- `[mse] = mean_squared_error(w, g)` - zwraca błąd średniokwadratowy
- `[TAB, MSE] = main(x_beg, x_end, y_beg, y_end, fun, n, m)` - zwraca macierz używaną w tablicowaniu (`TAB`) oraz wartość błędu średniokwadratowego (`MSE`)

Oprócz tego, skrypt `plot_script` służy do rysowania wykresów błędów.

4 Opis eksperymentów

Metodę testujemy dla różnych funkcji dwóch zmiennych, porównując błąd średniokwadratowy. W kolejnych wywołaniach programu zmieniamy liczby m i n oraz obszar, na którym interpolujemy. Uzyskane wyniki błędu średniokwadratowgo przedstawiamy na wykresach. Poszukujemy takich wartości m, n dla których błąd jest pomijalnie mały.

5 Przykłady obliczeniowe

Poniżej przedstawiam kilka przykładowych wywołań funkcji `main` wraz z fragmentami macierzy `TAB`:

1. `main(1, 10, 1, 10, @func,` n, m`)`
   $func(x, y) = x^2 + 2y^2 - y - 3x + 500sin(x)$

Tabela 1: Wartości przybliżone i dokładne w 3 wybranych punktach dla różnych n i m.

n	m	wartość dokładna	wartość przybliżona	błąd
10	10	405.6950	384.3992	0.0525
10	10	697.3998	654.2473	0.0619
10	10	624.4736	592.5299	0.0512
100	100	131.9448	132.0638	0.0009
100	100	90.3200	90.4776	0.0017
100	100	50.0428	50.2379	0.0039
10	200	416.6304	394.9755	0.0520
10	200	708.3353	664.8236	0.0614
10	200	635.4090	603.1063	0.0508

2. `main(1, 10, 1, 10, @func,` n, m`)`

   $func(x, y) = x + y$

Tabela 2: Wartości przybliżone i dokładne w 3 wybranych punktach dla różnych n i m.

n	m	wartość dokładna	wartość przybliżona	błąd
2	2	6.5	6.5	0
2	2	11	11	0
2	2	15.5	15.4999	0
10	200	16.6850	16.6850	$1.41e-13$
10	200	17.5850	17.5850	$1.42e-13$
10	200	18.4850	18.4850	$5.65e-13$

3. `main(1, 2, 1, 2, @func,` n, m`)`

   $func(x, y) = cos(x + y)$

Tabela 3: Wartości przybliżone i dokładne w 3 wybranych punktach dla różnych n i m.

n	m	wartość dokładna	wartość przybliżona	błąd
2	2	-0.8011	-0.7358	0.0816
2	2	-0.9900	-0.9092	0.0816
2	2	-0.8011	-0.6704	0.1632
80	20	-0.6996	-0.6992	0.0006
80	20	-0.6907	-0.6903	0.0006
80	20	-0.6816	-0.6812	0.0006

