Obliczenia naukowe - lista 4

Jakub Sofiński

Grudzień 2021

Zadanie 1

Algorytm zlicza z każdą iteracją kolejne kolumny ilorazów różnicowych. Pierwsza kolumna przyjmuje wszystkie wartości f[i], a pierwszy wiersz zawiera wszystkie kolejne obliczone ilorazy różnicowe postaci:
$f[x_0], f[x_0, x_1], ..., f[x_0, x_1, ..., x_n]$
Z każdą iteracją zapisujemy kolumna[1] jako wynik[i].

$$\begin{array}{lllll}
x_0 & f[x_0] \to & f[x_0, x_1] \to & f[x_0, x_1, x_2] \to & f[x_0, x_1, x_2, x_3] \\
x_1 & f[x_1] \nearrow\!\!\!\!\to & f[x_1, x_2] \nearrow\!\!\!\!\to & f[x_1, x_2, x_3] \nearrow & \\
x_2 & f[x_2] \nearrow\!\!\!\!\to & f[x_2, x_3] \nearrow & & \\
x_3 & f[x_3] \nearrow & & &
\end{array}$$

Obliczając każdy iloraz różnicowy odejmujemy od ilorazu z poprzedniej kolumny i wiersza niżej iloraz z poprzedniej kolumny, a różnicę dzielimy przez różnicę ostatniego argumentu obliczanego ilorazu z pierwszym argumentem, tak jak we wzorze poniżej.

$$f[x_0, x_1, ..., x_n] = \frac{f[x_1, x_2, ..., x_n] - f[x_0, x_1, ..., x_{n-1}]}{x_n - x_0}$$

Zadanie 2

Możliwe jest proste wyliczenie wartości w punkcie bezpośrednio z postaci Newtona, ale wykonane byłoby to w złożoności $O(n^2)$. Lepszym rozwiązaniem jest wyliczenie tego za pomocą schematu Hornera, co pozwala nam nie liczyć wielokrotnie $(x - x_i)$ i obliczyć wynik w złożoności $O(n)$.

Postać Newtona:
$w(x) = a_0 + \sum_{i=1}^{n} a_i \prod_{j=0}^{i-1} (x - x_j) = a_0 + a_1(x - x_0) + a_2(x - x_1)(x - x_0) + ... + a_n(x - x_{n-1})...(x - x_0)$

Mamy wielomian postaci:
$N_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + ... + f[x_0, x_1, ..., x_n](x - x_0)...(x - x_{n-1})$

wyciągamy kolejne $(x - x_0), (x - x_1), ...$ przed nawias otrzymując wzór:
$N_n(x) = f[x_0] + (x - x_0)(f[x_0, x_1] + ...(x_0 - x_{n-2})(f[x_0, ..., x_{n-1}] + (f[x_0, ..., x_n](x - x_{n-1})...)$
Z czego już łatwo zsumować kolejne części wzoru iterując od n-1 do 0.

Zadanie 3

Algorytm polega na kolejnym wymnażaniu wielomianu obecnego z (x - x[i-1]) i dodaniu $f[x_{i-1}]$

Tworzenie postaci Newtona
$f[x_n]$
$f[x_n](x - x_{n-1}) + f[x_{n-1}]$
$(f[x_n](x - x_{n-1}) + f[x_{n-1}])(x - x_{n-2}) + f[x_{n-2}]$
...

Tworzenie postaci naturalnej
$f[x_n]$
$f[x_n]x - f[x_n]x_{n-1} + f[x_{n-1}]$
$(f[x_n]x - f[x_n]x_{n-1} + f[x_{n-1}])(x - x_{n-2}) + f[x_{n-2}] =$
$= f[x_n]x^2 - f[x_n]x_{n-1}x + f[x_{n-1}]x - f[x_n]xx_{n-2} + f[x_n]x_{n-1}x_{n-2} - f[x_{n-1}]x_{n-2} + f[x_{n-2}]$
...

