Sprawozdanie numeryczne

Cel badania

Celem badania było sprawdzenie efektywności rozwiązywania układu równań metodą faktoryzacji LU dla macierzy 4-diagonalnej oraz obliczenie wyznacznika tej macierzy dla różnych wymiarów N.

Metoda badawcza

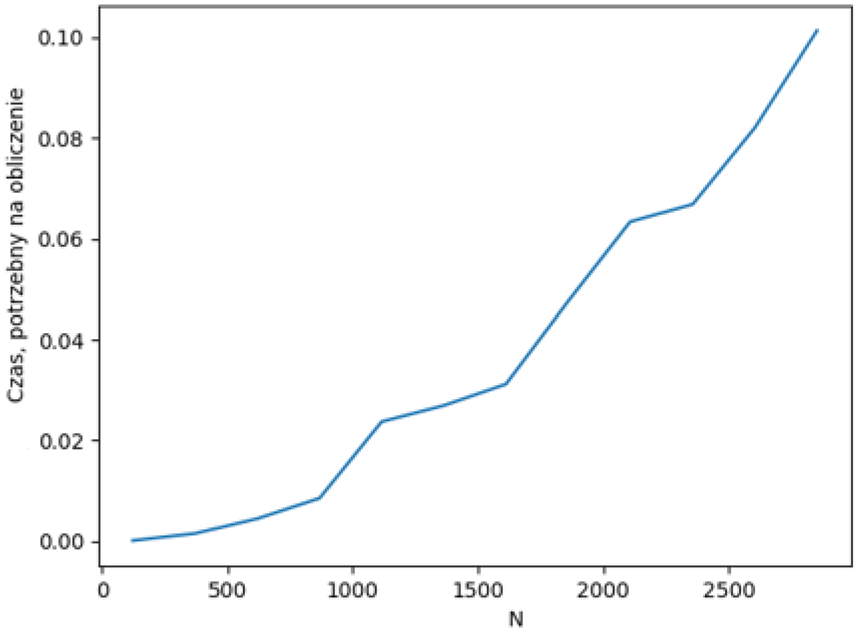
1. Zaimplementowano funkcje forward_substitutionn i backward_substitution, które odpowiadały za rozwiązanie układu równań metodą faktoryzacji LU.
2. Macierz była zaimplementowana jako słownik ze wskaźnikami dla efektywnego wykorzystania pamięci komputera. Słownik ten posiada jedynie wartości 4-ech diagonali.
3. Dla sprawdzania wyniku była stworzona macierz w standardowy sposób oraz wykonano wszystie odpowiednie operacje z wyznaczeniem straconego czasu
4. Funckja LU dzieli macierz A na macierze diagonalne L oraz U w sposób efektywny rozważając tylko przypadki zmiany elementów diagonalnych
5. Przeprowadzono serię pomiarów czasu dla różnych wartości N, poczynając od 124 do 4464 oraz dla różnych ilości powtórzeń pomiary, od 10 do 200.
6. Dokonano analizy wyznacznika i czasu rozwiązania układu równań, zarówno za pomocą funkcji bibliotecznych, jak i za pomocą własnej implementacji.

