Interpolacja wielomianowa

Mateusz Kwiatkowski

1 Problem

Wyznaczanie wielomianu interpolacyjnego Lagrange'a (w postaci Newtona) danej funkcji. Wykonanie wykresów: funkcji, wielomianu interpolacyjnego i błędu interpolacji. Obliczanie wartości funkcji, przybliżenia i błędu w m równoodległych punktach (z podanego przez użytkownika przedziału).
Do obliczenia wartości wielomianu interpolacyjnego zastosować uogólniony schemat Hornera.

2 Wstęp

Celem projektu było zaimplementowanie algorytmu tworzącego wielomian interpolacyjny Lagrange'a w postaci Newtona, wykorzystanie go do obliczania wartości w punkcie i wyznaczenie błędu interpolacji. Pierwszą część zrealizowałem w funkcji Newton(), która przyjmuje dwa wektory długości $n + 1$ reprezentujące zbiór węzłów interpolacji i wartości w tych punktach, a zwraca współczynniki wielomianu interpolacyjnego stopnia $n - tego$. Warto zaznaczyć, że nie są one podane w bazie standardowej, ale ten temat rozwinę w kolejnej sekcji. Do obliczania wartości wielomianu w punkcie służy funkcja wielomian(). Jej wartości wejściowe to:

- punkt, w którym chcemy obliczać wartość wielomianu
- wektor n współczynników wielomianu (które otrzymaliśmy dzięki funkcji Newton())
- wektor $n + 1$ węzłów interpolacji

Trzeba zaznaczyć, że ta funkcja wykorzystuje schemat Hornera, by zoptymalizować złożoność obliczeniową algorytmu.
W celu sprawdzenia błędu interpolacji napisałem dwie funkcje. Pierwsza z nich to Newton_blad(). Przyjmuje ona:

- funkcję interpolowaną
- wektor węzłów interpolacji
- dwa punkty będące początkiem i końcem przedziału, na którym obliczać będziemy wartości
- m - liczbę punktów, w których będziemy obliczać błąd

Newton_blad() zwraca wartości funkcji, wielomianu interpolacyjnego i błędu interpolacji w m równoodległych punktach z przedziału podanego przez użytkownika. Ostatnią funkcją są wykresy().
Ta funkcja rysuje wykresy funkcji interpolowanej, wielomianu interpolacyjnego i błędu interpolacji, na podstawie podanej funkcji, węzłów interpolacji i wpisanego przedziału.
Powyższe funkcje wykorzystałem do opisania problemu interpolacji, a najciekawsze przykłady obliczeniowe opisałem zarówno w sekcji 4, jak i w skrypcie Newton_testy.m.

3 Omówienie metody i jej implementacji

3.1 Wyznaczanie wielomianu interpolacyjnego Lagrange'a w postaci Newtona

Problem interpolacji polega na przybliżaniu dowolnej funkcji f wielomianem p przy pomocy węzłów interpolacji $x_0, x_1, ..., x_n$, czyli punktów, w których znamy wartość f, w taki sposób że $f(x_i) = p(x_i)$. Dla $n+1$ węzłów jesteśmy w stanie skonstruować wielomian $n - tego$ stopania. Wielomian interpolacyjny w postaci Newtona jest przedstawiony w bazie $1, x - x_0, (x - x_0)(x - x_1), ...,$ (x-x$_0$)$(x - x_1)...(x - x_{n-1})$.

$$p(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + ... + c_n(x - x_0)(x - x_1)...(x - x_{n-1})$$

By jednoznacznie wyznaczyć wielomian interpolacyjny trzeba obliczyć wartości współczynników: $c_0, ..., c_n$.

