Barzowska Monika, 238143
Bienias Jan, 238201
3I TE1

08.01.2018

Algorytmy numeryczne
Zadanie 4: Aproksymacja

1. Wstęp

Celem zadania było przeprowadzenie pomiarów czasu działania programu dla gry z zadania 3, stosując aproksymację średniokwadratową dyskretną znaleźć wielomian aproksymacyjny dla każdej z serii pomiarów, wyliczyć jak długo trwałyby obliczenia każdą z metod.

2. Materiały i metody

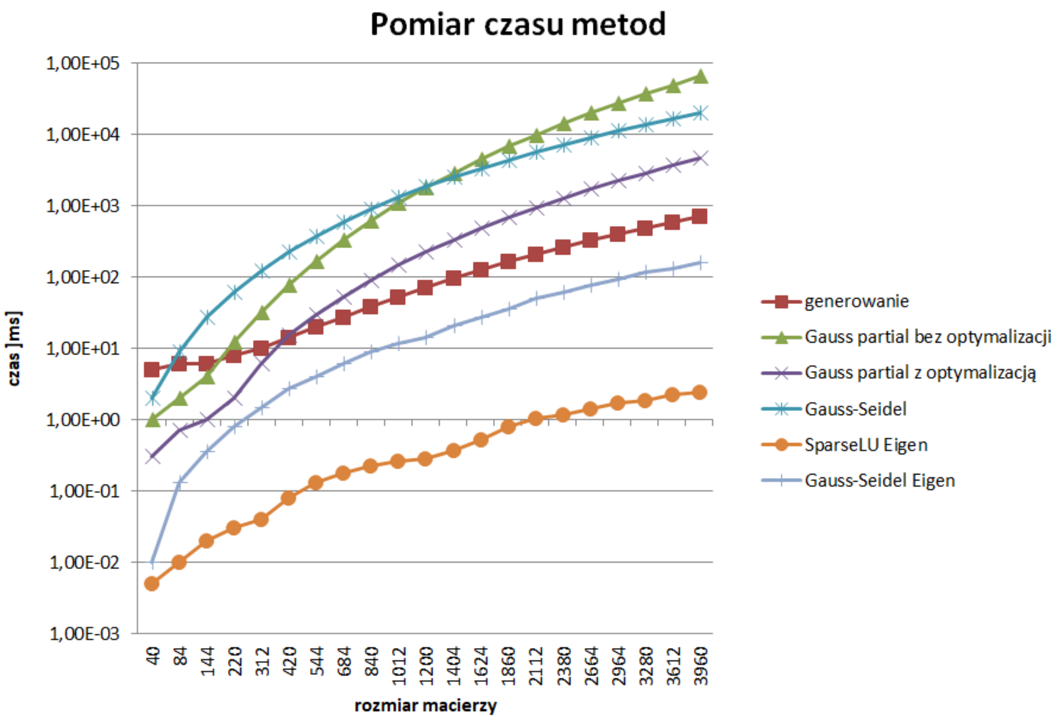
Pomiary zostały przeprowadzone dla 20 macierzy o rozmiarach od 40 do 3960 elementów.

Jedną z metod potrzebnych do wykonania zadania była metoda Gaussa-Seidela z wykorzystaniem struktur danych występujących w bibliotece Eigen3 (do własnej implementacji). Wyniki otrzymane z użyciem tej funkcji były zbliżone do wyników otrzymanych innymi metodami (różnica występowała dopiero na 11 miejscu po przecinku, ponieważ zgodnie z treścią zadania Ɛ = 1E-10). Dowodzi to poprawności implementacji algorytmu.