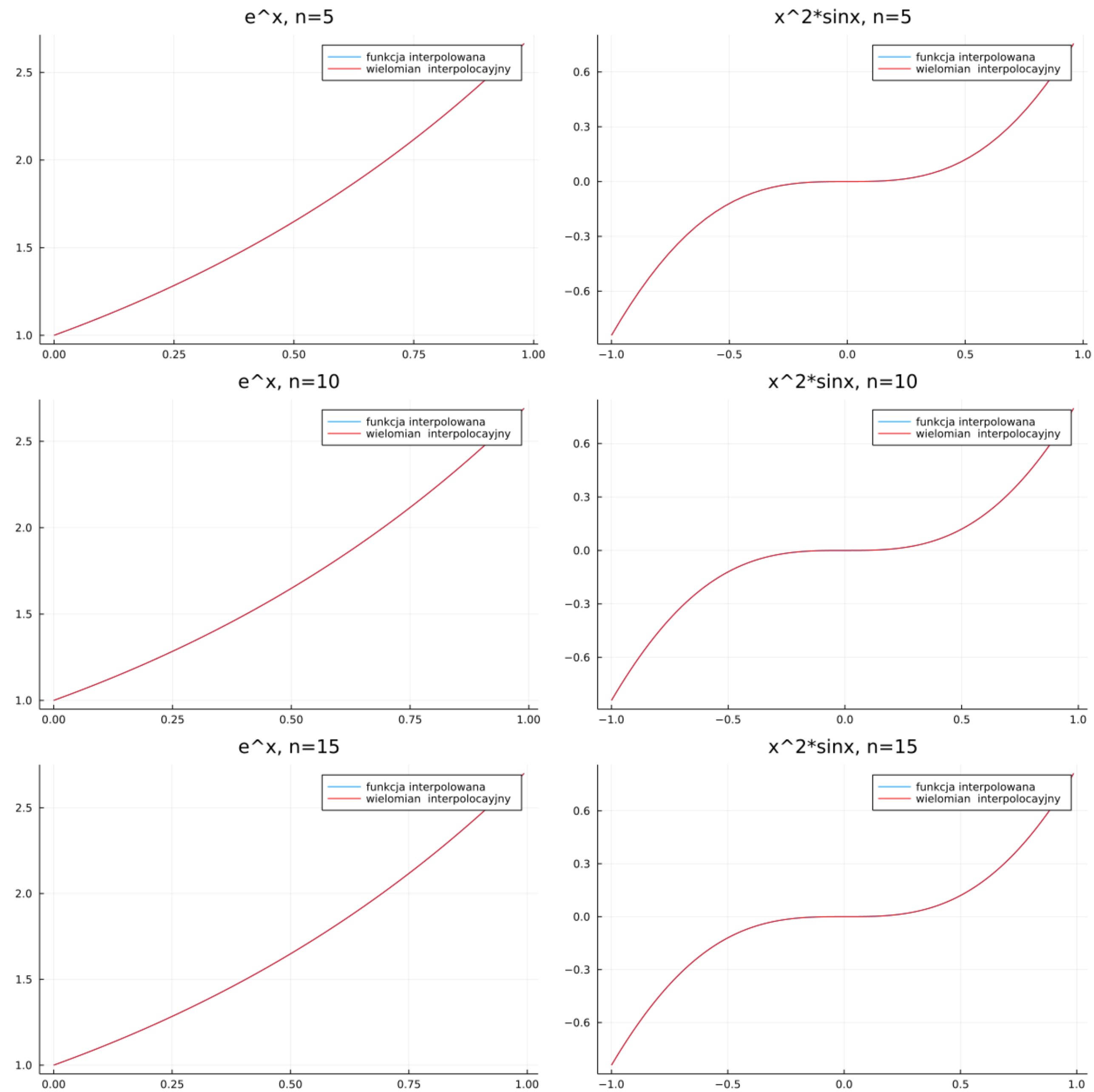
Współczynniki przy kolejnych potęgach x trzymane są wyliczone w jednej tabeli, np. z przykładu wyżej na drugiej pozycji tabeli (dla x^1) znajdowałoby się wyliczone $(f[x_n]x_{n-1} + f[x_{n-1}] - f[x_n]x_{n-2})$

Dla x:[-2.0, -1.0, 0.0, 1.0, 2.0, 3.0]; f:[-25.0, 3.0, 1.0, -1.0, 27.0, 235.0], fx: [-25.0, 3.0, 1.0, -1.0, 27.0, 235.0], co w postaci Newtona wygląda: $25 + 28(x + 2) - 15(x + 2)(x + 1) + 5(x + 2)(x + 1)x + 1(x + 2)(x + 1)x(x - 1)(x - 2)$. Funkcja naturalna w tym przypadku zwróci nam [1.0, -3.0, 0.0, 0.0, 0.0, 1.0], co odpowiada $1 - 3x^3 + x^5$.

Zadanie 4

Najpierw wyliczam równoodległe argumenty oraz ich wartości do wyliczenia ilorazów różnicowych. Następnie rysuje obie funkcje (na podstawie 100 * n punktów), dokładną za każdym razem wyliczając wartość z podanej funkcji oraz wielomian interpolacyjny wykonując dla każdego punktu warNewton().