Wyniki

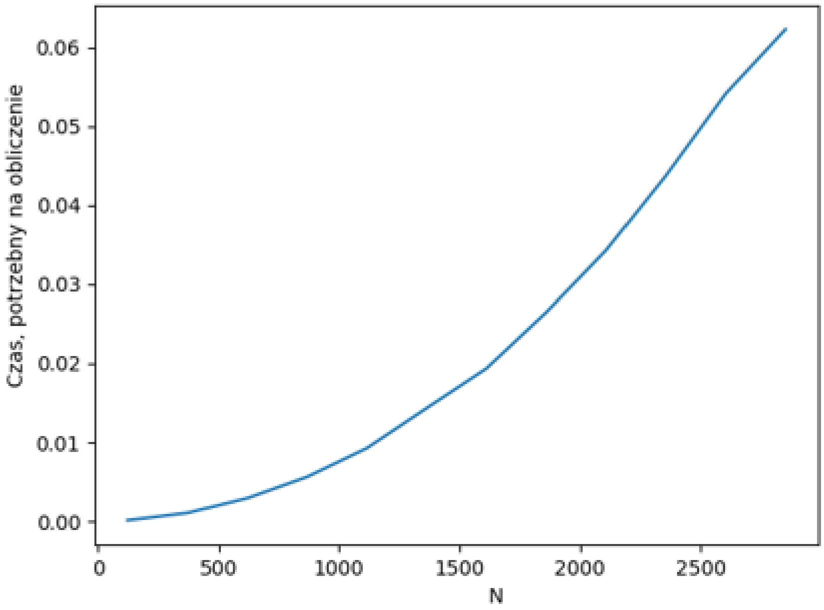
Odpowiednia metoda(LU) została wybrana przez swoją szybkość i dostasowano do niej korzystanie ze słownika, co pozwala nam znacznie zmniejszyć złożoność pamięciową. Metoda biblioteczna jest w znacznym stopniu wolniejsza od wybranej, na co wskazują badania czasu oraz potrzebuje tworzenia macierzy NxN, co zajmuje znaczną ilość pamięci. Metoda LUP nie została wybrana, bo typ macierzy jest dobrany w taki sposób, że i tak wiemy, że nie będziemy dzielić przez zero. Faktoryzacja Thomasa dobrze działa z macierzami diagonalnymi symetrycznymi, nasza macierz jednak nie jest symetryczna. Nie używaliśmy do tego faktoryzacji Cholesky'ego, bo liczenie pierwiastku spowalnia algorytm i nasza macierz nie jest symetryczna. Faktoryzacja QR też pokazuje gorsze wyniki czasowe, dlatego została wybrana metoda LU Doolittle'a.

Poniższe wykresy odpowiadają realizacji kodu z używaniem słownika. Cały algorytm jest realizowany w O(n), jeandk widać jest, że wykres odpowiada funkcji kwadratowej, co świadczy, że złożoność algorytmu jest większa od O(n). Jest to związane z tym, że w Pythonie dostęp do wartości pod kluczem nie jest O(1), dlatego korzystania ze słownika spowalnia program. Dla dużego N ta róznica jest jaknajberdziej widoczna.

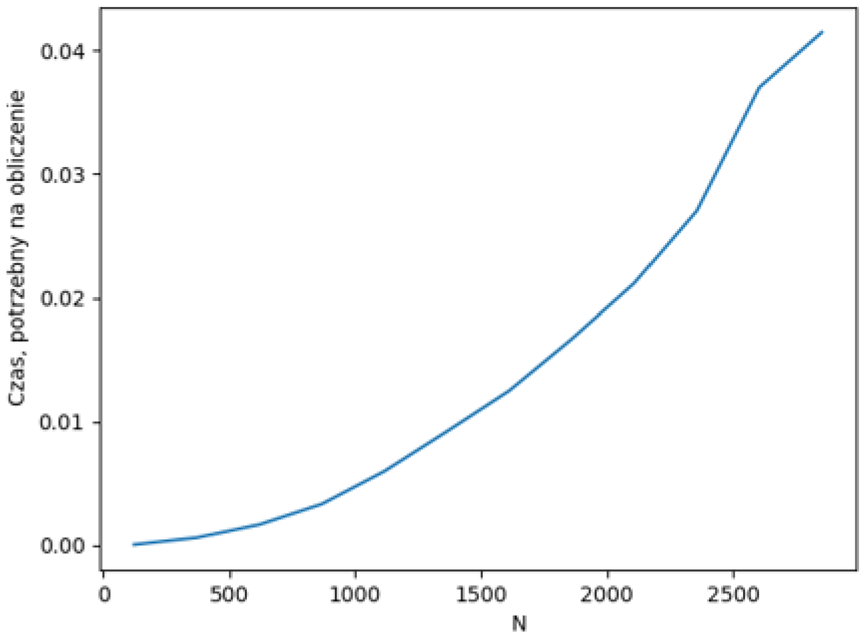
Pierwsze pomiary czasu dla rozwiązania układu równań metodą LU wyniosły około 0.00014(N=124) sekundy dla N w przedziałach [124, 2976] przy dziesięciu próbach wymiaru. Metoda biblioteczna trwała 0.0039 sekund. Dla N = 2976, pomiar czasu LU wynosi 0.1



