3.5 Rozwiązywanie układu równań liniowych metodą Doolittle'a (LU:L(i,i)=1)

Oliwia Masian 127324, Informatyka, Wydział Informatyki, II rok,
Elementy Analizy Numerycznej

1. Zastosowanie.

Program rozwiązuje układ równań liniowych w postaci

$$Ax = b$$

gdzie A oznacza macierz kwadratową stopnia n oraz $x, b \in R^n$, metodą Doolittle'a.

2. Opis metody.

Rozkład LU.

W wielu zagadnieniach numerycznych dotyczących macierzy kwadratowych A, poszukujemy takich macierzy trójkątnych L i U, że $A = L \cdot U$, gdzie L jest macierzą dolnotrójkątną z jedynkami na diagonali:

$$L = \begin{matrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \dots & \dots \\ \dots & \dots & \dots & 0 \\ l_{n1} & \dots & l_{n\,n-1} & 1 \end{matrix}$$

z kolei U macierzą górnotrójkątną:

$$U = \begin{matrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & \dots & \dots \\ \dots & \dots & \dots & u_{n-1\,n} \\ 0 & \dots & 0 & u_{nn} \end{matrix}$$

Metoda Doolittle'a umożliwia wyznaczenie takiego rozkładu, traktując równość $A = LU$ jako układ n^2równań z n^2niewiadomymi l_{ij} dla $i > j$ i niewiadomymi u_{ij} dla $i \leq j$. Równania te wygodnie jest rozwiązywać na przemian wierszami i kolumnami, to znaczy raz wyznacza się wiersz macierzy U, raz kolumnę macierzy L.

W przypadku ogólnym elementy l_{ij} oraz u_{ij} obliczamy następująco:

dla $i = 1, 2, \dots n$ obliczamy kolejno:

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj}\,, j = i, i+1, \dots n$$

$$l_{ji} = \frac{1}{u_{ii}} \left(a_{ji} - \sum_{k=1}^{i-1} l_{jk} u_{ki} \right), j = i+1, i+2, \dots n$$

Liczba potrzebnych mnożeń do wykonania rozkładu wynosi $\frac{1}{3}n^3 - \frac{1}{3}n$, a dodawań $\frac{1}{3}n^3 - \frac{1}{2}n^2 + \frac{1}{6}n.$

Rozwiązanie układu

Po otrzymaniu rozkładu LU rozwiązujemy układ

$$L \cdot y = b$$

a następnie

$$U \cdot x = y$$

W celu zwiększenia niezawodności metody (zabezpieczenie przed dzieleniem przez 0, gdy $u_{ii} = 0$), metoda ta została rozszerzona o częściowy wybór elementu podstawowego.

Wybór elementu podstawowego (wybór w kolumnie)

Teoretycznie za element podstawowy można wybrać każdy element na głównej przekątnej $a_{r1} \neq 0$ (dla pierwszego wiersza). Z uwagi jednak na poprawność rozwiązania (zmnniejszenie błędów zaokrągleń) wybiera się

