

**SPRAWOZDANIE**

**LABORATORIUM NR 1**

**TEMAT:**

**Rozwiązywanie układów równań liniowych metodami bezpośrednimi**

**PROWADZĄCY:**

Dr inż. Barbara Głut

**AUTOR:**

Małgorzata Olszewska

1. **Cel ćwiczenia.**

Celem ćwiczenia było zapoznanie się z podstawowymi metodami bezpośrednimi rozwiązywania układów równań liniowych oraz pojęciem uwarunkowania zadania polegającego na wyznaczeniu rozwiązania takiego układu.

1. **Przebieg ćwiczenia.**

W ramach pierwszych zajęć laboratoryjnych należało wykonać trzy zadania – każde z nich obejmowało inny typ macierzy.

Pierwszy etap każdego z eksperymentów polegał na przyjęciu określonych wartości wektora rozwiązań i macierzy współczynników oraz obliczeniu na tej podstawie wektora wyrazów wolnych.

Następnie należało wyznaczyć wektor rozwiązań na podstawie znanych wartości wektora wyrazów wolnych i macierzy współczynników oraz porównać wartości otrzymanego wektora rozwiązań z wartościami wzorcowymi przyjętymi w pierwszym etapie ćwiczenia dla różnych precyzji obliczeń.

Ostatnie zadanie dodatkowo wymagało porównania czasu działania oraz zajętości pamięci pomiędzy różnymi metodami rozwiązywania układów równań.

1. **Opracowanie wyników.**

**3.1. Informacje wstępne.**

W każdym z zadań przyjęto, iż wektor wzorcowy **X** jest naprzemienną permutacją liczb: -1 i 1, to jest:



dla *i = 0, 1, …, P-1*. Wartość wektora współczynników **B** obliczana była ze wzoru na postać macierzową układu równań:

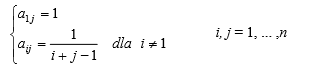
***AX = B****.*

Pomiary wykonywane były jednym rodzaju sprzętu – komputerze o pamięci 8GB DDR3, i systemie Windows 10 Education z domyślnym kompilatorem programu DevC++.

Wszystkie zadania zostały wykonane w języku C++. Do pomiaru czasu zastosowana została biblioteka *chrono* dostępna w standardzie C++11 (flaga kompilacji: *-std=c++11*). Pomiar stanu pamięci został wykonany przy pomocy darmowego narzędzia analizującego *Valgrind*.

**3.2. Zadanie pierwsze.**

Pierwszą z macierzy **A** była zmodyfikowana macierz Hilberta zadana wzorem:



Stosując metodę eliminacji Gaussa wyznaczono wektor rozwiązań **X** dla układu: **AX = B**, a następnie porównano jego wartości z wektorem wzorcowym przyjmując różną precyzję liczb zmiennoprzecinkowych (float – pojedyncza oraz double – podwójna), a także wielkość układu. Aby porównać wektory rozwiązań obliczono normy (euklidesową oraz maksimum) wektora powstałego w wyniku odejmowania wektora wzorcowego i otrzymanego wyniku.

Wyniki przedstawia poniższa tabela:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **P** | **DOUBLE** | | **FLOAT** | | **UWARUNKOWANIE** |
|  |  |  |  |
| 2 | 0 | 0 | 0 | 0 | 18 |
| 3 | 1.369e-15 | 1.110e-15 | 3.345e-06 | 2.682e-06 | 864 |
| 4 | 4.802e-14 | 3.708e-14 | 0.0001021 | 7.951e-05 | 37920 |
| 5 | 2.178e-12 | 1.520e-12 | 0.002302 | 0.001611 | 1.442e+06 |
| 6 | 1.908e-11 | 1.316e-11 | 0.09177 | 0.06286 | 5.634e+07 |
| 7 | 2.180e-09 | 1.512e-09 | 2.478 | 1.718 | 2.232e+09 |
| 8 | 5.207e-08 | 3.447e-08 | 8.805 | 5.860 | 8.155e+10 |
| 9 | 2.062e-06 | 1.274e-06 | 12.950 | 8.604 | 2.843e+12 |
| 10 | 5.388e-05 | 3.418e-05 | 27.410 | 17.150 | 1.090e+14 |
| 11 | 0.0007215 | 0.0004502 | - | - | 3.957e+15 |
| 12 | 0.003235 | 0.001929 | - | - | 1.426e+17 |
| 13 | 0.7726 | 0.4563 | - | - | 4.647e+18 |
| 14 | 1.085 | 0.6394 | - | - | 5.127e+18 |
| 15 | 1.858 | 1.155 | - | - | 3.162e+18 |
| 16 | 1.654 | 1.028 | - | - | 3.346e+18 |
| 17 | 2.135 | 1.284 | - | - | 9.213e+18 |
| 18 | 5.291 | 3.132 | - | - | 1.275e+20 |
| 19 | 2.867 | 1.493 | - | - | 4.212e+19 |
| 20 | 2.740 | 1.578 | - | - | 8.144e+19 |

