Obliczenia Naukowe - Laboratorium 4
Interpolacja wielomianowa

Krzysztof Zając

6 grudnia 2025

Wstęp

Celem laboratorium była implementacja algorytmów interpolacji wielomianowej w bazie Newtona oraz analiza numeryczna dokładności interpolacji w zależności od doboru węzłów.

1 Zadanie 1 - Ilorazy różnicowe

1.1 Opis problemu

Implementacja funkcji `ilorazyRoznicowe` obliczającej współczynniki wielomianu interpolacyjnego Newtona $c_k = f[x_0, \ldots, x_k]$. Wymagano użycia struktury danych o złożoności pamięciowej $O(n)$.

1.2 Rozwiązanie

Zastosowano algorytm działający na tablicy jednowymiarowej "in-place". W pętli zewnętrznej ($j = 2 \ldots n$) i wewnętrznej ($i = n \ldots j$) aktualizowano wartości tablicy wg wzoru:

$$f[x_i, \ldots, x_{i+k}] = \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$$

1.3 Wyniki

Funkcja zwraca wektor ilorazów różnicowych. Poprawność zweryfikowano w kolejnych zadaniach (poprawne rysowanie wielomianów).

1.4 Wnioski

Algorytm ma złożoność czasową $O(n^2)$ i pamięciową $O(n)$. Uniknięcie alokacji macierzy trójkątnej jest korzystne dla zarządzania pamięcią.

2 Zadanie 2 - Wartość wielomianu (Horner)

2.1 Opis problemu

Obliczenie wartości wielomianu Newtona $N_n(t)$ w czasie $O(n)$ przy użyciu uogólnionego schematu Hornera.

2.2 Rozwiązanie

Zaimplementowano funkcję `warNewton`. Algorytm działa rekurencyjnie "w tył":

$$w_n := c_n, \quad w_k := c_k + (t - x_k)w_{k+1} \quad (k = n-1, \ldots, 0)$$

Wynikiem jest $N_n(t) = w_0$.

2.3 Wyniki

Funkcja została wykorzystana do generowania wykresów interpolacji.

2.4 Wnioski

Schemat Hornera jest standardem w obliczaniu wartości wielomianów ze względu na minimalizację liczby operacji zmiennoprzecinkowych i stabilność numeryczną.

3 Zadanie 3 - Postać naturalna

3.1 Opis problemu

Przeliczenie współczynników z postaci Newtona na postać naturalną $P(x) = \sum a_i x^i$ w czasie $O(n^2)$.

3.2 Rozwiązanie

Wykorzystano algorytm iteracyjnego mnożenia wielomianu przez dwumian $(x - x_k)$. W każdym kroku wektor współczynników jest aktualizowany, symulując operację $P_{new}(x) = c_k + (x - x_k)P_{old}(x)$.

3.3 Wyniki

Otrzymano funkcję `naturalna` zwracającą wektor współczynników $a_0 \dots a_n$.

3.4 Wnioski

Algorytm pozwala na zmianę bazy bez konieczności rozwiązywania układu równań z macierzą Vandermonde'a (co byłoby źle uwarunkowane).

4 Zadanie 4 - Funkcja interpolująca i rysująca

4.1 Opis problemu

Napisanie funkcji `rysujNnfx`, która dla zadanej funkcji f, przedziału $[a, b]$ i stopnia n obliczy wielomian interpolacyjny i wygeneruje wykres porównawczy. Funkcja ma obsługiwać węzły równoodległe oraz węzły Czebyszewa.

4.2 Rozwiązanie

Funkcja integruje rozwiązania z Zadań 1 i 2: 1. Generuje węzły interpolacji x_k (wzory analityczne dla równoodległych lub zer wielomianu Czebyszewa przeskalowanych na $[a, b]$). 2. Oblicza wartości $f(x_k)$. 3. Wyznacza ilorazy różnicowe (Zad. 1). 4. Próbkuje wielomian na gęstej siatce punktów używając Hornera (Zad. 2) i rysuje wykres.

4.3 Wyniki

Funkcja generuje obiekty wykresów, które zostały zaprezentowane w sekcjach dotyczących zadań 5 i 6.

4.4 Wnioski

Funkcja umożliwia szybką wizualną ocenę jakości interpolacji oraz badanie zjawisk takich jak efekt Rungego, bez ręcznego powtarzania obliczeń współczynników.

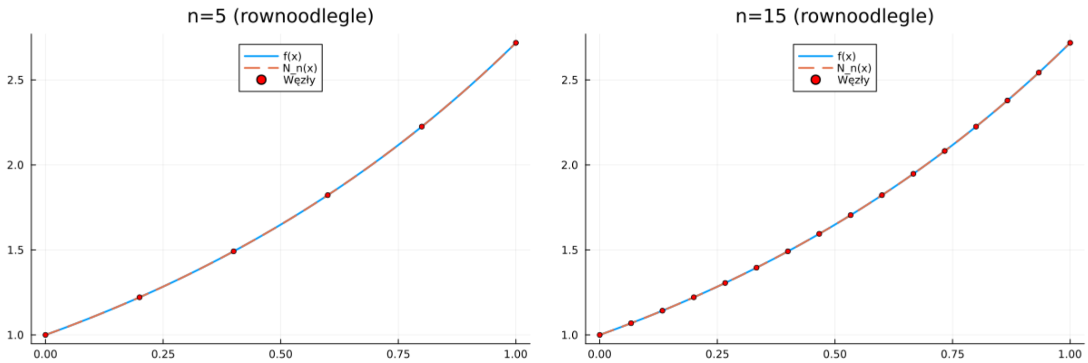
5 Zadanie 5 - Interpolacja funkcji gładkich

5.1 Opis problemu

Testowanie interpolacji na węzłach równoodległych dla funkcji e^x oraz $x^2 \sin(x)$.