$$|a_{r1}| = \max_{i=1,2,..,n} |a_{i1}|$$

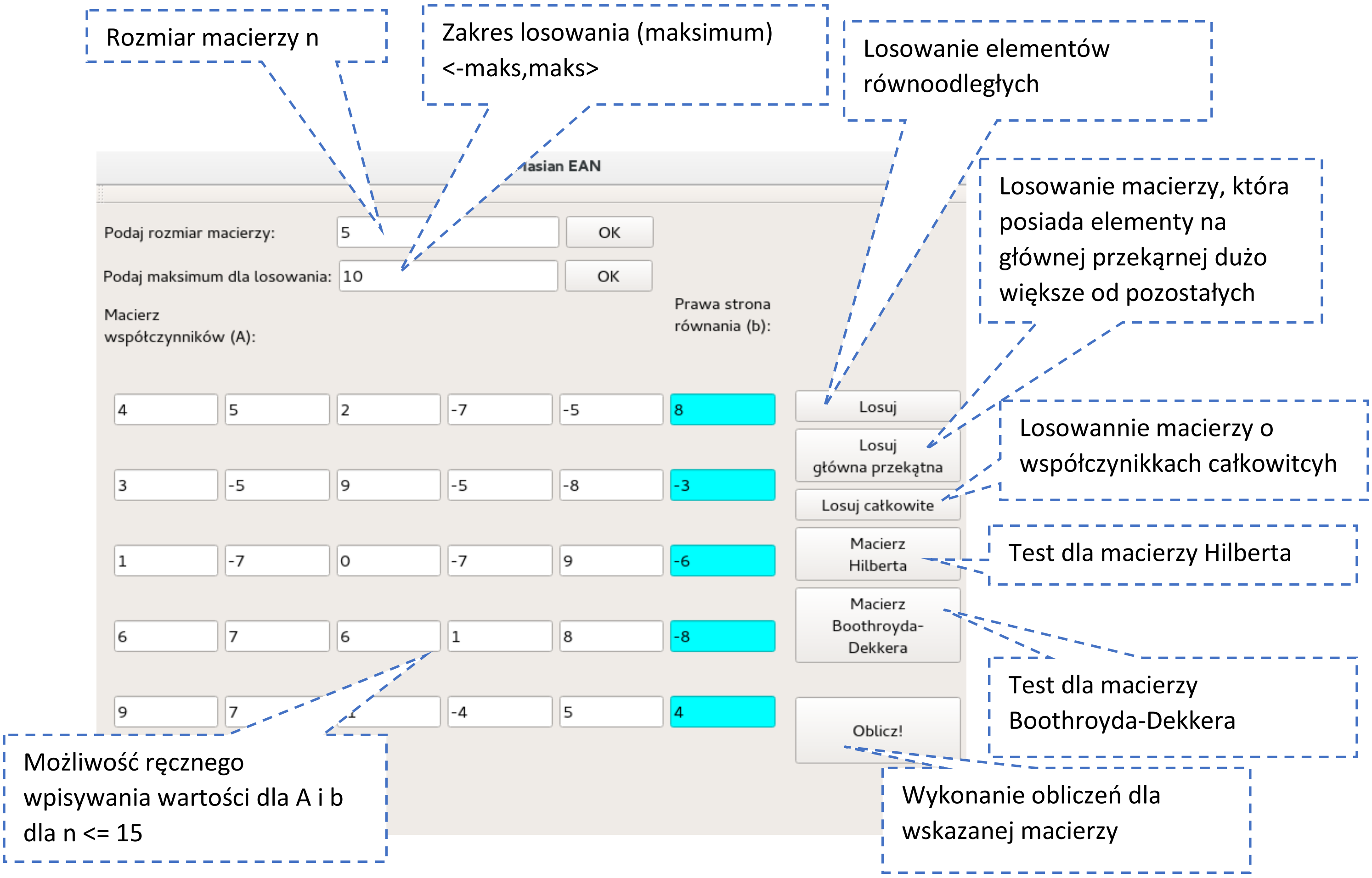
Następnie przestawia się wiersz pierwszy z i-tym, zapamiętuje się tę operację w macierzy permutacji. W przypadku rozkładu LU konieczna jest też zmiana w macierzy L, którą można opisać następującym pseudokodem:

```
//i to numer wiersza, który przekształcamy
for(int a = 0; a < i; a++)
{
        long double zamien3 = L[i][a]; //pomocnicza zmienna
        L[i][a] = L[imax][a];  //zamieniamy poszczegolne elementy
        L[imax][a] = zamien3;
}
```

Idea polega na niezamienianiu całego wiersza, a jedynie tych elementów, które zostały wyznaczone przez algorytm do momentu wykonywania zamiany.

Analogiczne postępowanie prowadzi się dla pozostałych wierszy.

3. Obsługa programu.



4. Typy parametrów.

long double: A, x, b w arytmetyce zwykłej

Interval <long double>: A, x, b w arytmetyce przedziałowej

int: n

5. Wykorzystane biblioteki.

Projekt został napisany w języku C++, z wykorzystaniem następujących bibliotek:

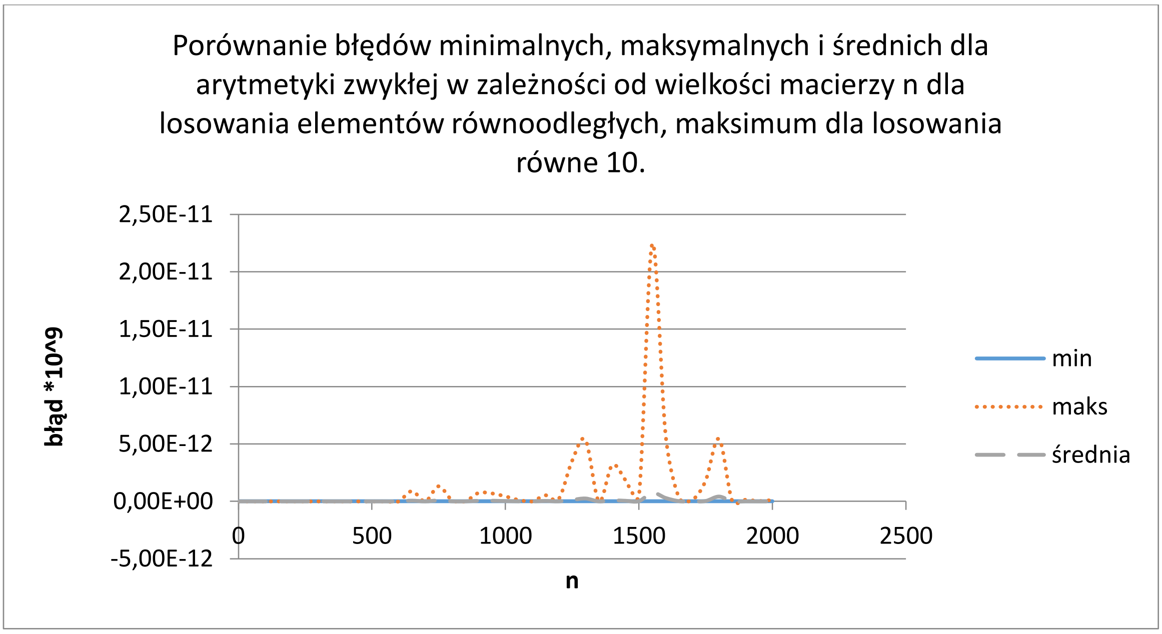
- #include<mpfr.h> (GNU MPFR library, GNU GMP library, potrzebne przy konwersji liczby z ciągu znaków na liczbę w arytmetyce przedziałowej, umożliwiają obliczenia w arytmetyce zmiennoprzecinkowej z odpowiednim zaokrąglaniem)
- #include<boost/numeric/interval.hpp> (z pakietu boost, arytmetyka przedziałowa)
- #include <boost/lexical_cast.hpp> (potrzebne przy konwersji liczby z ciągu znaków na liczbę w arytmetyce przedziałowej).

Interfejs użytkownika powstał przy pomocy środowiska Qt Creator (Qt 5.8.0 GCC 64bit), projekt został skompilowany przy użyciu kompilatora GCC, w środowisku linuksowym (openSUSE 64bit).

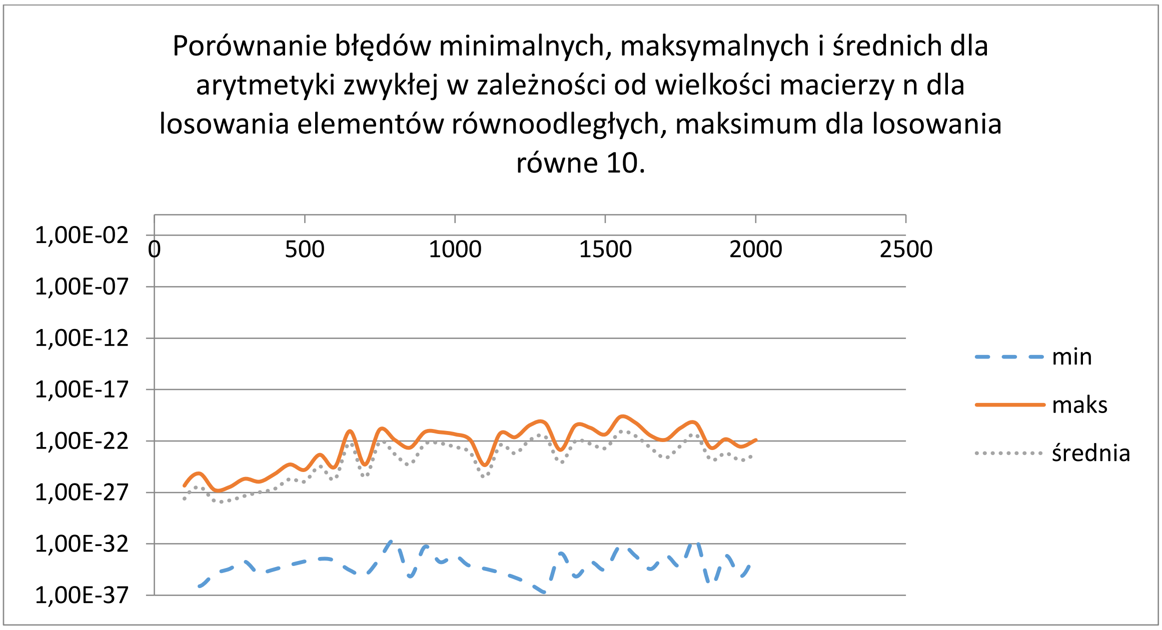
6. Wyniki testów.