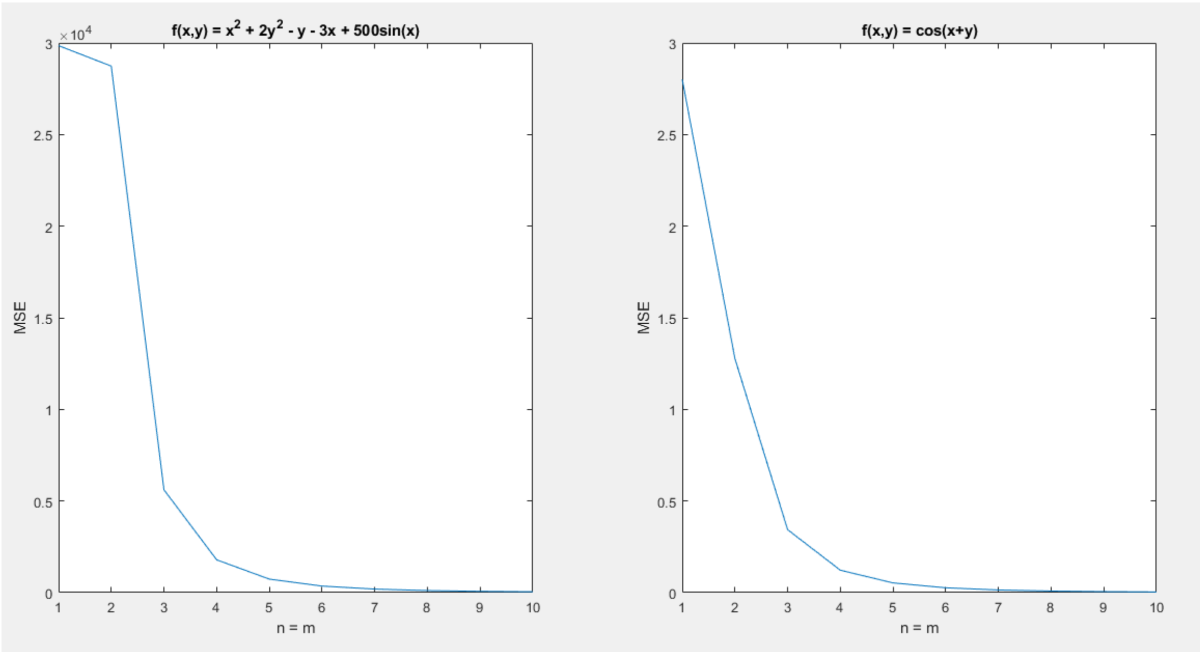
6 Analiza wyników

Za pomocą skryptu `plot_script` generujemy wykresy błędu średniokwadratowego dla różnych funkcji interpolowanych na obszarze $[1, 5] \times [1, 5]$ w zależności od m i n (na potrzeby analizy i sporządzenia wykresu przyjmujemy że n = m). Na zamieszczonych poniżej wykresach widać, że dla różnych funkcji błąd jest różnego rzędu.

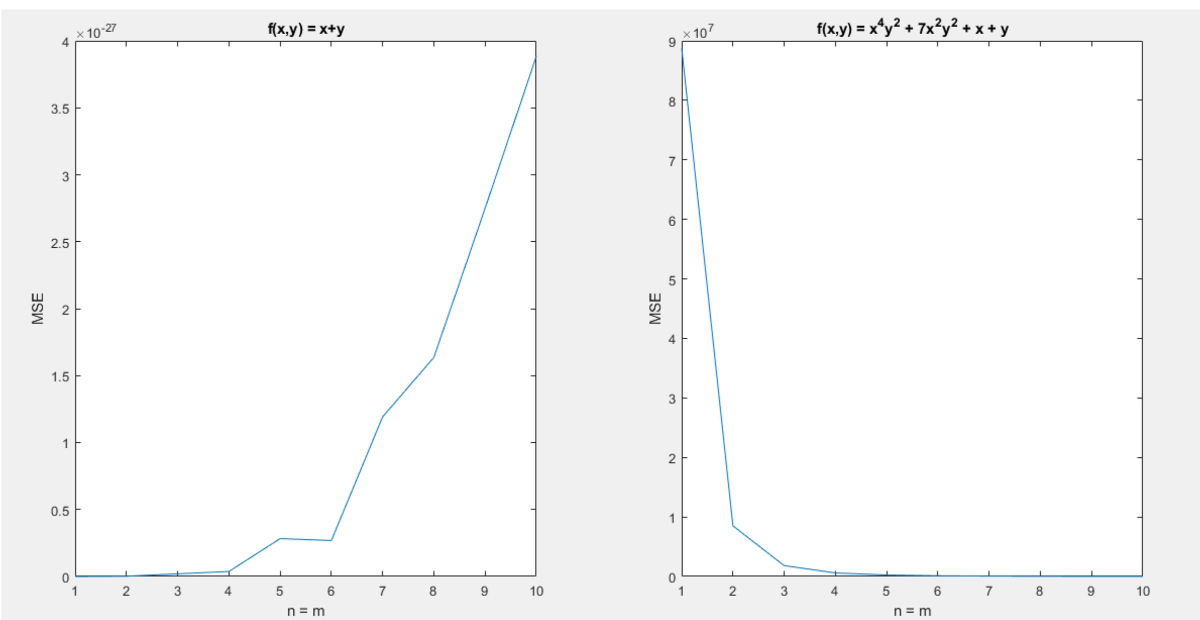
Co ciekawe, dla funkcji liniowej błąd rośnie. Jeśli jednak przyjrzymy się dokładnie wykresowi, zauważymy, że jest bardzo mały - rzędu $1e-27$. Jest to oczywista konsekwencja utraty dokładności spowodowanej dużą ilością obliczeń - jeśli podzielimy prostokąt na więcej trójkątów, musimy wykonać więcej obliczeń, a więc też i zaokrągleń. Na innych wykresach nie widać wzrostu błędu, ponieważ jest on bardzo mały względem "realnego" błędu metody.

Niezależnie od rzędu wielkości, błąd metody zbiega do zera wraz ze wzrostem n i m. Z wykresu odczytujemy, że dla $n = m = 10$ jest on już pomijalnie mały we wszystkich przypadkach. Zatem aby dla większości funkcji określonych na obszarze

$[1,5] \times [1,5]$ uzyskać bardzo dokładny wynik interpolacji wystarczy interpolować na $2 * m * n = 200$ trójkątach. W przypadku większych obszarów powinniśmy oczywiście interpolować na większej liczbie trójkątów.

