

Московский Авиационный Институт  
(Национальный Исследовательский Университет)  
Кафедра 806

Лабораторная работа №3  
по курсу “Численные методы”

Студент: М.А. Трофимов  
Группа: М8О-408Б-18  
Преподаватель: Д.Е. Пивоваров  
Оценка: \_\_\_\_\_  
Дата: \_\_\_\_\_  
Подпись: \_\_\_\_\_

Москва, 2021

## Постановка задачи

Решить краевую задачу для дифференциального уравнения эллиптического типа. Аппроксимацию уравнения произвести с использованием центрально-разностной схемы. Для решения дискретного аналога применить следующие методы: метод простых итераций (метод Либмана), метод Зейделя, метод простых итераций с верхней релаксацией. Вычислить погрешность численного решения путем сравнения результатов с приведенным в задании аналитическим решением  $U(x,y)$ . Исследовать зависимость погрешности от сеточных параметров  $h_x, h_y$ .

1.

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} &= 0, \\ u(0, y) &= y, \\ u(1, y) &= 1 + y, \\ u(x, 0) &= x, \\ u(x, 1) &= 1 + x. \end{aligned}$$

Аналитическое решение:  $U(x, y) = x + y$ .

## Решение

Данное уравнение решалось с помощью конечно-разностной схемы. Программа писалась в несколько файлов: `iters.*` - заголовочник(.hpp) и исходный файл(.cpp) с методами простой итерации (Либмана) с возможностью, Зейделя, `EllipticEquation.*` - аналогично заголовочный и исходный файлы для класса решателя. `main.cpp` - основной файл, в котором описывается уравнение, оно решается и выводит необходимую информацию.

Программа поддерживает ключи:

- `time` - для вывода времени работы программы,
- `error` - выводит MSE по полученному решению,  
`hx <x_step>` - позволяет задавать шаг сетки по оси OX,
- `hy <y_step>` - позволяет задавать шаг сетки по оси OY.

В случае подачи неправильного ключа выдаётся ошибка и подсказка по использованию.

Все файлы лежат на гитхабе в папке 3lab:

<https://github.com/student31415/Chislaki2/>

## Демонстрация работы:

```
schizophrenia@home:~/labs/4kurs/Chislaki2/3lab$ make clean
rm *.o EE
schizophrenia@home:~/labs/4kurs/Chislaki2/3lab$ make -j plot_ans
g++ -c main.cpp
g++ -c EllipticEquation.cpp
g++ -c iters.cpp
g++ EllipticEquation.o main.o iters.o -o EE
./EE > ans.txt
```

Warning:  $\|\alpha\| \geq 1$

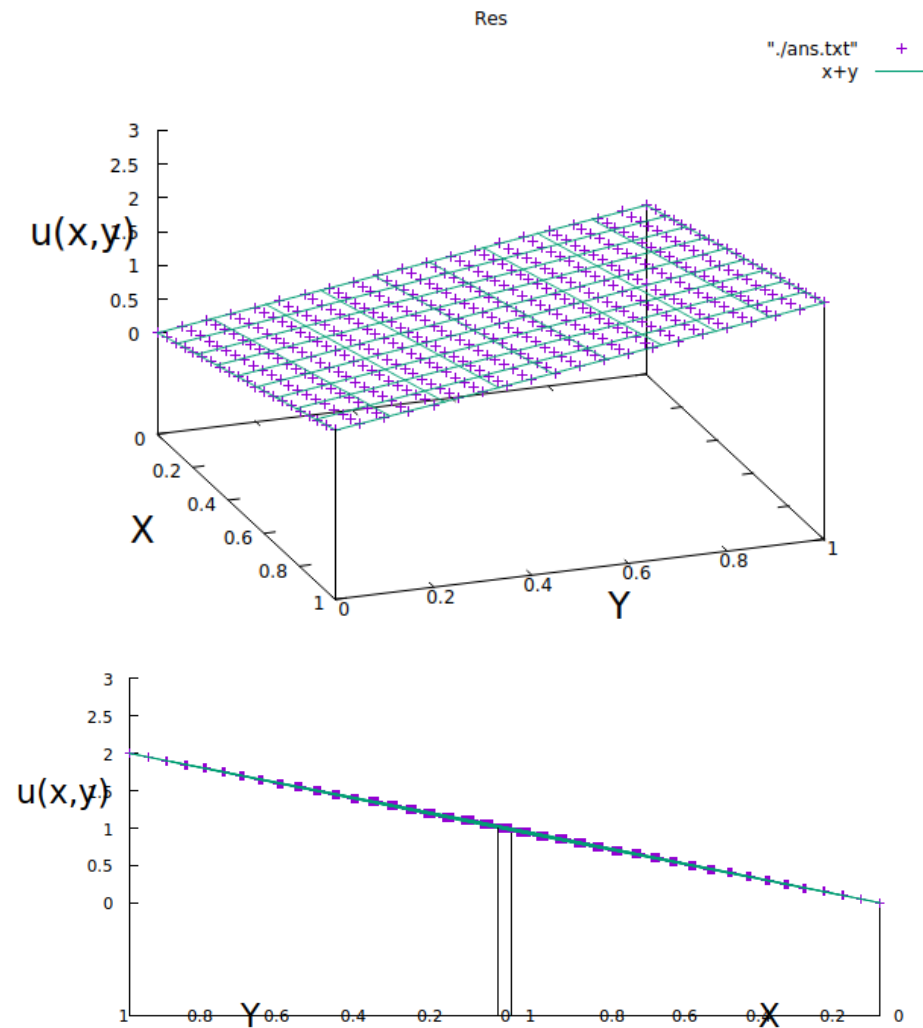
Alpha: 1

Iter finish: 729

gnuplot plot\_ans.gpi

Hit enter to continue

rm ans.txt



### Сравнение результатов работы

hx	hy	MSE	time, [ms]
0.5	0.5	0	0.086
0.33	0.33	2.85e-05	0.225

0.25	0.25	2.59997e-15	0.683
0.1	0.1	8.75278e-15	16.632
0.05	0.05	1.24372e-14	798.899
0.025	0.025	1.40509e-14	43461.8
0.01	0.01	-	>30min

### Вывод

Как видно, итоговая точность решения весьма большая, но основная сложность это время работы, которое уже на шагах 0.01, 0.01 превышает пол часа. Очевидно, что для достижения ускорения работы программы стоит смотреть в сторону использования более быстрых алгоритмов, заточенных под данную задачу, либо в сторону распараллеливания.