W celu umożliwienia wizualizacji wyników w programie Excel, błędy zostały przemnożone przez odpowiednie współczynniki, dla których możliwe jest wykonanie wykresów.

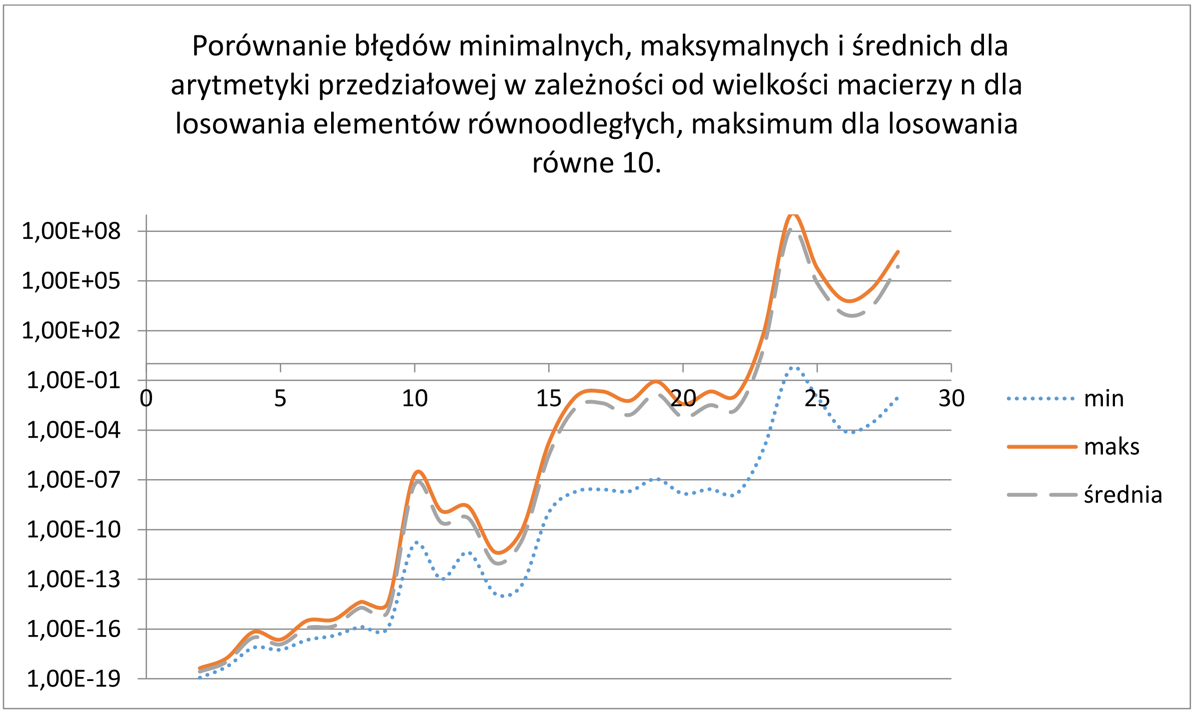
Dla arytmetyki zwykłej błąd został obliczony jako różnica między b -wejściowym z b^*-obliczonym (na podstawie otrzymanego x) podniesiona do kwadratu. Dla arytmetyki zwykłej błąd utożsamiany jest z szerokością przedziału.