3. Wyniki

3.1. Pomiary czasu działania programu

Załączony po prawej wykres obrazuje zarówno wyniki pomiaru czasu działania, jak i samej generacji

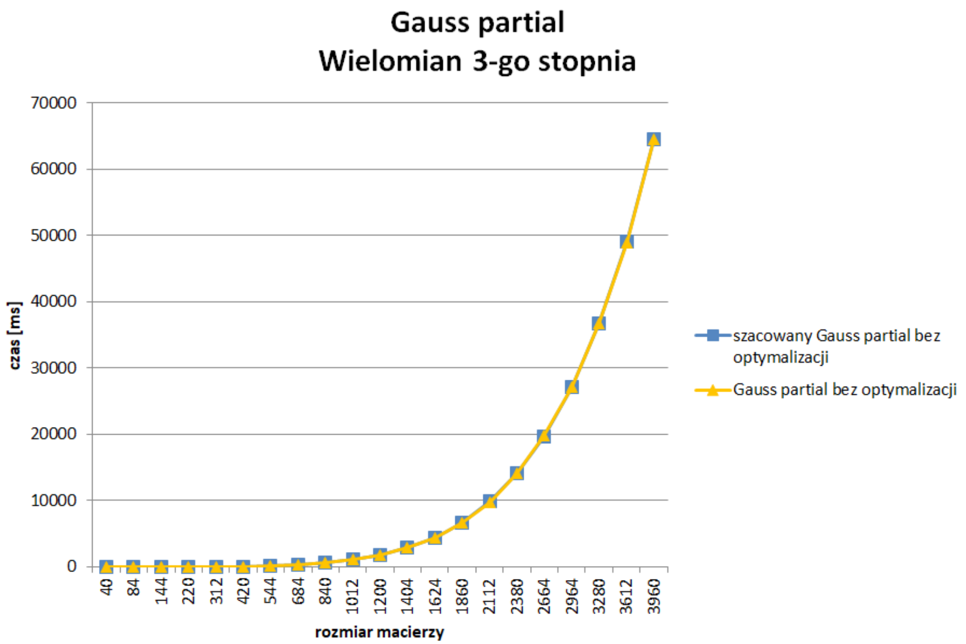


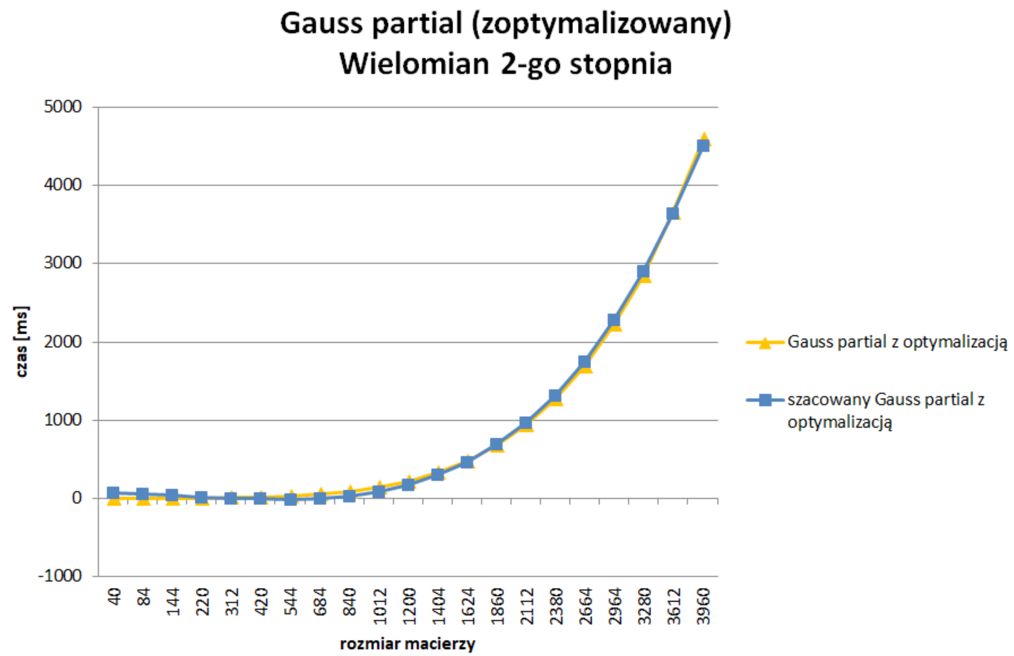
3.2. Wielomiany aproksymacyjne dla każdej z serii pomiarów

