Міністерство освіти і науки України

Національний університет “Львівська політехніка”

Інститут комп’ютерних наук та інформаційних технологій

Кафедра програмного забезпечення



**Звіт**

Про виконання лабораторної роботи № 6

**На тему:**

«Розв’язування перевизначених систем лінійних алгебраїчних рівнянь»

**Лектор:**

доц. каф. ПЗ

Мельник Н. Б.

**Виконала:**

ст. гр. ПЗ-18

Кириченко М.К.

**Прийняв:**

проф. каф. ПЗ

Гавриш В.І.

« … » … 2022 р.

∑ = \_\_\_\_\_\_\_\_\_\_\_\_

Львів – 2022

**Тема**: розв’язування перевизначених систем лінійних алгебраїчних рівнянь.

**Мета**: ознайомлення на практиці з методами розв’язування перевизначених систем лінійних алгебраїчних рівнянь.

**Завдання**

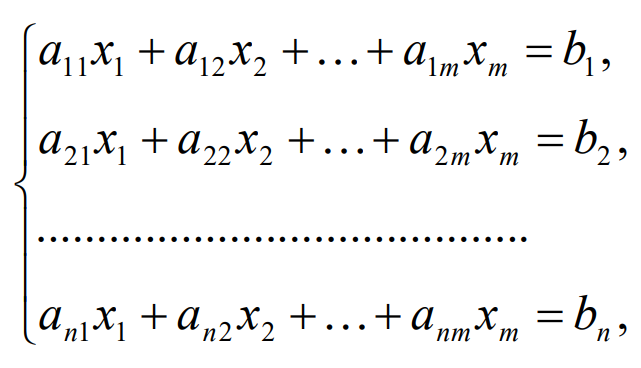
Варіант 8

Розв’язати перевизначену систему лінійних алгебраїчних рівнянь методом найменших квадратів. Отриману відповідну нормальну систему розв’язати методом квадратного кореня.

Код програмної реалізації подано у додатку.

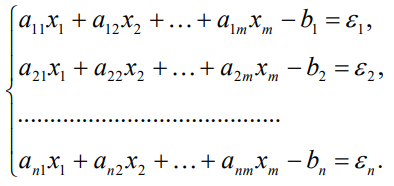
**Метод найменших квадратів для розв'язування перевизначених СЛАР**

Розглянемо систему лінійних алгебраїчних рівнянь, у якій кількість рівнянь є більшою за кількість невідомих

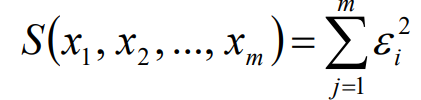
**** (рис. 1)

де n > m.

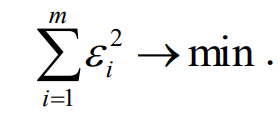
Знайдемо розв’язок системи х1, х2, ..., хm наближено, але щоб він задовольняв усі рівняння системи (рис. 1), а саме

 (рис. 2)

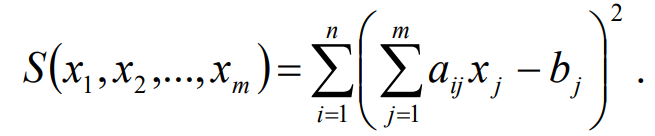
Розв’язок системи (рис. 2) будемо знаходити з використанням умови мінімізації суми квадратів відхилень, тобто з умов мінімізації функції

 (рис. 3)

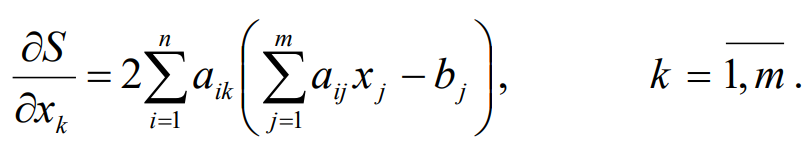
і вимагатимемо, щоб виконувалась умова

**** (рис. 4)

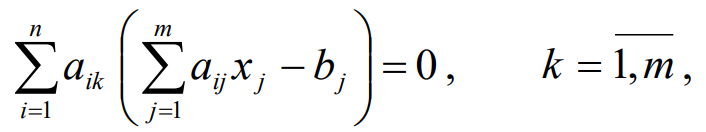
Проведемо деякі перетворення над системою (рис. 2), використовуючи умову (рис. 4). Розглянемо функцію