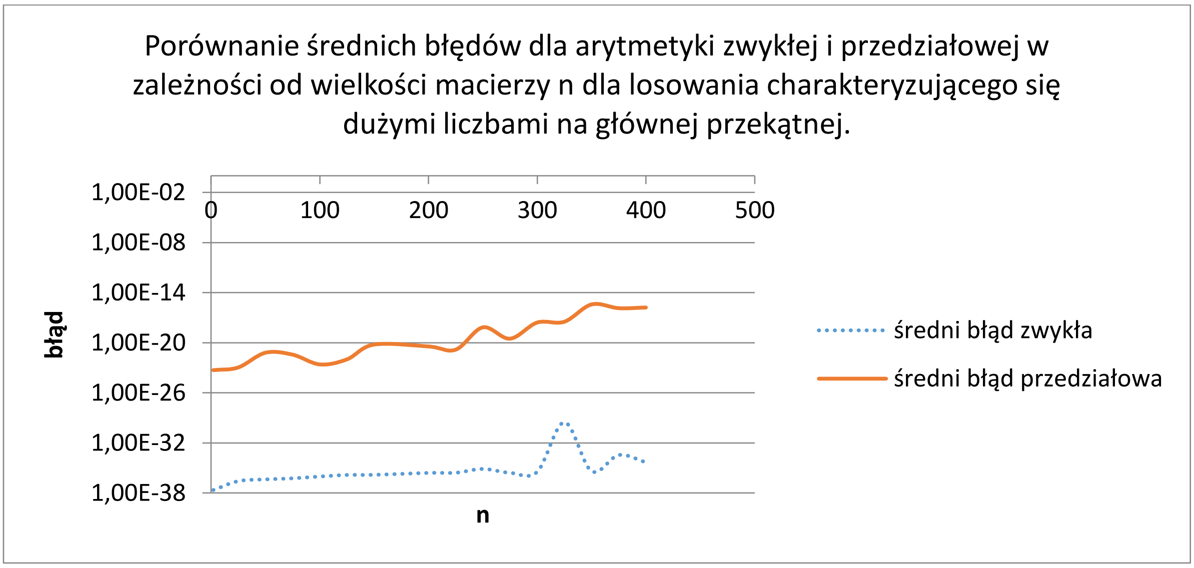
Ten sam wykres w skali logarytmicznej:



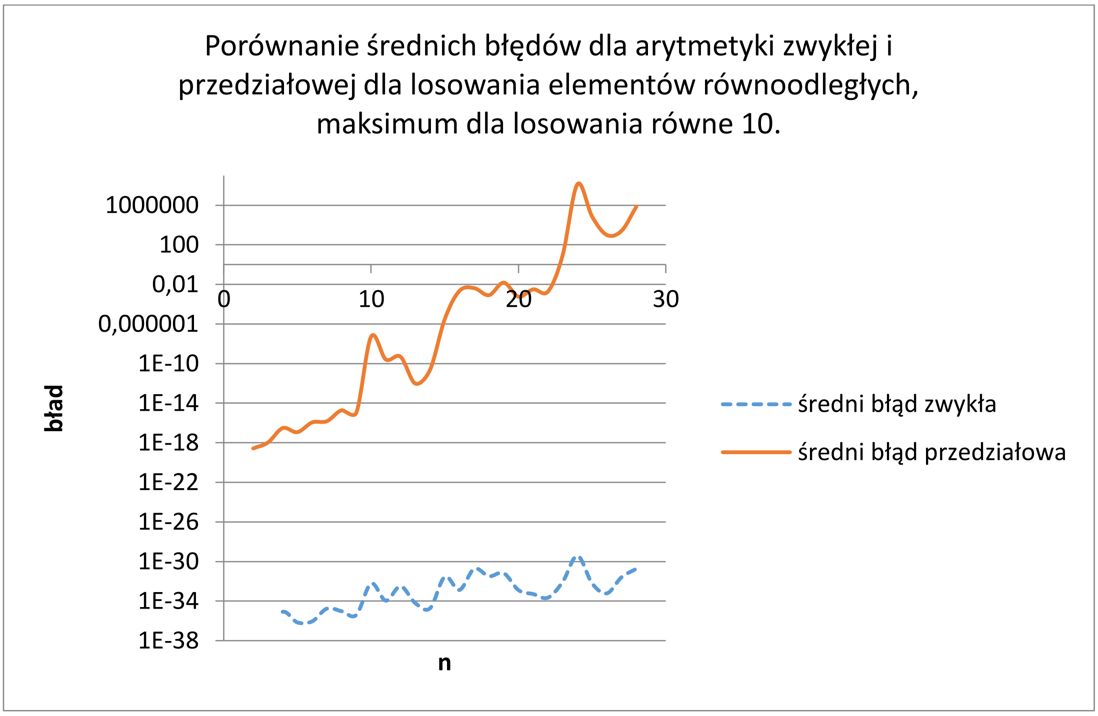
Skala liniowa- widzimy duże wahania dla maksymalnego błędu.