Obliczenie c_0 jest trywialne, ponieważ $c_0 = f(x_0) = p(x_0)$. By poznać kolejne współczynniki będziemy musieli obliczyć ilorazy różnicowe korzystając z poniższej definicji:

$$f_{j,j+1,...,j+k} = \frac{f_{j+1,...,j+k} - f_{j,...,j+k-1}}{x_{j+k} - x_j} \text{ dla } j = 0, .., n - k$$

$$c_i = f_{0,1,...,i}$$

Funkcja Newton() używa powyższego wzoru do obliczania ilorazów kolejnych rzędów (dla kolejnych k, zaczynając od 1), przy czym proces ten jest sfaktoryzowany, jako że można obliczyć od razu cały wektor ilorazów ustalonego rzędu. Jest to iloraz skalarny różnicy wektorów, który wyliczany jest z pętli:

$wsp = [f(x_0), ..., f(x_n)];$
for $i = 2, ..., (n + 1)$
    $wsp(i : (n + 1)) = (wsp(i : (n + 1)) - wsp((i - 1) : n))./(x(i : (n + 1)) - x(1 : (n + 2 - i)));$
end

Wektory numerowane są od 1, bo używamy tu składni "matlabowej". Po całym powyższym procesie otrzymamy $wsp(i) = f_{0,1,...,i}$, zatem otrzymany wektor jest wektorem współczynników c.

3.2 Prezentacja działania funkcji

By zaprezentować sposób działania algorytmu, przeanalizujmy interpolację funkcji $f(x) = x^4$ w węzłach $x_0 = -1, x_1 = 0, x_2 = 1, x_3 = 2$. Wektory będziemy numerować od 0. Na początku $wsp(0 : 3) = [f(-1), f(0), f(1), f(2)] = [1, 0, 1, 16]$, czyli $f_0 = 1$.
Wykonujemy kolejne przekształcenia:

$wsp(1 : 3) = [f_{01}, f_{12}, f_{23}] = (wsp(1 : 3) - wsp(0 : 2))./(x(1 : 3) - x(0 : 2)) = [-1, 1, 15]./[1, 1, 1] = [-1, 1, 15]$
$wsp(2 : 3) = [f_{012}, f_{123}] = (wsp(2 : 3) - wsp(1 : 2))./(x(2 : 3) - x(0 : 1)) = [2, 14]./[2, 2] = [1, 7]$
$wsp(3) = f_{0123} = (wsp(3) - wsp(2))./(x(3) - x(0)) = 6/3 = 2$

W taki sposób otrzymaliśmy, że $[c_0, c_1, c_2, c_3] = [f_0, f_{01}, f_{012}, f_{0123}] = wsp = [1, -1, 1, 2]$, czyli wielomian interpolacyjny ma następującą postać: $p(x) = 1 - (x + 1) + (x + 1)x + 2(x + 1)x(x - 1)$

3.3 Obliczanie wartości wielomianu interpolacyjnego w punkcie przy użyciu schematu Hornera

Skoro otrzymaliśmy już współczynniki wielomianu to warto zastanowić się w jaki sposób obliczać jego wartość w punkcie, tak by nie wpłynęło to na złożoność algorytmu. Z pomocą w tej sytuacji przychodzi schemat Hornera.