**Wnioski:**

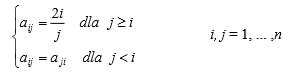
Zauważmy ze mamy niekorzystne uwarunkowanie danego układu. Dla macierzy Hilberta wyraża się ono zależnością: *c∙e3,5N*, zatem bardzo szybko (wykładniczo) rośnie. Podobnie jest i w tym wypadku – dla *P=8* wskaźnik uwarunkowania zadania jest rzędu 1010. Dla wartości P, dla których błędy obliczeniowe są duże (*P>12*), wartości wskaźnika uwarunkowania macierzy przestają zachowywać tendencję wykładniczego wzrostu – dla dwóch ostatnich pomiarów wynik jest mniejszy, niż dla *P=18*. Prawdopodobnie jest to spowodowane faktem niewłaściwego sposoby obliczania wskaźnika uwarunkowania – program odwracający macierz współczynników korzysta z algorytmu Gaussa, który działa z dużą porcją błędu, co potwierdzają normy wektorowe zebrane w tabeli. Zatem dla większych wartości *N* zastosowany algorytm zwraca błędne wyniki, pomimo tego, iż rozmiary macierzy są bardzo małe (nie większe od 20).

Przez zwiększenie dokładności arytmetyki możemy poradzić sobie z złym wskaźnikiem uwarunkowania zadania. Dla *P=10* i pojedynczej precyzji norma maksimum wektora różnic przekracza 17, co jest oczywiście błędem zupełnie nieakceptowalnym (przypomnieć należy, iż wektor wzorcowy składał się wyłącznie z liczb: -1 i 1, zatem otrzymana wartość jest kilkunastokrotnie za duża!). Dla arytmetyki podwójnej precyzji oraz tego samego rozmiaru układu błąd jest względnie niewielki, rzędu 10-5. Poważne wartości błędów (przekraczające 1) pojawiają się jednak już dla P*=15*, zatem zysk ze zwiększenia precyzji obliczeń jest niewielki. W arytmetyce pojedynczej precyzji błędy przekraczające wartość 1 pojawiały się już dla *N=7* i rosły zdecydowanie szybciej, niż w arytmetyce podwójnej precyzji.

Wiemy, że błędy wektora różnicy zależą m.in. od normy. Norma euklidesowa w tym wypadku jest bardziej restrykcyjna (generuje większe błędy) niż norma maksimum. Obie są jednak zbliżonego rzędu.

**3.3. Zadanie drugie.**

Drugie zadanie polegało na powtórzeniu eksperymentu z poprzedniego ćwiczenia dla macierzy **A** zadanej wzorem:



Podobnie jak poprzednio, stosując metodę eliminacji Gaussa, wyznaczono wartość wektora rozwiązań oraz normy wektora błędu dla różnych precyzji oraz rozmiarów układu.

