

Санкт-Петербургский политехнический университет Петра Великого  
Физико-механический институт  
Высшая школа теоретической механики и математической физики

Направление подготовки  
01.03.03 Механика и математическое моделирование

## Отчёт по лабораторной работе №3

Тема: "Уравнение Лапласа"

Дисциплина "Вычислительная механика"

**Выполнил:**  
Работинский А.Д.  
Группа: 5030103/10001  
**Преподаватель:**  
Е.Ю. Витохин

Санкт-Петербург  
2022

## 1) Постановка задачи

Решается уравнение Лапласа:  $\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = 0$

Задача решается для прямоугольника  $x \in [0; a]$   $y \in [0; b]$

Граничные условия:  $u(0, y) = \psi_0(y)$   $u(a, y) = \psi_M(y)$   $u(x, 0) = \phi_0(x)$   $u(x, b) = \phi_N(x)$

В нашем случае граничные условия имеют следующий вид:

$u(0, y) = 20y$   $u(a, y) = 30\sqrt{y}(1 - y)$   $u(x, 0) = 0$   $u(x, b) = 20(1 - x^2)$  Шаг по пространству:  $h = 0.1$ , требуемая точность решения  $\epsilon = 0.01$

Так же требуется найти  $\omega$ , которое позволяет найти решение за наименьшее количество итераций

## 2) Решение задачи

Конечно-разностные аппроксимации получаются схожим образом - из разложения в ряд Тейлора:

$$u(x_0 + h) = u(x_0) + \frac{h}{1!}u'(x_0) + \frac{h^2}{2!}u''(x_0) + o(h^2)$$

$$u(x_0 - h) = u(x_0) - \frac{h}{1!}u'(x_0) + \frac{h^2}{2!}u''(x_0) + o(h^2)$$

Осталось разрешить систему относительно  $u''(x_0)$ :

$$u''(x_0) = \frac{u(x_0 + h) - 2u(x_0) + u(x_0 - h)}{h^2}$$

Разложим в ряд Тейлора функцию перемещений около точки  $t_0$ :

$$u(y_0 + h) = u(y_0) + \frac{h}{1!}u'(y_0) + \frac{h^2}{2!}u''(y_0) + o(h^2)$$

$$u(y_0 - h) = u(y_0) - \frac{h}{1!}u'(y_0) + \frac{h^2}{2!}u''(y_0) + o(h^2)$$

Получаем аппроксимацию для:  $\ddot{u}(t_0)$ :

$$u''(y_0) = \frac{u(y_0 + h) - 2u(y_0) + u(y_0 - h)}{h^2}$$

Введем сетку следующим образом:

$$x = ih \quad i \in [0; N]$$

$$y = jh \quad j \in [0; M]$$

Подставляя полученные аппроксимации в исходное уравнение, получим:

$$u_{i,j} = \frac{1}{4}(u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1})$$

Выражение для начального приближения выглядит аналогично:

$$\tilde{u}_{i,j} = \frac{1}{4}(u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1})$$

Тогда выражение для поля  $u$  на следующей итерации получается следующим образом:

$$u_{i,j}^{k+1} = u_{i,j}^k + \omega(\tilde{u}_{i,j} - u_{i,j}^k)$$

Где  $\omega$ - коcтанта метода:  $\omega = 1 \Rightarrow$  метод Зейделя,  $\omega > 1 \Rightarrow$  метод последовательной верхней релаксации,  $\omega < 1 \Rightarrow$  метод последовательной нижней релаксации

Критерием остановки итерационного процесса служит:

$$\|u^{k+1} - u^k\|_{\infty} \leq \epsilon$$

Идея решения состоит в том, что на каждой итерации подсчитываются компоненты матрицы  $u^k$  далее пересчитывается начальное приближение:  $\tilde{u}$  и далее по формуле выше считаются компоненты матрицы  $u^{k+1}$

Получившиеся результаты для  $\omega \in (0; 2)$  представлены ниже:

**Получившиеся результаты:**

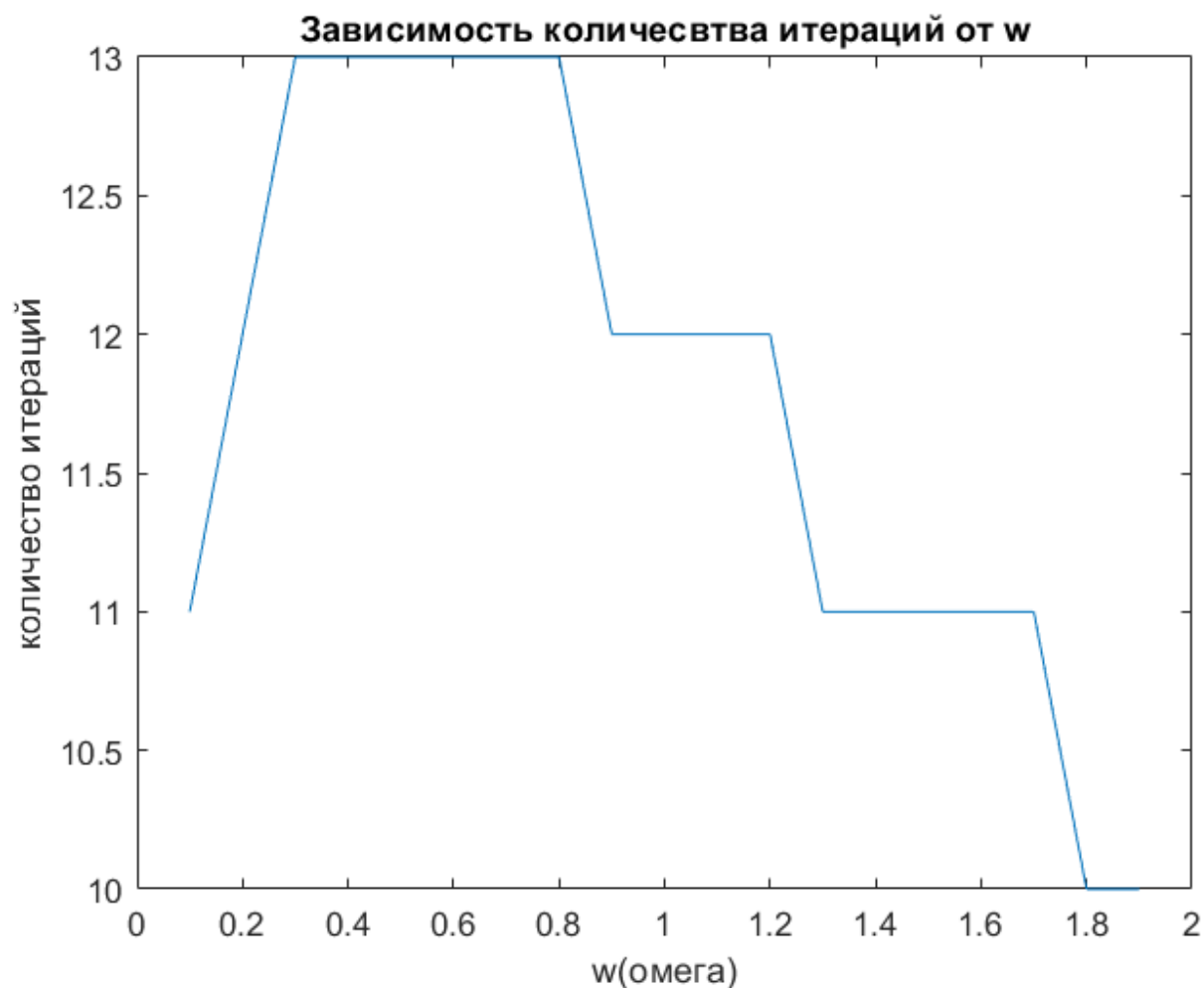


Рис. 1: График количества итераций от константы метода

Омега	11	12	13	13	13	13	13	13
Кол-во итераций	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8

Рис. 2: Таблица кол-ва итераций и соответствующих им констант метода

Отсюда несложно увидеть, что оптимальное  $\omega = 1.8$  и  $\omega = 1.9$ , график решения будет выглядеть следующим образом:

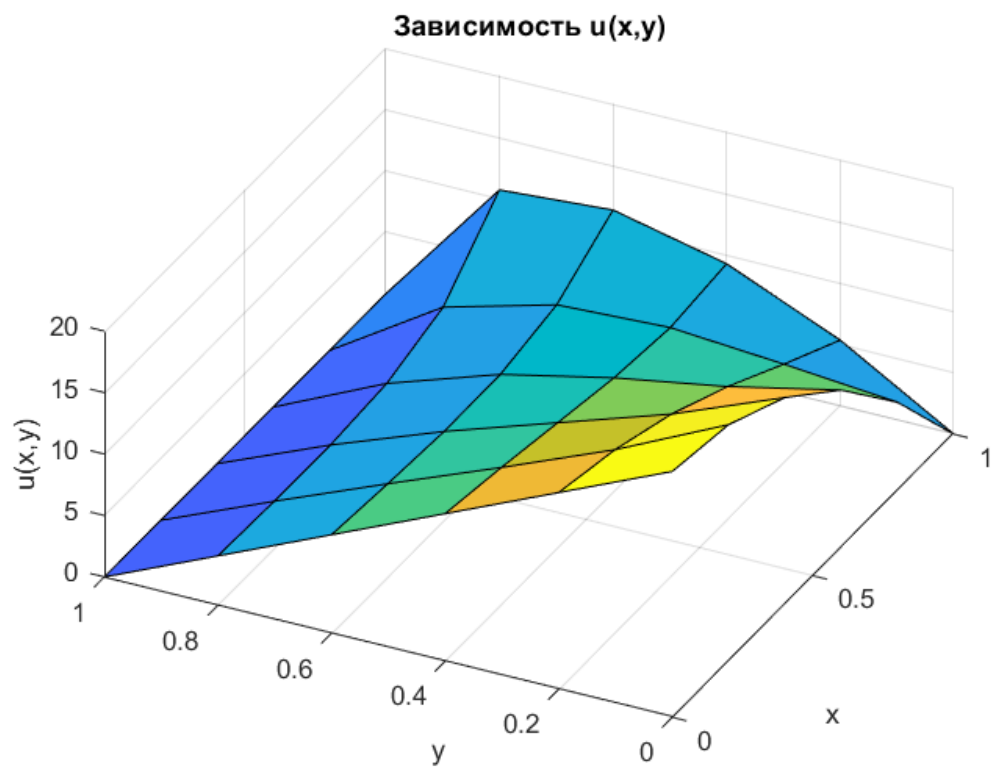


Рис. 3: Зависимость  $u(x,y)$

$X \longrightarrow$

	20,0000	19,2000	16,8000	12,8000	7,2000	0,0000
$\uparrow Y$	16,0000	14,8793	13,1659	10,8031	8,0184	5,3666
	12,0000	11,1512	10,1838	9,2290	8,7054	9,2952
	8,0000	7,5453	7,1899	7,2270	8,2799	11,3842
	4,0000	3,8372	3,8099	4,2097	5,8057	10,7331
	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000

Рис. 4: Таблица  $u(x,y)$

## Код (выполнен в MATLAB)

```
h=0.2;
x=[0:h:1];
y=[0:h:1];
N=size(x,2);
M=size(y,2);
kol_it=[];
eps=0.01;
for w=0.1:0.1:1.9
temp=1;
u=zeros(M,N);
kol=0;
for i=1:N
u(M,i)=0;
u(1,i)=20*(1-((i-1)*h)^2);
end
for j=M:-1:1
u(j,1)=20*((M-j)*h);
u(j,N)=30*sqrt((M-j)*h)*(1-(M-j)*h);
end
while(temp>eps)
for i=2:N-1
for j=M-1:-1:2
u(j,i)=0.25*(u(j,i-1)+u(j,i+1)+u(j-1,i)+u(j+1,i));
end
end
t=zeros(M,N);
t(1,:)=u(1,:);
t(M,:)=u(M,:);
t(:,1)=u(:,1);
t(:,N)=u(:,N);
for i=2:N-1
for j=M-1:-1:2
t(j,i)=0.25*(u(j,i-1)+u(j,i+1)+u(j-1,i)+u(j+1,i));
end
end
for i=1:N
for j=M:-1:1
u1(j,i)=u(j,i)+w*(t(j,i)-u(j,i));
end
end
temp=norm(u1-u,inf);
u=u1;
kol=kol+1;
end
kol_it(end+1)=kol;
end
figure()
plot(0.1:0.1:1.9,kol_it);
figure()
surf(x,y,u);
xlabel('x');
ylabel('y');
zlabel('u(x,y)');
```