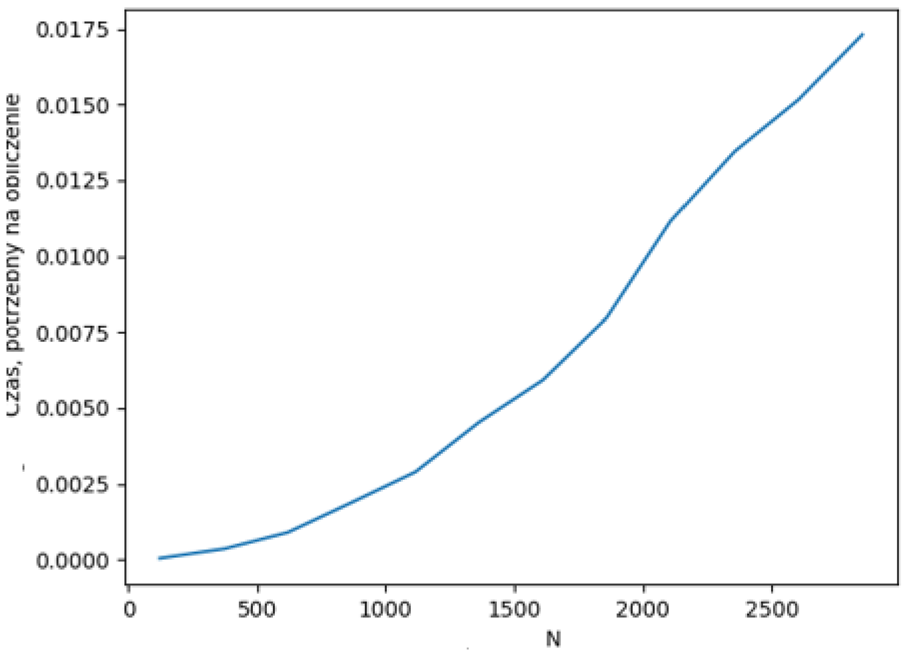
Drugie pomiary czasu dla rozwiązania układu równań metodą LU wyniosły około 0.00013(N=124) sekundy dla N w przedziałach [124, 2976] przy trzydziestu próbach wymiaru. Metoda biblioteczna trwała 0.0040 sekund. Dla N = 2976, pomiar czasu LU wynosi 0.06225



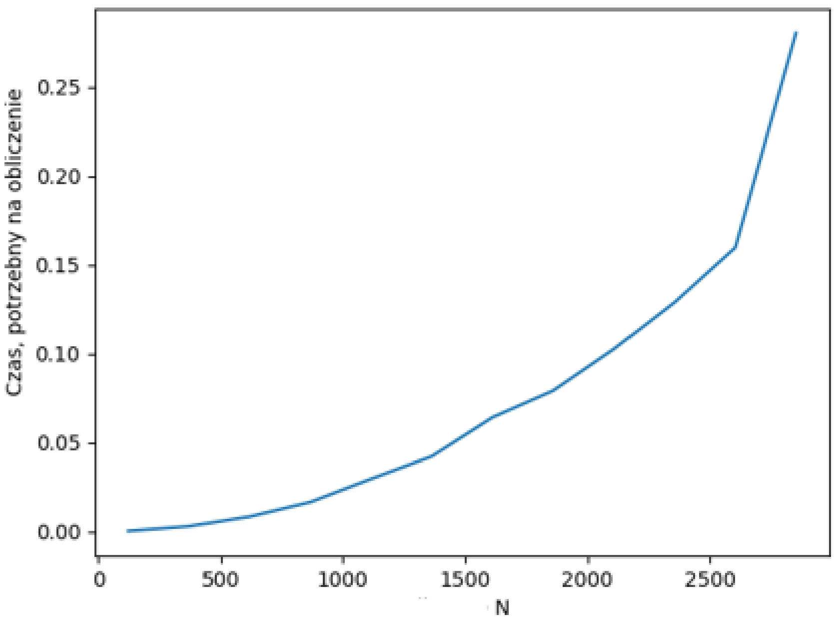
Trzecie pomiary czasu dla rozwiązania układu równań metodą LU wyniosły około 5.97e-05 (N=124) sekundy dla N w przedziałach [124, 2976] przy 50 próbach wymiaru. Metoda biblioteczna trwała 0.0045 sekund. Dla N = 2976, pomiar czasu LU wynosi 7.99e-05



Czwarte pomiary czasu dla rozwiązania układu równań metodą LU wyniosły około 4.0e-05(N=124) sekundy dla N w przedziałach [124, 2976] przy stu próbach wymiaru. Metoda biblioteczna trwała 0.0043 sekund. Dla N = 2976, pomiar czasu LU wynosi 0.0173