Wyniki przeprowadzonego zadania przedstawia poniższa tabela:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **P** | **DOUBLE** | | **FLOAT** | | **UWARUNKOWANIE** |
|  |  |  |  |
| 10 | 4.931e-15 | 3.553e-15 | 3.405e-06 | 2.503e-06 | 114.729 |
| 20 | 1.335e-14 | 7.550e-15 | 1.174e-05 | 7.093e-06 | 472.505 |
| 30 | 7.314e-14 | 4.263e-14 | 4.257e-05 | 1.842e-05 | 1072.874 |
| 40 | 1.486e-13 | 9.837e-14 | 7.727e-05 | 2.801e-05 | 1916.041 |
| 50 | 2.221e-13 | 1.261e-13 | 0.0001268 | 4.208e-05 | 3001.815 |
| 75 | 8.093e-13 | 3.336e-13 | 0.0003746 | 0.0001526 | 6777.6224 |
| 100 | 1.558e-12 | 5.280e-13 | 0.0006317 | 0.0003012 | 12069.671 |
| 150 | 2.934e-12 | 7.501e-13 | 0.001817 | 0.0004358 | 27202.486 |
| 200 | 6.476e-12 | 1.652e-12 | 0.003804 | 0.001352 | 48400.613 |
| 300 | 1.737e-11 | 4.182e-12 | 0.01070 | 0.002207 | 1.090e+05 |
| 400 | 4.311e-11 | 1.040e-11 | 0.02166 | 0.004543 | 1.939e+05 |
| 500 | 6.959e-11 | 1.105e-11 | 0.03853 | 0.00993 | 3.030e+05 |
| 750 | 1.783e-10 | 3.260e-11 | 0.1087 | 0.01484 | 6.819e+05 |
| 1000 | 4.308e-10 | 5.941e-11 | 0.2092 | 0.03014 | 1.212e+06 |

**Wnioski:**

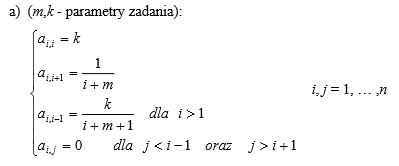
Porównując do poprzedniego zadania łatwo obserwujemy dużo lepszy wskaźnik uwarunkowania zadania. Dla rozmiaru *P=20*, uwarunkowanie macierzy w ćwiczeniu pierwszym osiągało rząd 1019, zaś w zadaniu drugim wartość ta jest rzędu 102 przy tym samym rozmiarze układu. Co więcej, dla *P=1000* wskaźnik uwarunkowania osiąga wartość rzędu 106. Ten sam rząd uwarunkowania występował dla macierzy w zadaniu pierwszym przy rozmiarze *P=5*. Dzięki wysokiej dokładności obliczeń zachowana jest również tendencja rosnąca wartości wskaźnika uwarunkowania macierzy. Ryzyko błędnego jego wyznaczenia jest niewielkie, gdyż metoda eliminacji Gaussa nie generuje poważnych błędów (z dużą dokładnością wyznaczana jest macierz odwrotna).

Zadanie to zostało rozwiązane w zakresie *P=1000* generując znacznie mniejsze błędy, niż zadanie pierwsze w zakresie *P=20*. Możemy zaobserwować również dużo wolniejszy, niewykładniczy wzrost wskaźnika uwarunkowania macierzy z zadania drugiego.

Wartość wskaźnika uwarunkowania jest ściśle powiązana z błędami generowanymi przy obliczeniach arytmetycznych. Błąd obliczeniowy dla arytmetyki podwójnej precyzji oscyluje w granicach: od 10-15 do 10-10, zaś dla pojedynczej precyzji: od 10-6 do 10-1. Zwracając uwagę na dużo mniejsze błędy, należy również nadmienić kwestię rozmiaru układu (błędy rzędu 10-10 w arytmetyce podwójnej precyzji oraz 10-1 w arytmetyce pojedynczej precyzji są generowane dla układu o rozmiarze *P=1000*; w zadaniu pierwszym błędy tego rzędu pojawiały się dla *P=6*).

**3.4. Zadanie trzecie a.**

Celem tego zadania jest jak w poprzednich ćwiczeniach - wyznaczenie błędów wektora rozwiązań oraz wskaźnika uwarunkowania macierzy **A** zadanej wzorem:



dla *k=3* oraz *m=3.* Rozwiązanie powyższego układu przeprowadzano na dwa sposoby: stosując algorytm eliminacji Gaussa oraz algorytm Thomasa. Przeanalizowano również działanie obu algorytmów pod kątem zużycia czasu procesora oraz pamięci.