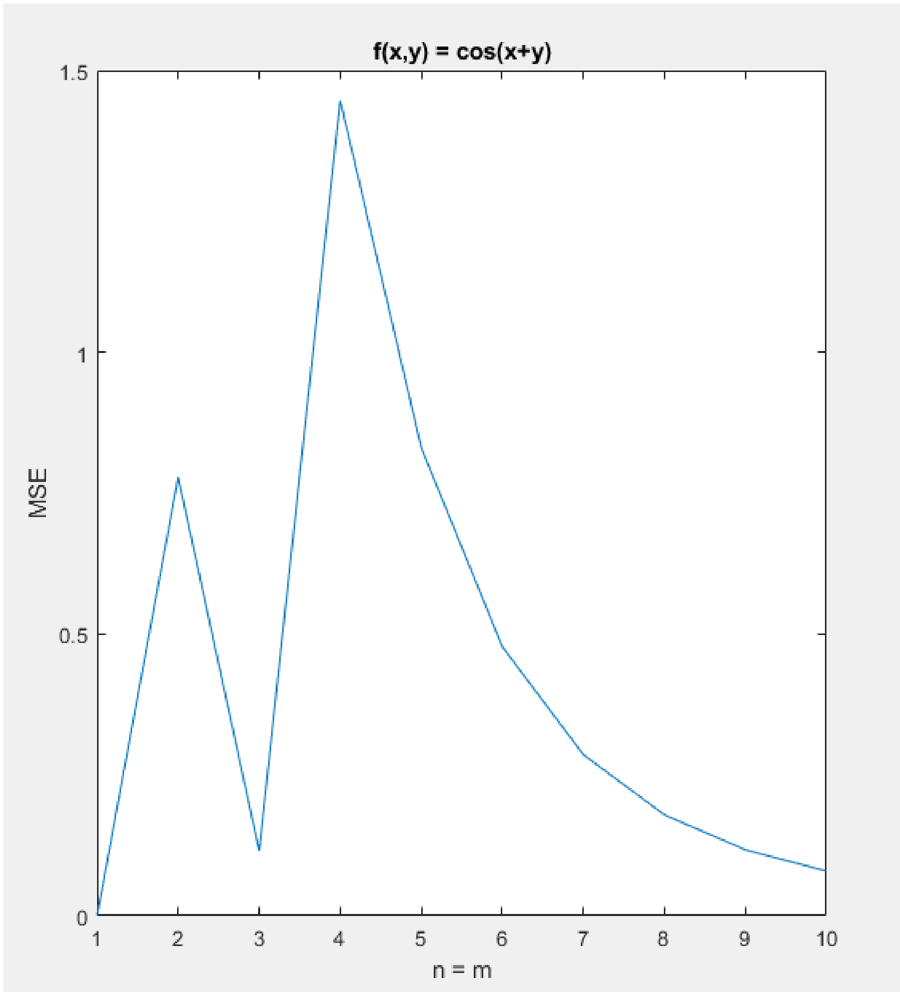


Rysunek 3: Wykresy błędów sumy wielomianu nieliniowego i funkcji sinus oraz funkcji okresowej cosinus.



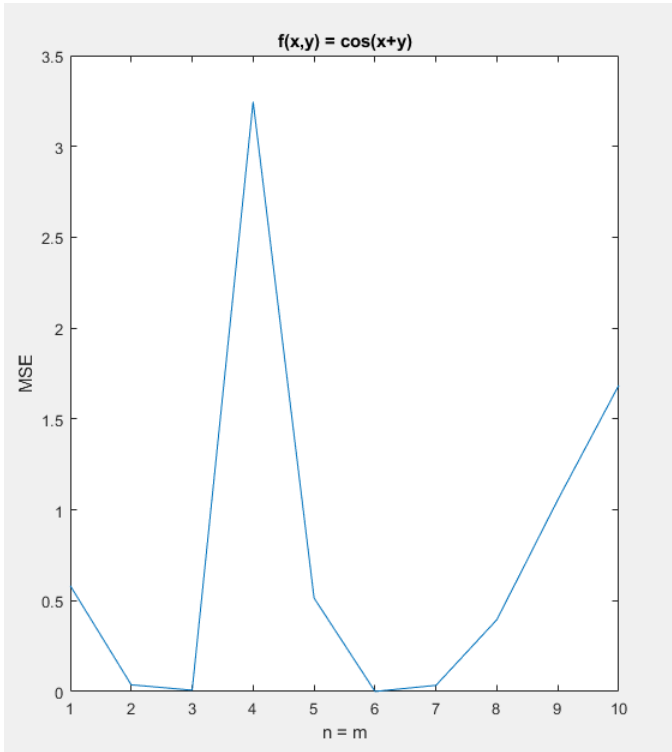
Rysunek 4: Wykresy błędów funkcji liniowej oraz wielomianu, w którym występują iloczyny argumentów.

