

НИЯУ МИФИ

Отчет по курсу СГМ

«Решения СЛАУ методом верхней релаксации»

Выполнил: студент группы Б17-511 Чудновец Иван

Проверил: Козин Рудольф Глебович

2019

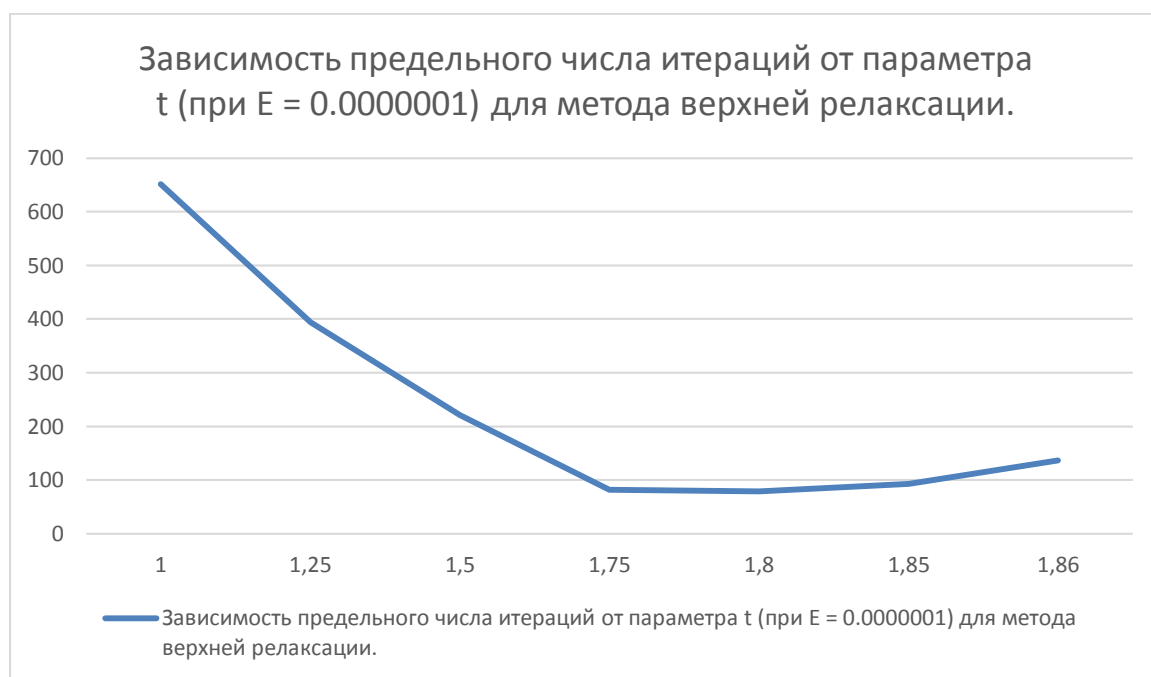
С помощью программы(код см. ниже) было проведено исследование сходимости метода верхней релаксации для модельной системы (табл. 1). Результаты исследования приведены в табл. 2.

$i \backslash j$	1	2	b	x	невязка
1	1	2	1		
2	3	4	2		

Матрица, правая часть системы $Ax = b$, её решение и невязка для полученного решения

t	1	1,25	1,5	1,75	1,8	1,85	1,86	1,87
Итерации	651	394	221	82	79	93	137	Расходится

Зависимость предельного числа итераций от параметра t (при $E = 0.0000001$) для метода верхней релаксации.