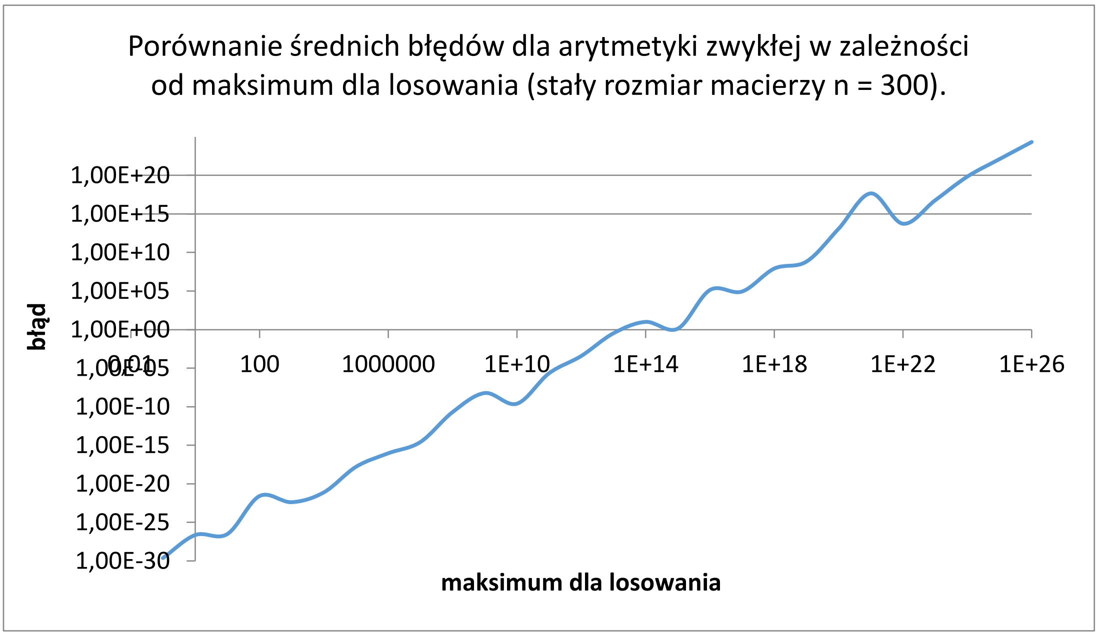
Skala logarytmiczna- wraz ze wzrostem wymiaru macierzy obserwujemy wzrost wszystkich rodzajów błędów. Maksymalny błąd zbliżony do średniego.