Interesującą zależność można zaobserwować dla funkcji $f(x, y) = cos(x+y)$. Przeanalizujmy wykres MSE dla obszaru $[1, 10] \times [1, 10]$ i $n = m = 10$:

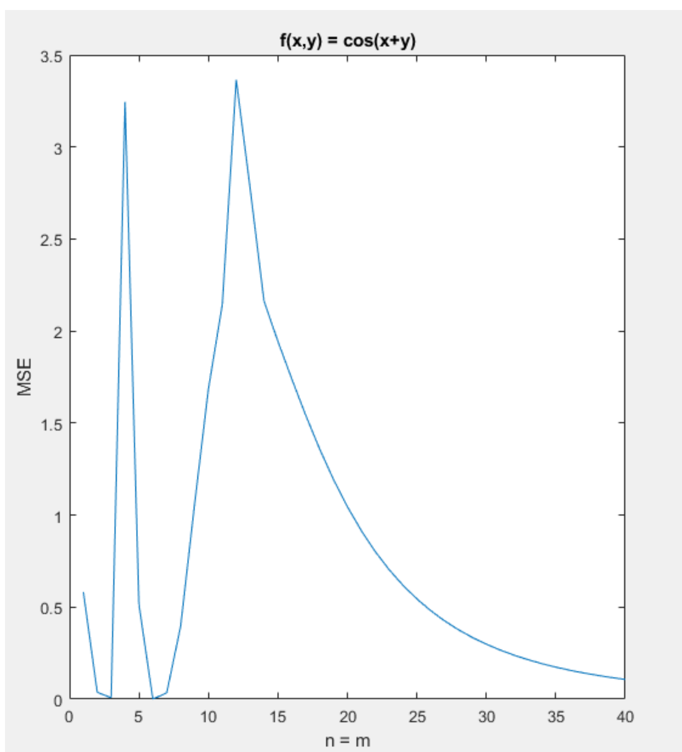


Rysunek 5: Wykres błędu dla obszaru $[1, 10] \times [1, 10]$ i $n = m = 10$.

Widać tutaj, że dla $n = m \leqslant 4$ nie można jednoznacznie określić czy błąd rośnie, czy maleje, ale dla $n = m \geqslant 5$ obserwujemy już regularny spadek wartości błędu. Zbadajmy zatem błąd interpolacji tej samej funkcji na większym obszarze: $[1, 40] \times [1, 40]$ i takim samym $n = m = 10$ (Rysunek 6). Ponownie widoczna jest nieregularność błędu, dodatkowo dla $n \geqslant 6$ błąd zaczyna rosnąć. Wnioskiem tego, podział tego obszaru na $2 * n * m = 200$ trójkątów dla funkcji $f(x, y) = cos(x + y)$ nie jest wystarczający. Na wykresie obserwujemy wartości $m = n$, dla których błąd interpolacji jest bardzo mały (np. 3, 6), ale bez sporządzenia wykresu trudno jest takie wartości określić. Zaobserwujmy zatem jak błąd będzie zmieniał się wraz ze wzrostem liczby $n = m$ do 40 i tego samego obszaru: $[1, 40] \times [1, 40]$ (Rysunek 7).