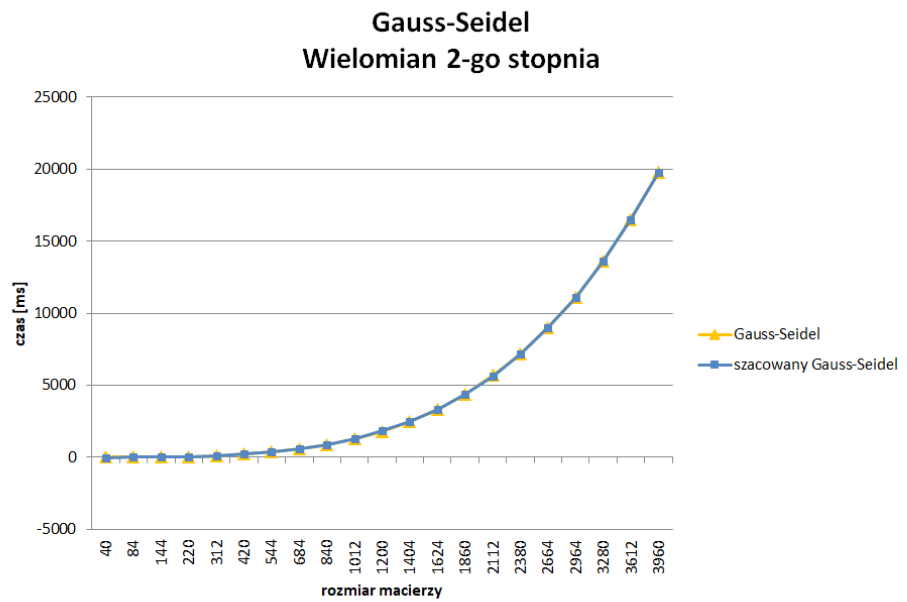
Metoda	**Wzór**
Gauss partial bez optymalizacji	$F(x)=1{,}02010019765106\text{E-}06*x^3+0{,}000105087710764997*x^2-0{,}125006604600517*x+22{,}1180704256306$
Gauss partial zoptymalizowany	$F(x)=0{,}000377646433521181*x^2-0{,}379193140620823*x+84{,}089339416471$
Gauss-Seidel	$F(x)=0{,}00125587133228398*x^2+0{,}0234658568491038*x-9{,}79212125259337$
Gauss-Seidel Eigen	$F(x)=0{,}0378160362811327*x-17{,}4261427512535$
SparseLU	$F(x)=0{,}000621457528516025*x-0{,}226376575917264$