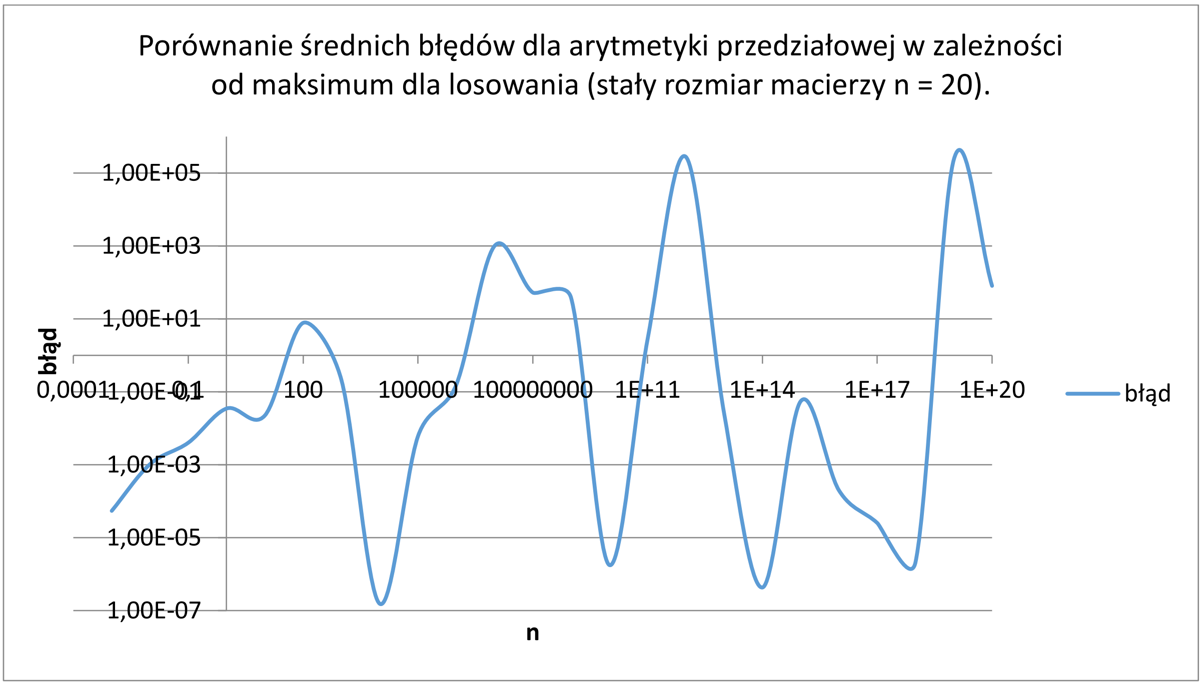
Obserwujemy różnice w błędach o kilkadziesiąt rzędów jedności, niemniej jednak istnieje wspólna prawidłowość- błędy w obydwu przypadkach średnio rosną przy zwiększaniu wymiaru macierzy.



Dla zwykłego losowania możliwe było przeprowadzenie testów dla n < 30.



Wraz ze wzrostem maksimum losowania, średni błąd zwiększa się.



Wykres przypomina szum- zbyt duże wahania błędów uniemożliwiają wyciągnięcie wniosków.

7. Wnioski.

- Rzędy wielkości błędów w arytmetyce zwykłej są dużo mniejsze od tych w arytmetyce przedziałowej. Należy jednak wziąć pod uwagę fakt, że w arytmetyce zwykłej błędy także są obarczone pewnym błędem przybliżenia. Tylko arytmetyka przedziałowa pozwala na wskazanie dokładnego przedziału, w jakim zawiera się wynik.
- Błędy w arytmetyce przedziałowej bardzo szybko rosną wraz ze wzrostem wymiaru macierzy, dla arytmetyki zwykłej odbywa się to dużo wolniej.
- Maksimum dla losowania wpływa na błędy. Widoczne jest to w zwykłej arytmetyce (niestety w arytmetyce przedziałowej nie jest możliwe przeprowadzenie takiej obserwacji ze względu na zbyt dużą wariancję błędów). Wraz ze wzrostem maksimum dla losowania, rośnie błąd rozwiązania. Wynika to z tego, że odległości pomiędzy kolejnymi liczbami w reprezentacji maszynowej na komputerze są większe dla dużych liczb.
- W przypadku losowania macierzy, której elementy na głównej przekątnej są dużo większe od pozostałych (zastosowano wzór (A[i][j]*100 + 100)^2, gdzie A[i][j] jest losową wartością ograniczoną maksimum dla losowania). Dla takiej macierzy możliwe są obliczenia w arytmetyce przedziałowej nawet dla n>500, czego przy standardowym losowaniu nie udaje się zrealizować.

8. Źródła.

- Fortuna, Macukow, Wąsowski, Metody numeryczne, WNT, Warszawa
- Andrzej Marciniak, Dorota Gregulec i Jan Kaczmarek, Podstawowe procedury numeryczne w języku Turbo Pascal, 1997