Wyniki przeprowadzonego zadania przedstawia poniższa tabela:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **P** | **DOUBLE** | | **FLOAT** | | **UWARUNKOWANIE** |
|  |  |  |  |
| 10 | 0 | 0 | 1.788e-07 | 1.192e-07 | 1 |
| 20 | 3.310e-16 | 2.220e-16 | 2.598e-07 | 1.192e-07 | 1 |
| 30 | 4.839e-16 | 2.220e-16 | 2.980e-07 | 1.192e-07 | 1 |
| 40 | 5.439e-16 | 2.220e-16 | 3.265e-07 | 1.192e-07 | 1 |
| 50 | 6.661e-16 | 2.220e-16 | 3.372e-07 | 1.192e-07 | 1 |
| 75 | 8.158e-16 | 2.220e-16 | 4.420e-07 | 1.192e-07 | 1 |
| 100 | 9.088e-16 | 2.220e-16 | 5.093e-07 | 1.192e-07 | 1 |
| 150 | 1.053e-15 | 2.220e-16 | 6.796e-07 | 1.192e-07 | 1 |
| 200 | 1.241e-15 | 2.220e-16 | 7.749e-07 | 1.192e-07 | 1 |
| 300 | 1.590e-15 | 2.220e-16 | 9.157e-07 | 1.192e-07 | 1 |
| 400 | 1.807e-15 | 2.220e-16 | 1.041e-06 | 1.192e-07 | 1 |
| 500 | 2.038e-15 | 2.220e-16 | 1.165e-06 | 1.192e-07 | 1 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **P** | **DOUBLE** | | | | **FLOAT** | | | |
| **Pamięć alg. Gaussa** | **Pamięć alg. Thomasa** | **Czas alg. Gaussa** | **Czas alg. Thomasa** | **Pamięć alg. Gaussa** | **Pamięć alg. Thomasa** | **Czas alg. Gaussa** | **Czas alg. Thomasa** |
| 10 | 1040 | 480 | 11 | 2 | 520 | 240 | 4 | 1 |
| 20 | 3680 | 960 | 60 | 3 | 1840 | 480 | 42 | 2 |
| 30 | 7920 | 1440 | 239 | 5 | 3960 | 720 | 89 | 2 |
| 40 | 13760 | 1920 | 494 | 6 | 6880 | 960 | 217 | 6 |
| 50 | 21200 | 2400 | 1018 | 7 | 10600 | 1200 | 374 | 9 |
| 75 | 46800 | 3600 | 3926 | 11 | 23400 | 1800 | 2610 | 11 |
| 100 | 82400 | 4800 | 6187 | 12 | 41200 | 2400 | 5537 | 13 |
| 150 | 183600 | 7200 | 11041 | 10 | 91800 | 3600 | 9964 | 14 |
| 200 | 324800 | 9600 | 32997 | 10 | 162400 | 4800 | 27261 | 17 |
| 300 | 727200 | 14400 | 89753 | 18 | 363600 | 7200 | 94142 | 17 |
| 400 | 1289600 | 19200 | 21094 | 21 | 644800 | 9600 | 215072 | 20 |
| 500 | 2012000 | 24000 | 391684 | 30 | 1006000 | 12000 | 404200 | 28 |

**Wnioski:**

Wskaźnik uwarunkowania zadanej w tym zadaniu macierzy jest stale równy 1 dla wszystkich liczb z badanego zakresu: *P=10* do *P=500*. Przez to obliczenia numeryczne mogą być przeprowadzane niemal bezbłędnie (z dokładnością do błędów reprezentacji, czy zaokrągleń oraz błędów na ostatnich miejscach rozwinięcia dziesiętnego).

Istotnie, obserwowane w tym zadaniu błędy są znikome (rzędu 10-15 dla arytmetyki podwójnej precyzji oraz 10-6 dla arytmetyki podwójnej precyzji i rozmiaru układu rzędu 102). Wartości błędów w metryce maksimum są stałe, natomiast w metryce euklidesowej nieznacznie rosną. Wartości błędów w obu arytmetykach i dla każdego rozmiaru układu (w pojedynczej i podwójnej precyzji obliczeń) były takie same przy zastosowaniu zarówno algorytmu eliminacji Gaussa, jak i algorytmu Thomasa. Wynika to oczywiście z faktu, iż algorytm Thomasa jest uproszczoną wersją metody Gaussa pomijającą zbędne działania arytmetyczne na elementach macierzy o wartości 0.

