Zadanie numeryczne 1 - sprawozdanie

Paweł Nykiel

October 2024

1 Zadanie

1.1 Polecenie

Napisz program wyliczajacy przybliżenie pochodnej ze wzorów:
(a)

$$D_h f(x) \equiv \frac{f(x+h) - f(x)}{h},$$

(b)

$$D_h f(x) \equiv \frac{f(x+h) - f(x-h)}{2h}.$$

Przeanalizuj, jak zachowuje sie bład $|D_h f(x) - f'(x)|$ dla funkcji $f(x) = \sin(x^3)$ oraz punktu $x = 0.2$ przy zmianie parametru h dla różnych typów zmiennoprzecinkowych (`float`, `double`). Wykreśl $|D_h f(x) - f'(x)|$ w funkcji h w skali logarytmicznej.

Poeksperymentuj również używajac innych funkcji i punktów.

1.2 Cel ćwiczenia

Zbadanie różnych metod numerycznego obliczania pochodnych oraz analiza błedów zwiazanych z tymi metodami. Porównanie dokładności przybliżenia pochodnej dla różnych wartości parametru h i typów zmiennoprzecinkowych. Wykres umożliwia wizualizacje zachowania błedów w kontekście obliczeń numerycznych.

2 Program

2.1 Jezyk programowania

Wyniki zostały policzone przy pomocy biblioteki *numpy* natomiast wykresy zostały narysowane z użyciem *matplotlib*. Jezyk programowania to *Python*

2.2 Jak uruchomić?

Aby uruchomić program należy użyć python3 num1.py

3 Opis ćwiczenia

3.1 Analiza problemu

•Zrozumienie celu zadania, którym jest obliczenie błedu (różnicy przybliżenia pochodnej funkcji $f(x) = sin(x^3)$ i dokładnej pochodnej) w punkcie $x = 0.2$ z wykorzystaniem dwóch metod
•Identyfikacja błedów zwiazanych z przybliżeniem pochodnej w zależności od wybranej metody oraz wartości parametru h.
•Rozważenie różnych typów zmiennoprzecinkowych (float, double) oraz ich wpływu na precyzje obliczeń.

3.2 Implementacja algorytmu przybliżania pochodnej

• Zaimplementowanie dwóch metod różnicowych do obliczania przybliżenia pochodnej:

Metoda 1: $D_h f(x) = \frac{f(x+h)-f(x)}{h}$
Metoda 2: $D_h f(x) = \frac{f(x+h)-f(x-h)}{2h}$

•Zastosowanie petli do zmiany wartości parametru hh oraz obliczenie błedu Dhf(x)f(x)Dhf(x)f(x) dla każdej wartości.
•Zbieranie wyników do analizy oraz późniejszej wizualizacji.

4 Wstep - teoria

1. Pochodna funkcji

Pochodna funkcji to fundamentalne pojecie w analizie matematycznej, które opisuje tempo zmian funkcji w danym punkcie. Definiowana jest jako granica ilorazu różnicowego, kiedy h daży do zera:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

W praktyce jednak, ze wzgledu na niemożliwość użycia nieskończenie małego kroku h, należy stosować metody numeryczne do obliczenia przybliżonej wartości pochodnej. Metody te wykorzystuja skończone wartości h, co prowadzi do powstawania pewnych błedów obliczeniowych.

2. Metody przybliżania pochodnej

W zadaniu wykorzystywane sa dwie metody numeryczne do przybliżania pochodnej:

- **Metoda różnicy w przód (forward difference):**
  Ta metoda wykorzystuje wartości funkcji w punktach x oraz $x + h$ do obliczenia przybliżenia pochodnej w punkcie x. Definiowana jest wzorem:

$$D_h f(x) = \frac{f(x+h) - f(x)}{h}$$

- **Metoda różnicy centralnej (central difference):**
  Druga metoda jest wykorzystanie wartości funkcji w punktach $x+h$ oraz $x-h$, co pozwala na zredukowanie błedu numerycznego. Definiowana jest wzorem:

$$D_h f(x) = \frac{f(x+h) - f(x-h)}{2h}$$

3. Bład przybliżenia pochodnej

W obliczeniach numerycznych ważna kwestia jest ocena dokładności przybliżenia pochodnej. Bład przybliżenia pochodnej zależy od kilku czynników:

- **Bład truncacji**: Wynika z użycia skończonego kroku h.
- **Bład zaokraglenia**: Wynika z ograniczonej precyzji komputerowej (np. używania zmiennoprzecinkowych typów danych takich jak `float` i `double`).

5 Wyniki

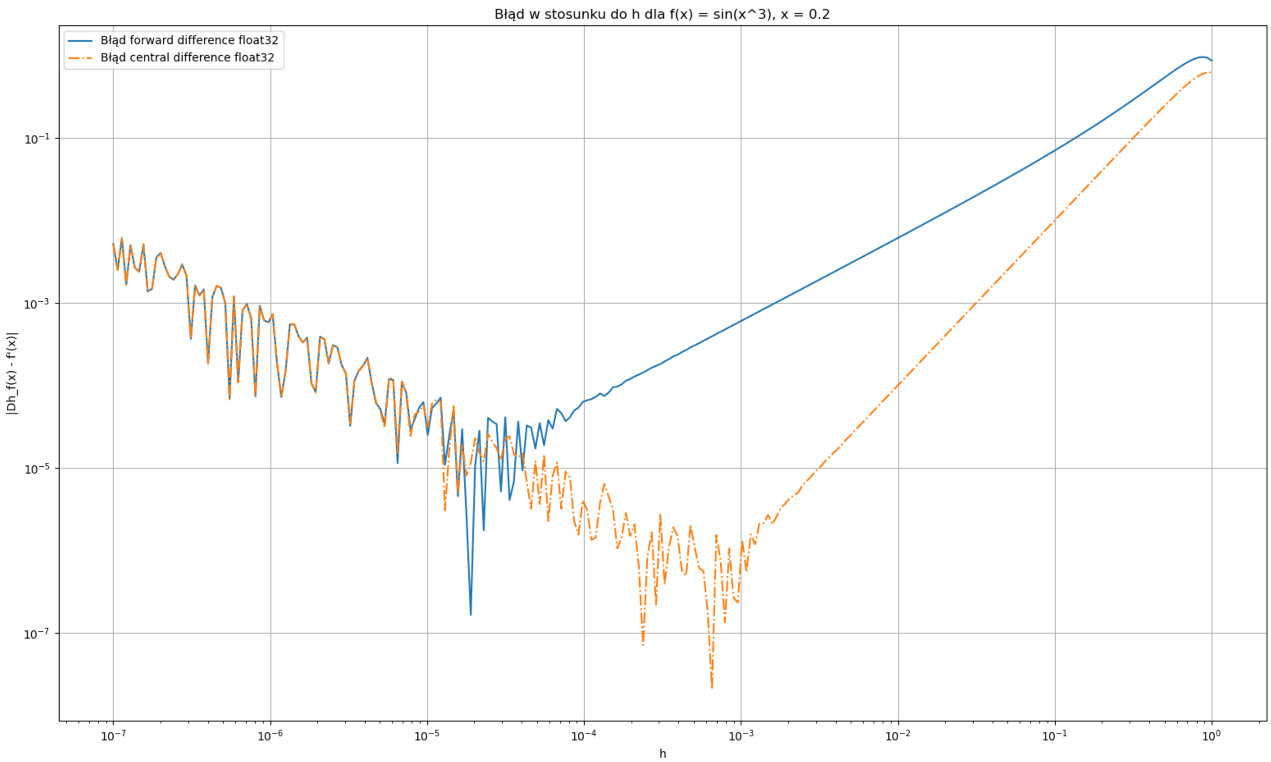

Figure 1: Wykres dla $float32$

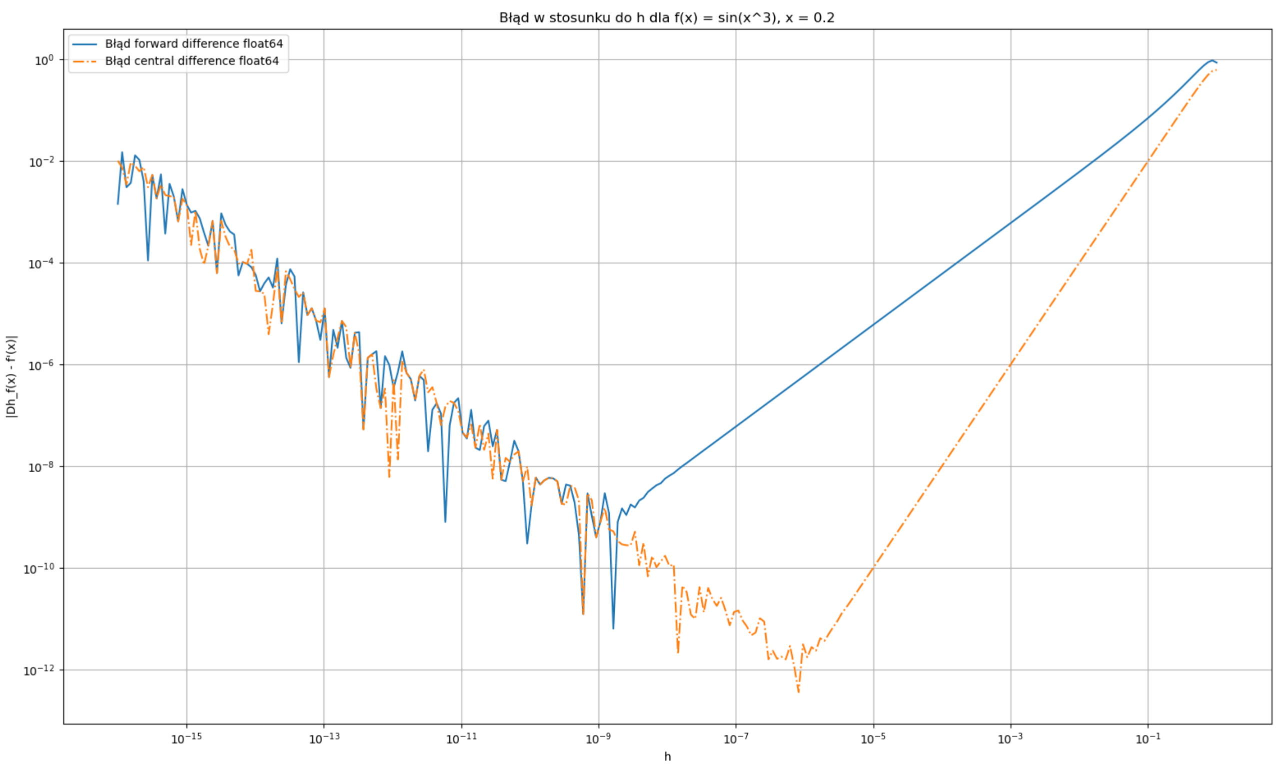

Figure 2: Wykres dla $float64$

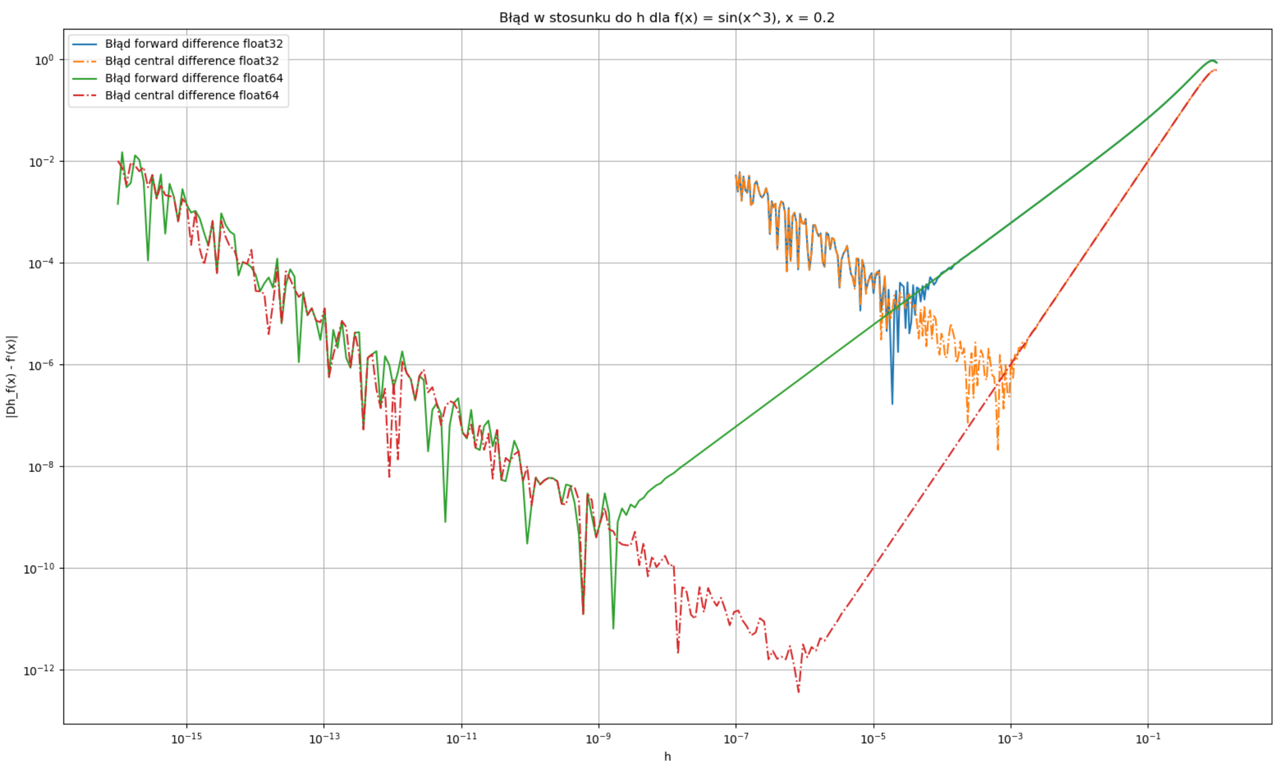

Figure 3: Wykres dla $float32$ oraz $float32$

6 Wnioski

Wartość błedu zmienia sie wraz ze zmiana kroku h. Istnieja takie wartości h, dla których wyliczenie pochodnej bedzie bardziej dokładne niż dla innych. Jeśli h bedzie zbyt duże, pojawia sie błedy truncacji, wynikajace z pominiecia wyższych rzedów w rozwinieciu Taylora. Natomiast przy bardzo małych wartościach h błedy beda wynikały z zaokragleń przy odejmowaniu, czyli ze straty precyzji. Dzieje sie tak, gdy różnice miedzy liczbami sa bardzo małe, a komputer nie jest w stanie ich dokładnie przedstawić ze wzgledu na ograniczona precyzje zmiennoprzecinkowa. Z wykresów możemy wywnioskować, że najmniejszy bład uzyskujemy przy użyciu typu `double` oraz metody różnicy centralnej. Typ `double` jest znacznie dokładniejszy niż `float` w obu metodach numerycznych, co wynika z faktu, że `double` wykorzystuje 64 bity do przechowywania liczb, podczas gdy `float` tylko 32 bity. Metoda różnicy centralnej daje bardziej dokładne wyniki niż metoda różnicy w przód.