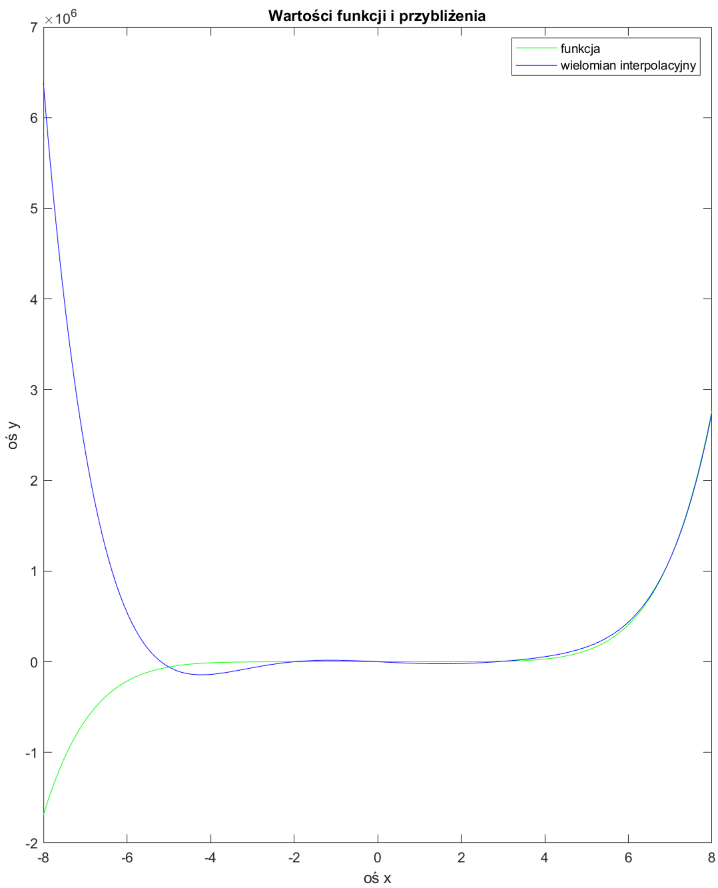
$p(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + ... + c_n(x - x_0)(x - x_1)...(x - x_{n-1}) =$
$c_0 + (x - x_0)(c_1 + c_2(x - x_1) + ... + c_n(x - x_1)...(x - x_{n-1})) =$
$c_0 + (x - x_0)(c_1 + (x - x_1)(c_2 + ... + c_n(x - x_2)...(x - x_{n-1}))) = ... =$
$c_0 + (x - x_0)(c_1 + (x - x_1)(...(c_{n-2} + (x - x_{n-2})(c_{n-1} + c_n(x - x_{n-1})))...))$
W funkcji wielomian() ten schemat został zaimplementowany przy pomocy poniższej pętli:

$p = c_n;$
for $i = n - 1, n - 2, ..., 0$
  $p = c_i + (x - x_i)p;$
end

4 Przykłady obliczeniowe i analiza wyników

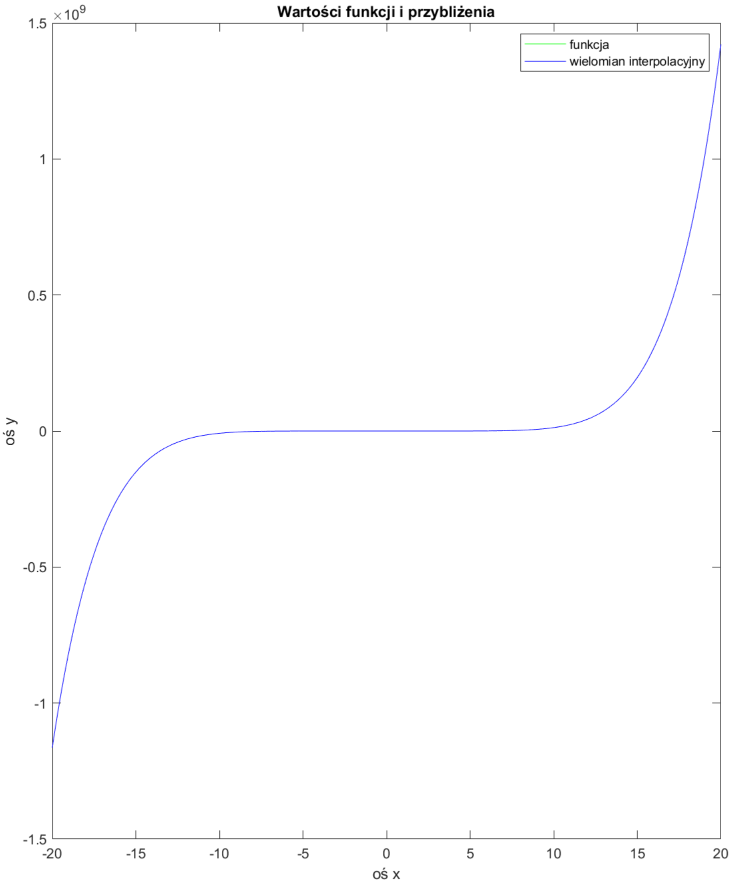
4.1 Wielomiany

Interpolowanie wielomianów jest zagadnieniem stosunkowo łatwym, ponieważ oczywiste jest, że dla odpowiednio wielu węzłów wielomian interpolacyjny pokryje się z wielomianem interpolowanym. Wynika to z wniosku z podstawowego twierdzenia algebry, który mówi, że dwa różne wielomiany mogą mieć maksymalnie n punktów wspólnych, gdzie n to maksimum ze stopni wielomianów. Sprawdźmy zatem jak przebiega interpolacja wielomianu $f(x) = x^7 + 2x^6 + 4x^5 + 3x^4 - 5x^2 + 13x$. Na początek przybliżmy ten wielomian wielomianem stopnia szóstego w węzłach: -5, -2, 0, 3, 7, 9, 12. Do narysowania poniższego wykresu wykorzystałem funkcję wykresy().



Jak widać, przybliżenie na przedziale (-2,8) jest dobre, jednak wielomiany mają stopnie różnej parzystości, co sprawia, że wielomian interpolowany w minus nieskończoności zbiega do minus nieskończoności a wielomian interpolacyjny do plus nieskończoności, zatem błąd poza określonym powyżej przedziałem jest nieograniczony.

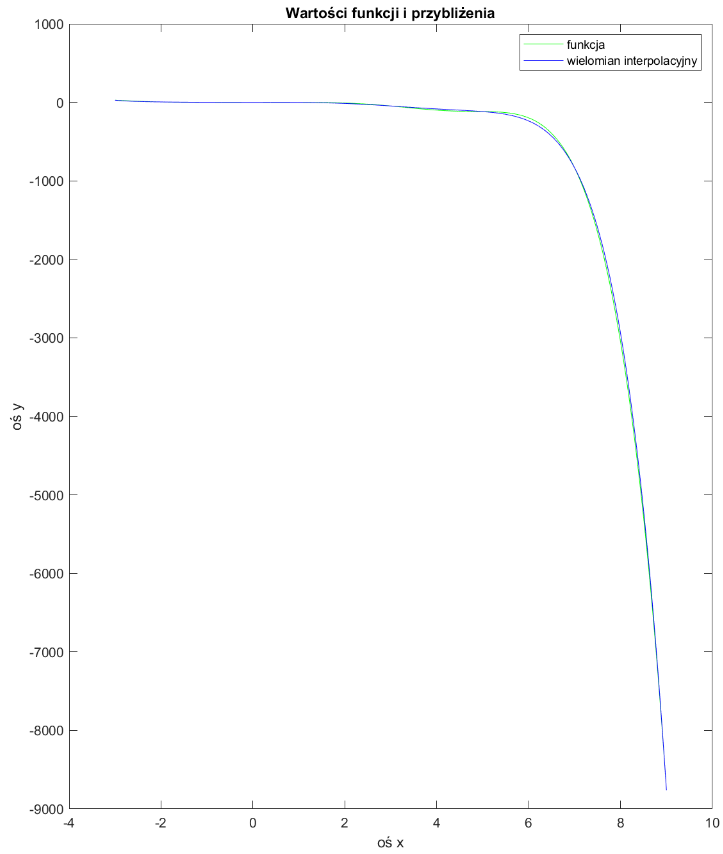
Dodajmy teraz dwa węzły interpolacji w punktach 13 i 20. To sprawia, że przybliżamy wielomian stopnia siódmego wielomianem stopnia maksymalnie ósmego, zatem możemy podejrzewać, że przybliżenie będzie dobre.



Okazuje się, że wykresy tych funkcji pokrywają się, ponieważ wyliczony wielomian interpolacyjny jest równy wielomianowi interpolowanemu.