**** (рис. 5)

Необхідною умовою мінімуму функції від багатьох змінних є рівність нулеві її частинних похідних. Використаємо цей факт і продиференціюємо функцію (рис. 5) за змінними i x (i =1,m). У результаті отримаємо

**** (рис. 6)

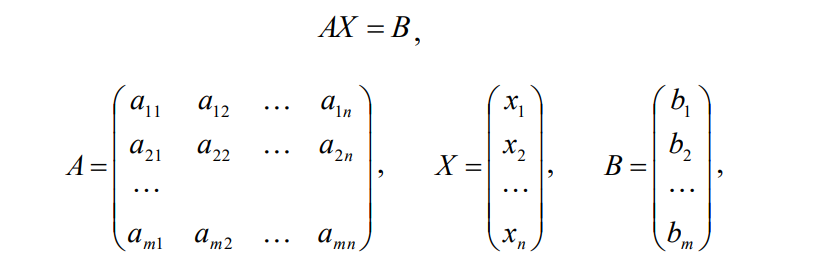
Прирівнявши вирази (рис. 6) до нуля, отримаємо нормальну систему рівнянь

**** (рис. 7)

в якій кількість рівнянь системи дорівнює кількості невідомих. Нормальні системи лінійних алгебраїчних рівнянь характеризуються тим, що матриці їх коефіцієнтів завжди є симетричними, а діагональні елементи - додатніми.

Систему (рис. 7) розв'язують довільними прямими або ітераційними методами. Якщо матриця коефіцієнтів системи рівнянь (рис. 7) є додатньо визначеною (визначник матриці є більшим за нуль), то рекомендують для її розв’язування використовувати метод квадратного кореня.

Запишемо систему лінійних алгебраїчних рівнянь (рис. 1) у матричному вигляді

 (рис. 8)

де A - матриця коефіцієнтів системи розмірності m × n, X – матриця-стовпець невідомих розмірності m × 1, B - матриця-стовпець вільних членів системи розмірності m × 1.

Матричне рівняння (рис. 8) помножимо на транспоновану матрицю AТ до матриці A . У результаті отримаємо матричне рівняння

NX = C (рис. 9)

де N – матриця коефіцієнтів нормальної системи

N = AТ А, (рис. 10)

C -стовпець вільних членів

С = AТ В. (рис. 11)

Розв’язавши нормальну систему лінійних алгебраїчних рівнянь, отримаємо її точний розв’язок (якщо використано прямі методи) або наближений розв’язок (якщо використано ітераційні методи). Отриманий розв’язок буде наближеним для СЛАР (рис. 1).

**Метод квадратного кореня**

Якщо матриця невироджена та �симетрична, то ми можемо розкласти її на добуток матриць �=����A = LDLТ, де L �llUISCHD — одинична нижня [трикутна матриця](https://uk.wikipedia.org/wiki/%D0%A2%D1%80%D0%B8%D0%BA%D1%83%D1%82%D0%BD%D0%B0_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D1%8F); D  �dD— [діагональна матриця](https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D1%8F).

Отримаємо систему:����⋅�=�L

LDLТ \* x = b

Розв'язок �отримаємо послідовно розв'язавши дві трикутні СЛАР:

LD \* y = b та

LТ \* x = y.

Порівняно з загальнішими методами ([метод Гауса](https://uk.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%93%D0%B0%D1%83%D1%81%D0%B0) чи [LU-розклад матриці](https://uk.wikipedia.org/wiki/LU-%D1%80%D0%BE%D0%B7%D0%BA%D0%BB%D0%B0%D0%B4_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D1%96)) він стійкіший і потребує вдвічі менше арифметичних операцій.