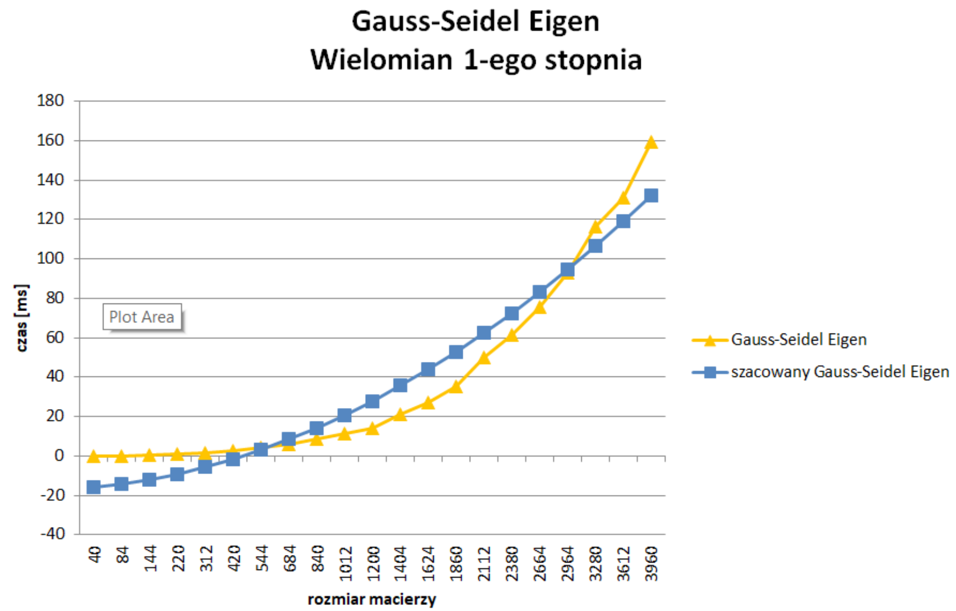
3.3. Poprawność rozwiązania

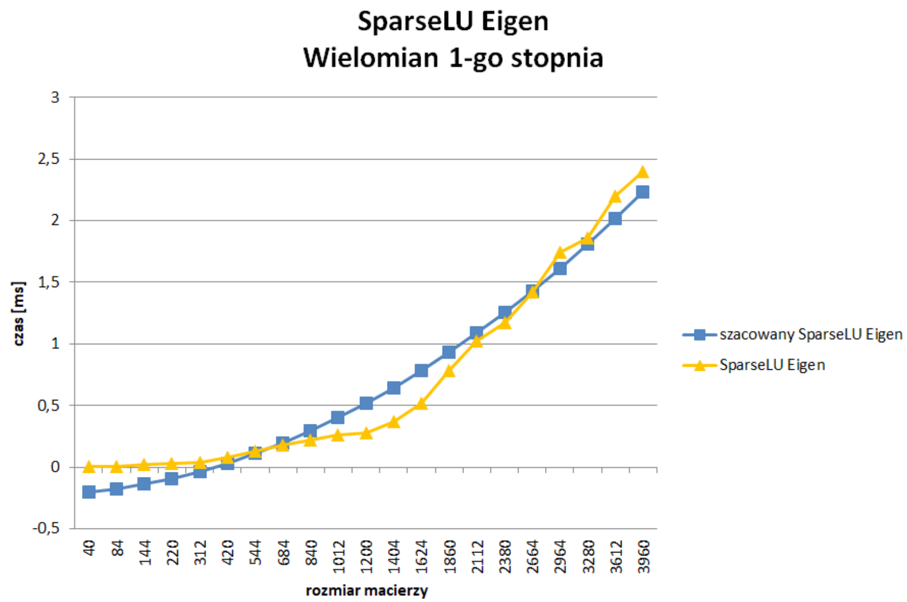
Wszystkie zadane funkcje nałożono na odpowiedniki swoich wyników na kolejnych, poniższych wykresach. Takie nałożenie na siebie funkcji pozwala na łatwe zobrazowanie dobrego odwzorowania (dokładności) wyliczonych funkcji aproksymacyjnych. Można przyjąć, że wartości wyliczone przez funkcje aproksymacyjne dość dobrze przybliżają nas do oczekiwanego czasu.











3.4. Szacowanie czasu działania metod; ekstrapolacja (zadany rozmiar układu: 100000 równań)

Gauss partial	**Gauss partial (zoptymalizowany)**	**Gauss-Seidel**	**Gauss-Seidel Eigen**	**SparseLU**
1 021 138 596 ms = 1 021 138,596 s = 17 018,9766 min = 283,64961 h	3 738 629 ms = 3 738,629 s = 62,310483 min = 1,038508 h	12 561 050 ms = 12 561.05 s = 209,350833 min =3,489181 h	3764 ms = 3,764 s	62 ms = 0,062 s
~30 minut dla 5 600	~30 minut dla 22 000	~30 minut dla 12 000		

3.4.1. Próba obliczenia problemu o wyznaczonym rozmiarze najszybszą metodą

Metoda	Czas oczekiwany	Czas wyliczony	Różnica (niedoszacowanie)
SparseLU	0,062 s	0,213 s	0,213 s – 0,062 s = 0,151 s $\frac{0{,}151\,s}{0{,}213\,s} \approx 0{,}71 = 71\%$

4. Szczegóły wykonania zadania

Podział obowiązków	
Barzowska Monika	Bienias Jan
Implementacja Gaussa-Seidela w C++	Implementacja funkcji aproksymacyjnej
Obróbka wyników, sprawozdanie	Testy i generowanie wyników

5. Źródła

1. Recktenwald G., *Stopping Criteria for Iterative Solution Methods* [online], [dostęp: 6.01.2018]
Dostępny w Internecie: http://web.cecs.pdx.edu/~gerry/class/ME448/notes_2012/pdf/stoppingCriteria.pdf
2. Ciskowski K., *Wykłady uczelniane* [online], [dostęp: 7.01.2018]
Dostępny w Internecie: https://eti.pg.edu.pl/documents/176593/26763380/Wykl_AlgorOblicz_3.pdf
3. https://eigen.tuxfamily.org/dox/group__TutorialSparse.html