4.2 Przybliżenia w węzłach interpolacji

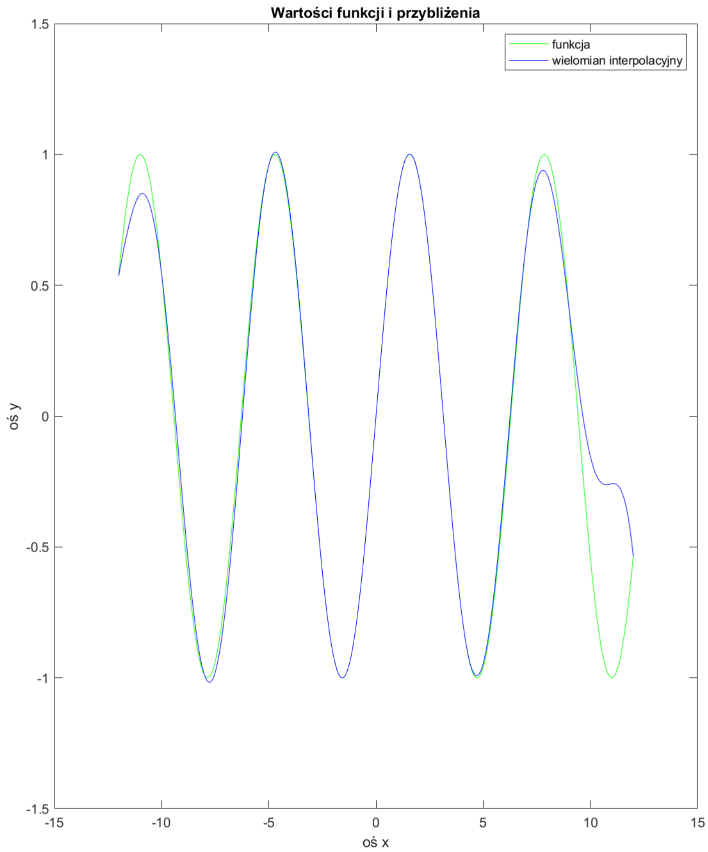
Podstawowy warunek interpolacji mówi, że wielomian interpolowany musi być równy funkcji interpolowanej w węzłach. Zweryfikowałem to przy użyciu metody Newton_blad(). Do sprawdzenia wykorzystałem funkcję $f(x) = xsin(x) + x^3cos(x) - e^x$ oraz węzły interpolacji 0, 2, 4, 6, 8, 10. W wyniku otrzymałem, że błąd interpolacji w tych węzłach jest równy odpowiednio: $0, 0, 0, 0,$ -4.55$\cdot 10^{-13}, 3.638 \cdot 10^{-12}$. Niezerowe błędy spowodowane są zaokrągleniami podczas obliczeń wartości wielomianu interpolacyjnego, jednak są one tak małe, że nie mają praktycznie żadnego wpływu na wyniki.
Możemy również zbadać jaki jest błąd interpolacji na przedziale, na którym jest wiele węzłów. Teraz przeprowadzę interpolację powyższej funkcji z węzłami: -3, -2, 0, 1, 3, 5, 7, 9.