Код программы:

```
#include <iostream>
#include <cmath>
#include <time.h>
#include <cstdlib>
#include <iomanip>
#define B(i, j) MM[i * (n + 2) + j]
#define A(i, j) M[i * (n + 2) + j]
using namespace std;

float *M, *MM, eps, epst, t;
int n, iter, it;
int iter_Up();
void var_M();
void Ax_b();
int main() {
    setlocale(LC_ALL, "");
    int i, j, code, fl = 0, c;
    int fl1 = 0, iopt;
    float topt, dt = 0.05;
    char s[2];
    cout << "***Программа находит решение ";
    cout << "системы методом верхней релаксации***\n";
    cout << "Введите размерность системы: ";
    cin >> n;
    M = new float[n * (n + 2)];
    MM = new float[n * (n + 2)];
    cout << "Введите 0/1 - задать систему случайным образом или вручную ";
    cin >> c;
    switch (c) {
        case 0:
            srand(time(NULL));
            for (i = 0; i < n; i++) {
                for (j = 0; j < n; j++)
                    B(i, j) = 0.5 - rand() / (RAND_MAX + 1.0);
            }
            break;
        default:
            for (i = 0; i < n; i++) {
                for (j = 0; j < n; j++) {
                    cout << "Введите B(" << i << ", " << j << "): ";
                    cin >> B(i, j);
                }
                cout << "Введите b(" << i << "): ";
                cin >> B(i, n);
            }
    }

    cout << "Введённая система:\n";
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            cout << B(i, j) << " ";
        }
        cout << B(i, n) << "\n";
    }
    var_M();
    cout << "Преобразованная система:\n";
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            cout << A(i, j) << " ";
        }
        cout << A(i, n) << "\n";
    }
    while (fl == 0) {
        cout << "Число итераций, точность и параметр t: ";
        cin >> iter >> eps >> t;
```

```

var_M();
code = iter_Up();
switch (code) {
    case 0:
        cout << "Код = " << code << ", число использованных итераций";
        cout << " = " << it << ", точность = " << setprecision(8) << epst;
        cout << "\nРешение\n";
        for (i = 0; i < n; i++) cout << A(i, n + 1) << " ";
        Ax_b();
        cout << "\nНевязка\n";
        for (i = 0; i < n; i++) cout << B(i, n + 2) << " ";
        /*cout << "Найти оптимальное t -> введите 0, иначе 1 - ввод новых ";
        cout << "параметров, 2 - выход: ";
        cin >> fl1;
        if (fl == 1) break;
        if (fl == 2) {
            fl = 1;
            break;
        }
        if (fl == 0) {
            iopt = iter;
            t = 0;
            for (i = 0, i < 40; i++) {
                t += dt;
                var_M();
                code = iter_Up();
                if(!code)&&(it < iopt))
            }
        }
        cout << "\nПолучены следующие оптимальные значения t и iter: ";
        cout << setprecision(3) << topt << "; " << iopt;
        cout << "Повторить вычисления для новых параметров итерационного ";
        cout << "процесса - введите 0, иначе - 1: ";
        cin >> fl;
        */
        break;
    case 1:
        cout << "\nВ матрице системы есть диагональный компонент = 0";
        fl = 1;
        break;
    case 2:
        cout << "Метод разошёлся!";
        fl = 1;
        break;
}
}
cout << "\nДля завершения программы нажмите любую клавишу и ENTER";
cin >> s[0];
system("pause>nul");
return 0;
}

int iter_Up() {
    int i, j, k, fl = 0;
    float nt, buf, norm, norm_old = 0, t1 = 1 - t;
    for (i = 0; i < n; i++) {
        A(i, n + 1) = A(i, n);
        if (abs(A(i, i)) < 1.e-30) return 1;
        A(i, n) = t * A(i, n) / A(i, i);
        for (j = 0; j < n; j++) if (i != j) A(i, j) = t * A(i, j) / A(i, i);
    }
    it = 0;
    for (k = 1; k <= iter; k++) {
        it += 1; norm = 0;
        for (i = 0; i < n; i++) {
            buf = t1 * A(i, n + 1) + A(i, n);
            for (j = 0; j < n; j++) {
                if (i != j) buf = buf - A(i, j) * A(j, n + 1);
            }
            nt = abs(buf - A(i, n + 1));

```

```

        A(i, n + 1) = buf;
        if (nt > norm) norm = nt;
    }
    epst = norm;
    if (norm <= eps) return 0;
    if (norm > norm_old) fl = fl + 1;
    else fl = 0;
    if (fl > 5) return 2;
    norm_old = norm;
}
return 0;
}
void var_M() {
    int i, j, k;
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            A(i, j) = 0;
            for (k = 0; k < n; k++) {
                A(i, j) += B(k, i)*B(k, j);
            }
        }
    }
}
void Ax_b() {
    int i, j;
    for (i = 0; i < n; i++) {
        B(i, n + 1) = B(i, n);
        for (j = 0; j < n; j++) {
            B(i, n + 1) -= B(i, j)*A(j, n + 1);
        }
    }
}
}

```