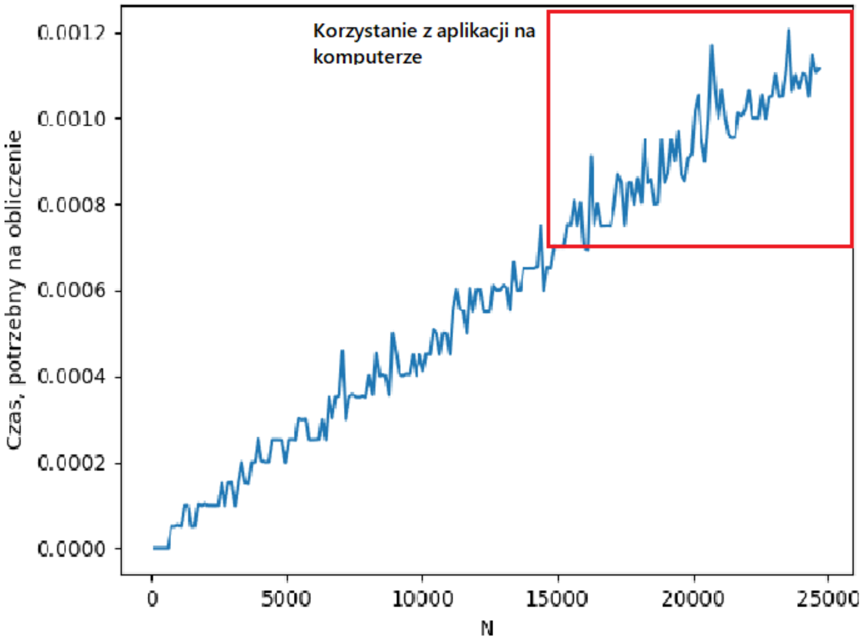


Piąte pomiary czasu dla rozwiązania układu równań metodą LU wyniosły około 0.00039(N=124) sekundy dla N w przedziałach [124, 2976] przy 10 próbach wymiaru. Metoda biblioteczna trwała 0.0040 sekund. Dla N = 2976, pomiar czasu LU wynosi 0.28

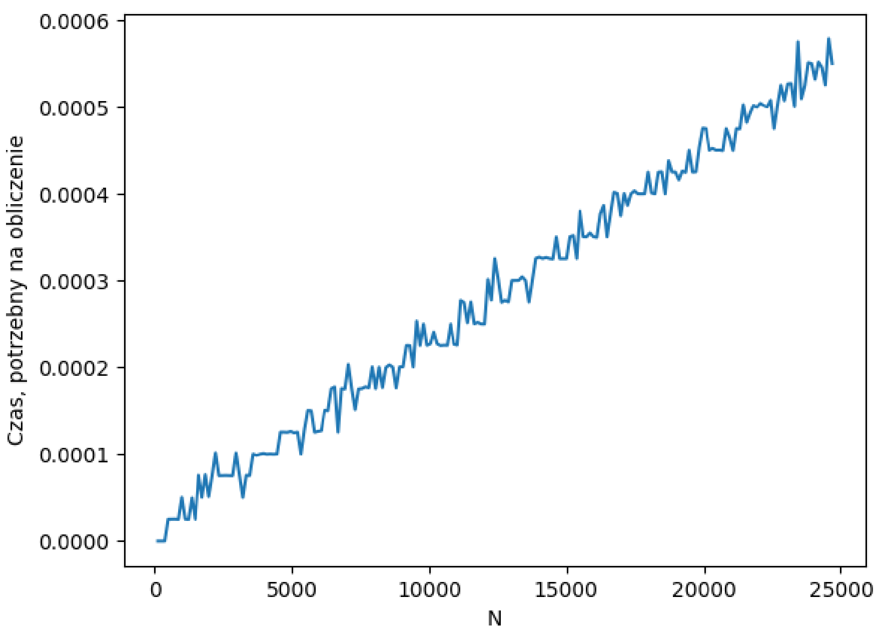


Wartość wyznacznika macierzy A (dla N=124), obliczona za pomocą metody bibliotecznej, wyniosła około 6141973498.857843. Natomiast dla metody własnej implementacji wynik wyniósł około 6141973498.857843, co znaczy, że wyznacznik obliczony jest w sposób poprawny.

Następujące wykresy odpowiadają realizacji programu przy użyciu tablicy wymiarów 4xN. Algorytm nie był zmieniony, zmiany zostały wprowadzone jedynie dla typu danych.