**Результат виконання завдання**

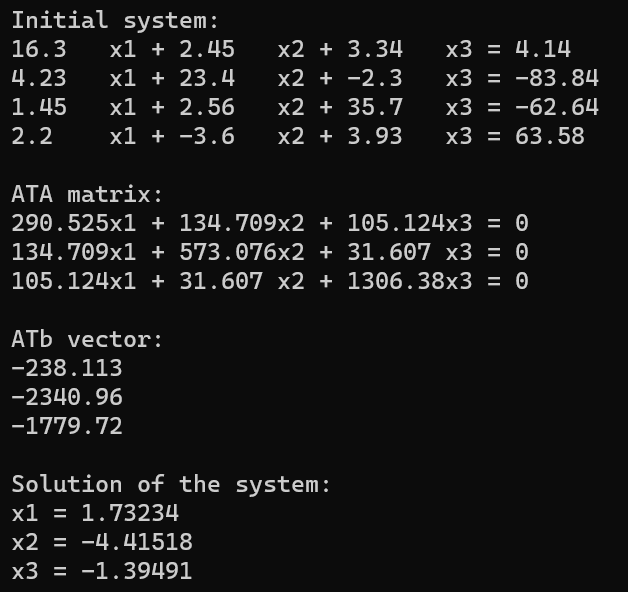
****

Рис. 12. Результат обчислення CЛАР методом найменших квадратів

**Висновки**

У результаті виконання лабораторної роботи створено обчислювальний

алгоритм, за допомогою мови С++, для розв’язування перевизначеної системи

лінійних алгебраїчних рівнянь методом найменших квадратів для заданої

системи лінійних алгебраїчних рівнянь.

**Додаток**

**Main.cpp**

#include <iostream>

#include <vector>

#include <cmath>

#include <iomanip>

void displaySystem(const std::vector<std::vector<double>>& A, const std::vector<double>& b) {

size\_t m = A.size();

size\_t n = A[0].size();

for (size\_t i = 0; i < m; ++i) {

for (size\_t j = 0; j < n; ++j) {

std::cout << std::setw(7) << std::left << A[i][j] << "x" << j + 1;

if (j < n - 1) {

std::cout << " + ";

}

}

std::cout << " = " << std::setw(7) << std::left << b[i] << std::endl;

}

}

std::vector<double> solveLinearSystem(std::vector<std::vector<double>>& mat, std::vector<double>& vec) {

size\_t n = vec.size();

for (size\_t i = 0; i < n; i++) {

for (size\_t j = i + 1; j < n; j++) {

double ratio = mat[j][i] / mat[i][i];

for (size\_t k = 0; k < n; k++) {

mat[j][k] -= ratio \* mat[i][k];

}

vec[j] -= ratio \* vec[i];

}

}

std::vector<double> res(n);

for (int i = n - 1; i >= 0; i--) {

res[i] = vec[i];

for (size\_t j = i + 1; j < n; j++) {

res[i] -= mat[i][j] \* res[j];

}

res[i] /= mat[i][i];

}

return res;

}

std::vector<double> leastSquares(const std::vector<std::vector<double>>& A, const std::vector<double>& b) {

size\_t m = A.size();

size\_t n = A[0].size();

std::vector<std::vector<double>> ATA(n, std::vector<double>(n, 0.0));

std::vector<double> ATb(n, 0.0);

for (size\_t i = 0; i < n; ++i) {

for (size\_t j = 0; j < n; ++j) {

for (size\_t k = 0; k < m; ++k) {

ATA[i][j] += A[k][i] \* A[k][j];

}

}

for (size\_t k = 0; k < m; ++k) {

ATb[i] += A[k][i] \* b[k];

}

}

std::cout << "\nATA matrix:\n";

displaySystem(ATA, std::vector<double>(n, 0.0));

std::cout << "\nATb vector:\n";

for (double v : ATb) {

std::cout << std::setw(7) << std::left << v << std::endl;

}

return solveLinearSystem(ATA, ATb);

}

int main() {

std::vector<std::vector<double>> A = {

{16.30, 2.45, 3.34},

{4.23, 23.40, -2.30},

{1.45, 2.56, 35.70},

{2.20, -3.60, 3.93}

};

std::vector<double> b = { 4.14, -83.84, -62.64, 63.58 };

std::cout << "Initial system:\n";

displaySystem(A, b);

std::vector<double> x = leastSquares(A, b);

std::cout << "\nSolution of the system:\n";

for (size\_t i = 0; i < x.size(); ++i) {

std::cout << "x" << i + 1 << " = " << std::setw(7) << std::left << x[i] << "\n";

}

return 0;

}