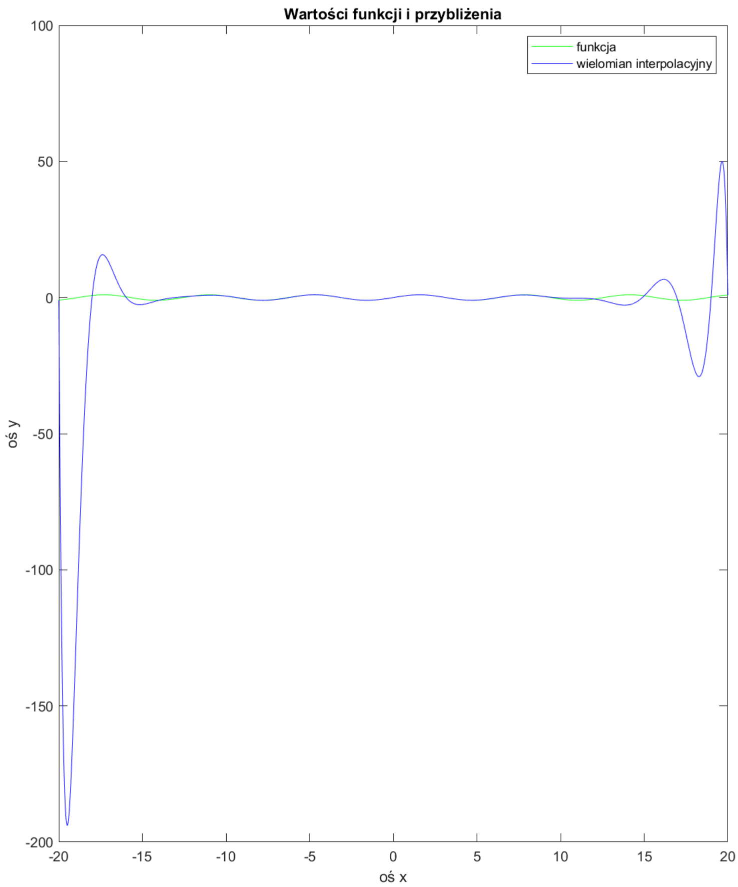
Jak widać, na przedziale (-3,9) między skrajnymi węzłami interpolacji wykresy obu funkcji praktycznie pokrywają się. Co ciekawe, błąd w punkcie x=8 wynosi ponad 100, jednak ciężko zauważyć to na wykresie, ponieważ błąd względny jest bardzo mały. Wniosek z tego jest taki, że interpolacja wielomianowa jest dobrym narzędziem do przybliżania funkcji na przedziale, jeśli mamy danych wiele węzłów.

4.3 Funkcje okresowe

Przybliżanie funkcji okresowych wielomianami na całej dziedzinie rzeczywistej nie jest dobrym rozwiązaniem, ponieważ wielomiany nie są ogarniczone i poza określonym przedziałem rosną do minus nieskończoności lub maleją do minus nieskończoności.
Przeprowadzę interpolację funkcji sinus w węzłach -20, -18, -16, -14, -12, -10, -8, -5, -2, 0, 1, π, 4, 7, 9, 12, 15, 17, 19, 20. Na początku zobaczmy jak wygląda to przybliżenie na przedziale (-12,12) blisko węzłów interpolacji.



Widać, że na podanym przedziale przybliżenie jest dobre. Duża liczba węzłów sprawiła, że wielomian interpolacyjny jest 19-stego stopnia, co powoduje, że dla x bliskich zeru wykresy obu funkcji pokrywają się. Zobaczmy jednak jak zachowuje się wielomian na większym przedziale.



Ten wykres ilustruje uwagę, którą napisałem na początku tej podsekcji, że poza małym przedziałem przybliżenie jest bardzo złe. W tym przypadku błąd interpolacji funkcji sinus wynosi prawie 200, a na całej osi rzeczywistej rośnie on do nieskończoności, co oznacza, że dobra interpolacja funkcji okresowej na całej osi jest niemożliwa.

5 Podsumowanie

Interpolacja wielomianowa w postaci Newtona ma liniową złożoność obliczeniową i dobrze przybliża funkcję interpolowaną blisko węzłów interpolacji lub na przedziale, na którym jest dużo węzłów. Nie zawsze możemy jednak określić błąd poza tym przedziałem, co sprawia, że globalnie może on rosnąć w sposób nieograniczony. Jeśli funkcją interpolowaną jest wielomian to dla odpowiednio wielu węzłów otrzymamy interpolację o zerowym błędzie. Jeśli jednak interpolujemy funkcję ograniczoną lub okresową, to błąd interpolacji wraz ze wzrostem x będzie rósł do nieskończoności, ponieważ wielomiany nie są ograniczone na zbiorze liczb rzeczywistych. To sprawia, że za każdym razem, gdy rozwiązujemy problem interpolacji, musimy zadbać o odpowiedni dobór węzłów i musimy liczyć się z wystąpieniem błędu interpolacji.