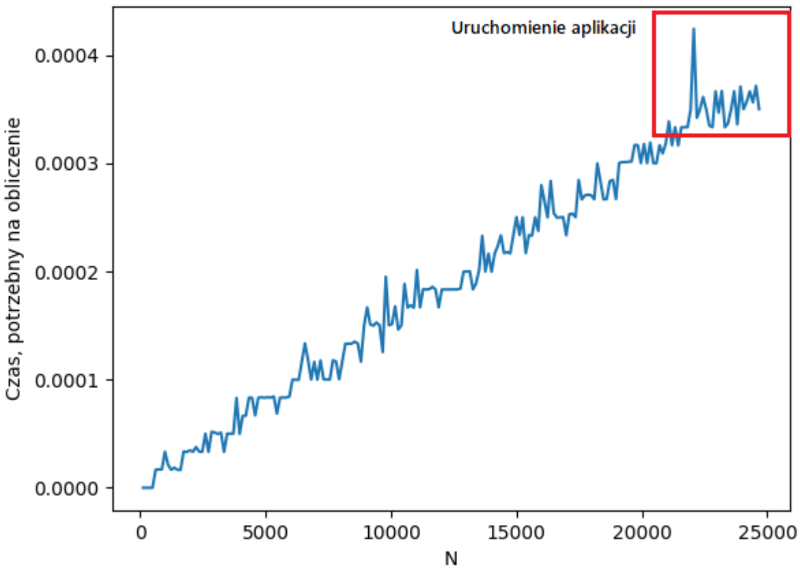


Pierwszy test był przeprowadzony dla N w zakresie [124, 25000], przy uśrednieniu wyniku pomiaru czasu z 20 prób. Wynik nie ma dużej precyzyjności, oraz w drugiej połowie programu widoczne są duże skoki z powodu uruchomienia pewnej ilości innych programów, co wpływały na wynik w znacznym stopniu. Czas dla rozwiązywania metodą biblioteczną wynosi 0.0045 dla N = 124 , własna metoda: 0.00005 sekund, dla N = 25000, pomiar czasu pokazał 0.0012 sekund.



Drugi test przeprowadzony jest dla 40 prób przy wyłączonych programach na komputerze. Widać stąd, że wcześniej programy miały duży wpływ na stabilność wyniku, który również można zminimalizować za pomocą zwiększenia ilości prób eksperymentu numerycznego.

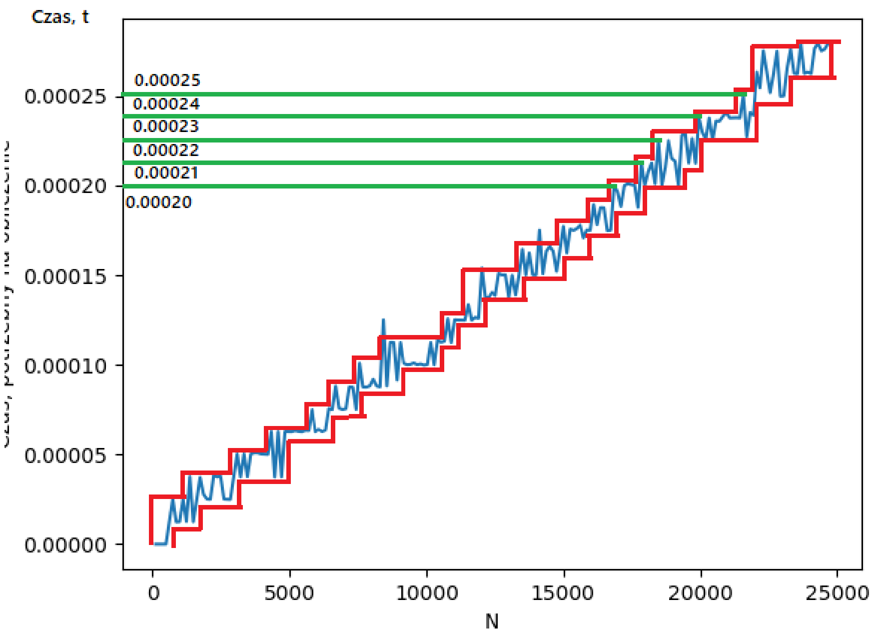
Dla N=124, czas wynosi 0.0043 sekundy, dla naszej metody 0.00003 sekundy. Dla N=25000 układ jest rozwiązywany za 0.0006 sekundy.