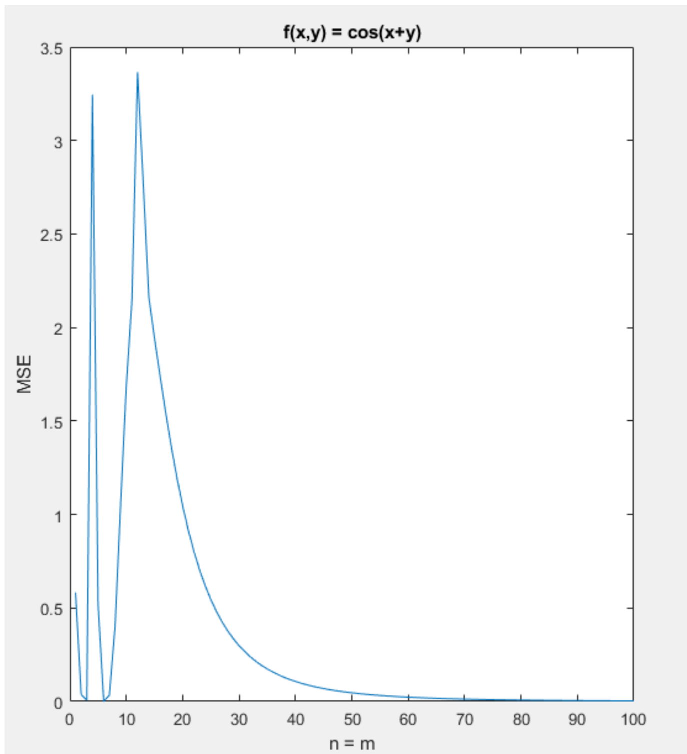


Rysunek 6: Wykres błędu dla obszaru $[1, 40] \times [1, 40]$ i $n = m = 10$.



Rysunek 7: Wykres błędu dla obszaru $[1, 40] \times [1, 40]$ i $n = m = 40$.

Wykres z Rysunku 7 przypomina ten z Rysunku 5. Mianowicie, od pewnego momentu (tutaj $n \geqslant 15$) funkcja błędu jest ściśle malejąca. Pozwala to wysnuć wniosek, że aby interpolacja funkcji okresowej była dokładna, należy odpowiednio dobrać liczby, na jakie dzielimy obszar do wielkości tego obszaru. W przypadku funkcji $f(x, y) = cos(x + y)$ liczby n i m powinny być zbliżone do długości przedziałów określających prostokąt. Poniżej przedstawiam dwa dodatkowe wykresy popierające prawdziwość tego wniosku.

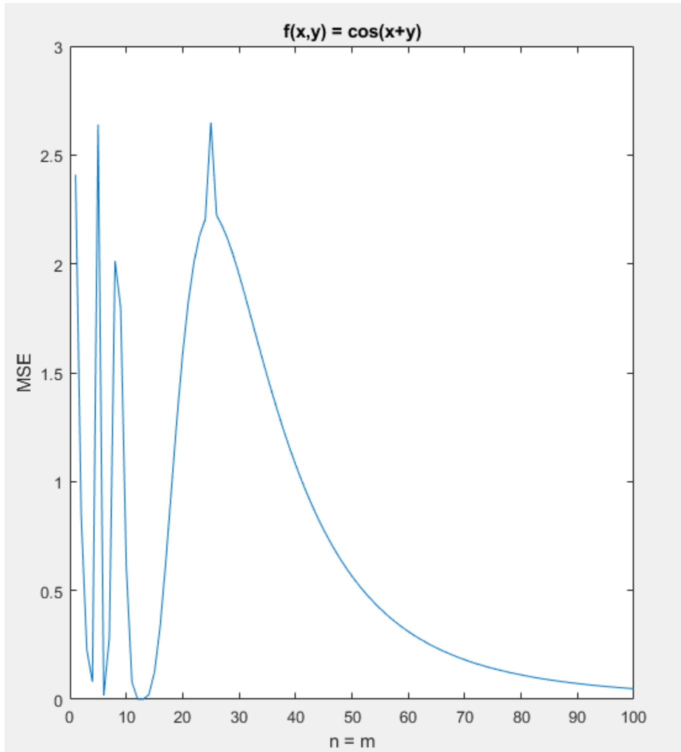


Rysunek 8: Wykres błędu dla obszaru $[1, 40] \times [1, 40]$ i $n = m = 100$.

Na Rysunku 8 widzimy, że dla $n = m \geqslant 40$ błąd jest pomijalnie mały i ściśle malejący. Oznacza to, że w tym przypadku wystarczy interpolować dla liczb $n = m = 40$, czyli zbliżonych do szerokości przedziału - dalsze zwiększanie tych liczb daje znikomą poprawę dokładności, przy i tak bardzo małym błędzie.

Z kolei na Rysunku 9 (poniżej) przestawiony jest błąd interpolacji dla obszaru: $[1, 80] \times [1, 80]$. Również tutaj łatwo zauważyć, że bezpieczną wartością m i n jest 80, czyli liczba zbliżona do długości przedziałów określających obszar interpolacji.

Podsumowując, metoda interpolacji funkcjami liniowymi na prostokątnym obszarze podzielonym na trójkąty jest dokładna przy odpowiednim doborze liczby trójkątów (a tym samym ich wielkości) w zależności od rodzaju funkcji interpolowanej.



Rysunek 9: Wykres błędu dla obszaru $[1, 80] \times [1, 80]$ i $n = m = 100$.