5.2 Wyniki

Dla obu funkcji ciąg wielomianów interpolacyjnych jest zbieżny do funkcji interpolowanej.



(a) $e^x, n = 5$

(b) $e^x, n = 15$

Rysunek 1: Interpolacja funkcji e^x. Błąd maleje wraz ze wzrostem n.

5.3 Wnioski

Błąd interpolacji wyraża się wzorem:

$$E_n(x) = f(x) - N_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^{n} (x - x_i)$$

Dla badanych funkcji ($e^x, \sin x$) pochodne wszystkich rzędów są wspólnie ograniczone przez pewną stałą M (w przedziale). Węzły są równoodległe, ale czynnik silni $(n+1)!$ w mianowniku rośnie znacznie szybciej niż iloczyn węzłowy $\prod(x - x_i)$. Dlatego dla funkcji całkowitych (analitycznych w całej płaszczyźnie zespolonej) interpolacja na węzłach równoodległych jest zbieżna jednostajnie.

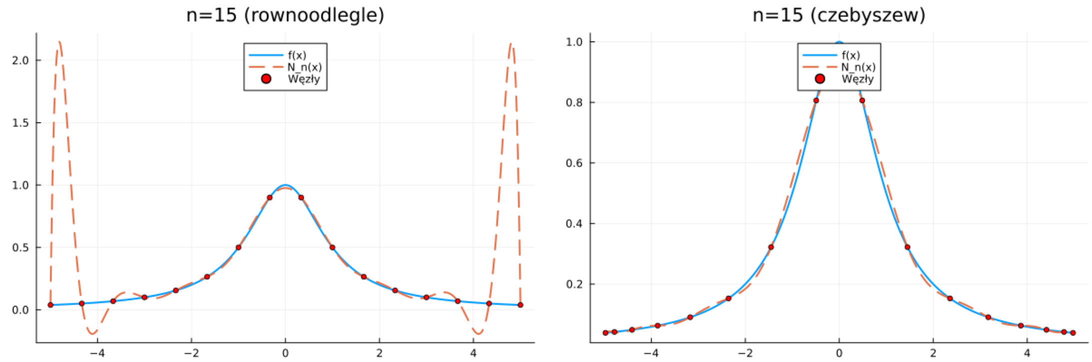
6 Zadanie 6 - Zjawisko Rungego

6.1 Opis problemu

Analiza interpolacji funkcji $f(x) = \frac{1}{1+x^2}$ (Rungego) oraz $f(x) = |x|$ przy użyciu węzłów równoodległych i Czebyszewa.

6.2 Wyniki

Dla funkcji Rungego przy węzłach równoodległych zaobserwowano silne oscylacje na krańcach przedziału (błąd rośnie z n). Zastosowanie węzłów Czebyszewa eliminuje ten problem.



(a) Węzły równoodległe ($n = 15$) - Rozbieżność (b) Węzły Czebyszewa ($n = 15$) - Zbieżność

Rysunek 2: Porównanie dla funkcji Rungego.

6.3 Wnioski

1. Dlaczego interpolacja na węzłach równoodległych zawodzi? Dla węzłów równoodległych wielomian węzłowy $\omega(x) = \prod_{i=0}^{n}(x - x_i)$ osiąga bardzo duże wartości blisko krańców przedziału. Dodatkowo, tzw. stała Lebesgue'a Λ_n (mierząca wrażliwość interpolacji) rośnie wykładniczo ($\sim 2^n$). Dla funkcji Rungego pochodne wysokich rzędów rosną szybciej niż $(n+1)!$, co w połączeniu z dużymi wartościami $\omega(x)$ powoduje rozbieżność ($E_n(x) \to \infty$).

2. Dlaczego węzły Czebyszewa działają? Węzły te są rzutem punktów równoodległych z okręgu na odcinek, co powoduje ich zagęszczenie na krańcach przedziału (z gęstością $\frac{1}{\sqrt{1-x^2}}$). Taki rozkład węzłów minimalizuje normę sup wielomianu węzłowego $\|\omega(x)\|_\infty$ (własność minimaksowa). Stała Lebesgue'a rośnie wtedy tylko logarytmicznie ($\sim \ln n$). Dzięki temu błąd interpolacji jest "trzymany w ryzach"i interpolacja jest zbieżna dla każdej funkcji spełniającej warunek Lipschitza, a tym bardziej dla funkcji analitycznych jak funkcja Rungego.

3. Przypadek $|x|$: Funkcja nie jest różniczkowalna w zerze. Interpolacja wielomianowa nie jest w stanie idealnie odwzorować "ostrza", ale węzły Czebyszewa pozwalają na znacznie lepsze przybliżenie globalne (unikanie efektu Gibbsa przy brzegach) niż węzły równoodległe.