Trzeci test dla 60 prób pokazał większą precyzyjność wyniku i tutaj już wyraźnie widać „schodki", o których mowa będzie dalej. W końcu uruchomiłem przez przypadek menedżer procesów, co spowodowało wydłużenie czasu rozwiązywania równania, co wskazuje, że nawet dla dużej ilosci prób eksperymentu, działanie innych programów moze wpływać na wynik pomiaru czasu.

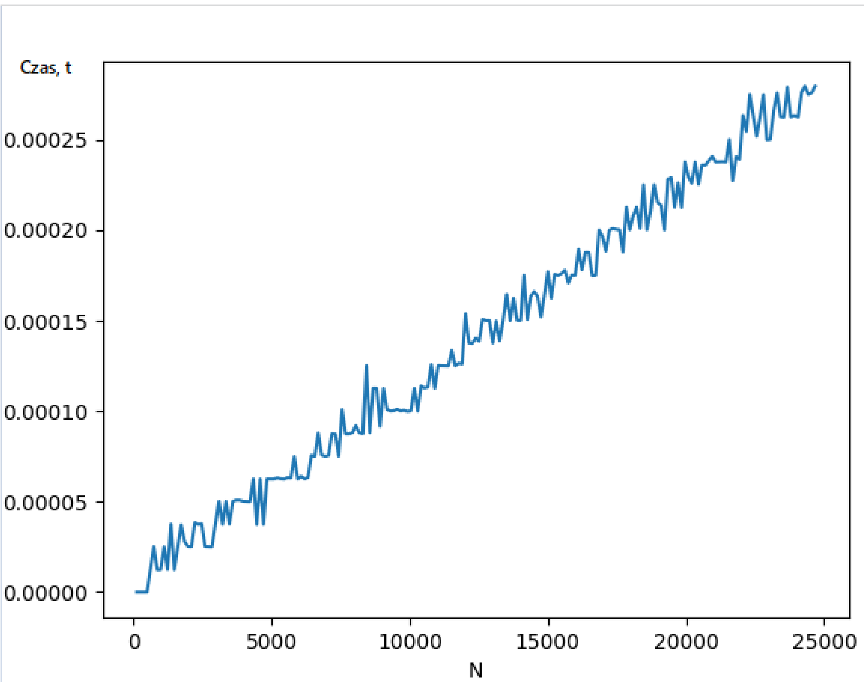
N=124 => t1(czas dla metudy bibliotecznej) = 0.00042, t2(czas dla metody własnej) = 0.00002

N= 25000 => t2 = 0.00035



Schodki

Schodkowy wzrost czasu jest związany z reprezentacją wartości na wykresach dla funkcji plot() z biblioteki matplotlib.pyplot. Ta biblioteka dzieli przestrzeń między wartościami t na 5 części, przez co przy wypisywaniu czas jest zaokrąglany do najbliższej z tych części(wartości na osi t).



Grafik z wykresu powyżej. Prób jest 80, precyzyjność wyniku jest wyższa.

N = 124 => t1(biblioteczny) = 0.00042, t2(własny) = 0.00001

N = 25000 => t2 = 0.00028.

Macierz bibblioteczna dla N = 25000 liczona nie jest z powodu braku takiej ilości pamięci i mocy obliczeniowej komputera, na którym zostały przeprowadzone eksperymenty.

W każdym z eksperymentów wyznacznik oraz wynik są poprawne i wynoszą:

Wyznacznik = 6141973498.857843

Wektor **wynik** dla N =124

[0.448700827728733, 1.4132732869357947, 2.1348778535462736, 2.8690132654396248, 3.5914885705595205, 4.311604959915503, 5.029827173723323, 5.747011462584994, 6.463503693914558, 7.179525964548697, 7.895212596895915, 8.610651859797315, 9.325903619364162, 10.041009954626537, 10.756001271894783, 11.470900080737994, 12.185723394049369, 12.900484305313817, 13.615193053266005, 14.329857758065131, 15.044484941235352, 15.759079899902147, 16.473646980766127, 17.18818978377055, 17.902711315621058, 18.617214106977666, 19.331700302956037, 20.046171733762908, 20.76062997036838, 21.47507636878333, 22.189512105570603, 22.903938206548368, 23.6183555701598, 24.332764986629712, 25.047167153767735, 25.761562690082965, 26.475952145728595, 27.190336011683936, 27.904714727496145, 28.61908868783829, 29.33345824808956, 30.047823729103516, 30.762185421298863, 31.476543588182352, 32.19089846939369, 32.90525028334641, 33.61959922952588, 34.33394549049516, 35.048289233651346, 35.762630612767495, 36.4769697693505, 37.19130683383959, 37.90564192666726, 38.6199751592003, 39.33430663457682, 40.04863644845216, 40.762964689665075, 41.47729144083417, 42.19161677889273, 42.905940775569235, 43.62026349782013, 44.33458500822004, 45.04890536531427, 45.763224623938, 46.47754283550541, 47.19186004827236, 47.90617630757509, 48.62049165604775, 49.334806133820685, 50.04911977870149, 50.76343262634067, 51.47774471038327, 52.192056062607854, 52.9063667130542, 53.62067669014054, 54.33498602077156, 55.04929473043772, 55.76360284330703, 56.47791038230969, 57.192217369216245, 57.906523824710035, 58.62082976845425, 59.335135219153926, 60.049440194613666, 60.763744711791205, 61.47804878684707, 62.192352435190934, 62.90665567152471, 63.62095850988271, 64.3352609636691, 65.04956304569288, 65.76386476820053, 66.47816614290662, 67.19246718102224, 67.90676789328191, 68.62106828996849, 69.33536838093679, 70.0496681756356, 70.76396768312829, 71.47826691211242, 72.19256587093781, 72.90686456762393, 73.62116300987596, 74.33546120510012, 75.04975916041795, 75.76405688267984, 76.47835437847795, 77.19265165415811, 77.90694871583132, 78.62124556938441, 79.33554222049035, 80.04983867461782, 80.76413493704023, 81.47843101284448, 82.19272690693916, 82.90702262406211, 83.62131816878798, 84.33561354553508, 85.04990875857204, 85.76420381203626, 86.47849870460276, 87.19279247126808, 87.90778524137869, 88.68203579310355]

Analiza

Wpływ uśrednienia wyników

Uśrednianie wyników pomiarów okazało się ważnym aspektem. Wykonując pomiar czasu wielokrotnie i obliczając średnią uzyskanych wartości, eliminujemy wpływ czynników losowych na ostateczny wynik. Zauważyliśmy, że pojedyncze pomiary czasu mogą być obarczone znaczną zmiennością, co wynika z działania systemu operacyjnego i innych czynników zewnętrznych.

Wpływ innych programów

Inne programy mają znaczny wpływ na czas rozwiązywania równania i nie zawsze jest możliwe jego wyeliminowanie za pomocą zwiększenia ilości prób w związku z tym, że wszystkie takie próby są przeprowadzone w czasie działania tego programu i wszystkie próby będą mieli zwiększony czas.

Wpływ rozmiaru macierzy

Nasze badanie wyraźnie pokazało, że czas obliczeń rośnie w sposób liniowy wraz ze zwiększaniem rozmiaru macierzy N. Oznacza to, że złożoność obliczeniowa metody faktoryzacji LU dla macierzy 4-diagonalnych jest liniowa. Dla mniejszych wartości N czas obliczeń jest relatywnie krótki, ale w miarę wzrostu N staje się bardziej wymagający obliczeniowo. Jest to charakterystyczne dla tego typu algorytmów i ilustruje, że dla dużych problemów numerycznych, takich jak te związane z analizą wielkich ilości danych, czas obliczeń może być znaczący.

Otrzymane wyniki potwierdzają skuteczność zastosowanej metody faktoryzacji LU do rozwiązywania układów równań liniowych dla macierzy 4-diagonalnych.

Wnioski

1. Metoda faktoryzacji LU jest skutecznym sposobem rozwiązywania układów równań dla macierzy 4-diagonalnych.

2. Złożoność obliczeniowa metody wzrasta wraz ze zwiększaniem rozmiaru macierzy, co może wpływać na czas wykonywania obliczeń.

3. Wartość wyznacznika macierzy A, obliczona za pomocą funkcji bibliotecznych oraz własnej implementacji, była zgodna, co potwierdza poprawność obliczeń.

4. W miarę wzrostu rozmiaru macierzy N, czas obliczeń rośnie liniowo, co może mieć znaczenie przy analizie dużych zestawów danych.

Autor

Maksym Yankovenko