Zadanie 5

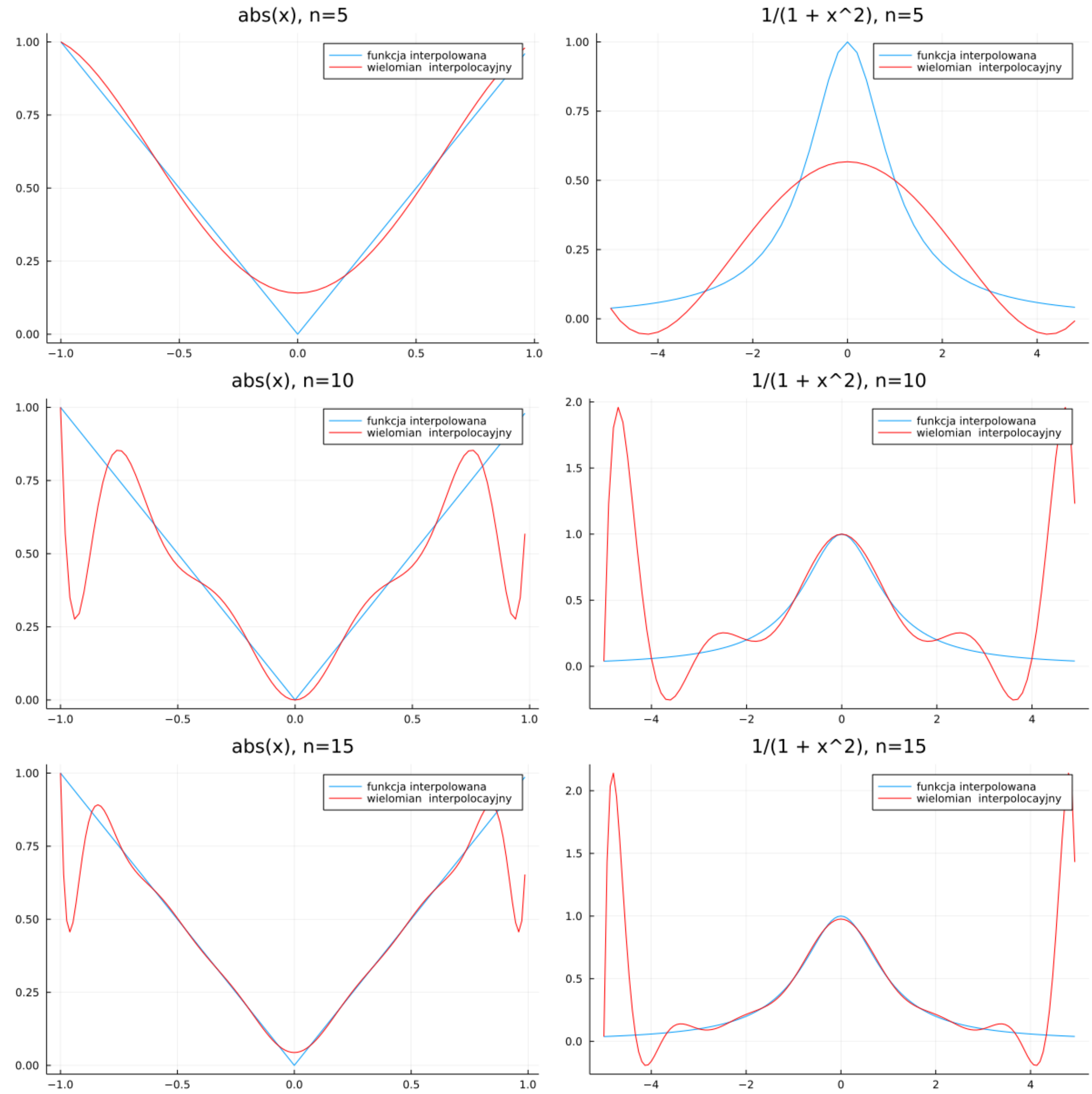


Wszystkie interpolacje wielomianowe w tym zadaniu świetnie pokryły funkcje dla każdej ilości węzłów.

n=5

Funckja	Argument	Funckja	Wielomian	różnica
e^x	0.5	1.6487212707001282	1.6487217891555785	$5.184554503490091e-7$
e^x	0.9	2.45960311115695	2.4596054194204404	$2.308263490569118e-6$
$x^2 * sinx$	0.5	0.11985638465105075	0.11978698679579014	$6.939785526061115e-5$
$x^2 * sinx$	0.9	0.6344947967982616	0.6349631163084872	0.00046831951022563434

Zadanie 6



Interpolacja wielomianowa dla funkcji z tego zadania przynosi bardzo duże błędy, szczególnie przy granicach przedziału interpolowanego.
Widać pogorszenie jakości interpolacji wielomianowej, mimo zwiekszenia liczby jej węzłów, jest to efekt Rungego. Rozwiązaniem tego problemu byłoby rozłożenie węzłów nieregularnie, gęściej przy krańcach przedziału, bo jak widać na wykresach, tam interpolacja przynosi największe błędy.