Analiza zużycia pamięci oraz czasu procesora jest bardzo niekorzystna dla metody eliminacji Gaussa. Oba zasoby rosną nieliniowo wraz ze wzrostem rozmiaru układu. Dla dużych rozmiarów macierzy (*P=500*) można zaobserwować ogromne różnice (sięgające kilku rzędów wielkości) zarówno w kwestii zużycia czasu procesora, jak i pamięci operacyjnej.

**Pamięć**

Komputer, na którym został wykonany eksperyment (WINDOWS 10) ma następujące standardy odnośnie rozmiarów danych: typ arytmetyczny podwójnej precyzji (*double*) zajmuje 8 bajtów, zaś typ arytmetyczny pojedynczej precyzji (*float*) – 4 bajty. Oprócz stałej pamięci zużywanej przez oba algorytmy zużywana jest również dodatkowa pamięć, której wielkość jest uzależniona od rozmiarów macierzy. W poniższej tabeli zebrano ilość potrzebnych komórek pamięci dla poprawnego działania obu algorytmów w zależności od rozmiaru układu *P*:

|  |  |  |
| --- | --- | --- |
| **Wymagane struktury** | **Metoda eliminacji Gaussa** | **Algorytm Thomasa** |
| Macierz współczynników **A** | *P2* | *3P* |
| Wektor wyrazów wolnych **B** | *P* | *P* |
| Wektor rozwiązań **X** | *P* | *P* |

Widzimy, że zaletą algorytmu Thomasa jest to, że do poprawnego działania potrzebuje wyłącznie *O(P)* zaalokowanych komórek pamięci, zatem ich zużycie wzrasta liniowo wraz ze wzrostem wielkości układu. Metoda eliminacji Gaussa zużywa *O(P2)* komórek pamięci. Czyli P krotnie więcej.

Kolejna zaletą algorytmu Thomasa jest krótszy czas działania. W tym wypadku zysk jest jeszcze większy, gdyż metoda eliminacji Gaussa działa w czasie *O(P3)*, zaś algorytm Thomasa – w czasie *O(P)*. Algorytm Thomasa działa trzykrotnie szybciej niż algorytm Gaussa.

1. **Wnioski z wykonanych zadań.**

* zadanie obliczenia układu zadanego w pierwszym ćwiczeniu jest źle uwarunkowane – wskaźnik uwarunkowania macierzy wyrazów wolnych rośnie w sposób wykładniczy. Już dla małych rozmiarów układu otrzymujemy wyniki odbiegające w znaczy sposób od wartości rzeczywistych.
* wskaźnik uwarunkowania macierzy w zadaniu drugim przyjmuje znacznie mniejsze wartości, niż wskaźnik uwarunkowania macierzy z zadania pierwszego. Tempo jego wzrostu jest wielokrotnie mniejsze. Przekłada się to bezpośrednio na błędy generowane w trakcie obliczeń zmiennoprzecinkowych – dla znacznie (nawet kilkuset-krotnie) większych układów generowane błędy są mniejsze nawet o 20 rzędów wielkości. Wskaźnik uwarunkowania jest wielkością, przy pomocy której możemy ocenić, jakie trudności możemy napotkać podczas rozwiązywania numerycznego układów równań metodami bezpośrednimi.
* zadanie rozwiązania układu zadanego w trzecim ćwiczeniu jest idealnie uwarunkowane z punktu widzenia metod numerycznych. Wskaźnik uwarunkowania macierzy jest stale równy 1, co powoduje, iż błędy obliczeniowe są znikomo małe
* algorytm Thomasa jest doskonałym usprawnieniem metody eliminacji Gaussa dla macierzy trójdiagonalnych – zużywa on mniej pamięci oraz ma dużo lepszą (liniową) złożoność obliczeniową