



Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра системного анализа

Трусов Николай Всеволодович

Численное решение краевой задачи для уравнения Пуассона с потенциалом в прямоугольной области

ВТОРОЕ ЗАЧЁТНОЕ ЗАДАНИЕ

Группа:

615

Вариант задания:

8

Москва, 2019

Содержание

1	Постановка задачи	3
2	Определение функций $F(x, y)$, $\psi(x, y)$	5
3	Разностная схема задачи	6
4	Метод решения СЛАУ	18
5	Численная реализация на сгущающихся сетках	21
5.1	$M = 20, N = 20$	21
5.2	$M = 40, N = 40$	21
5.3	$M = 80, N = 80$	22
5.4	$M = 160, N = 160$	22
5.5	Приложение 1	23
	Список литературы	32

1 Постановка задачи

В прямоугольнике $\Pi = [A_1, A_2] \times [B_1, B_2]$, граница Γ которого состоит из отрезков

$$\gamma_R = \{(A_2, y), B_1 \leq y \leq B_2\}, \quad (1.1)$$

$$\gamma_L = \{(A_1, y), B_1 \leq y \leq B_2\}, \quad (1.2)$$

$$\gamma_T = \{(x, B_2), A_1 \leq x \leq A_2\}, \quad (1.3)$$

$$\gamma_B = \{(x, B_1), A_1 \leq x \leq A_2\}. \quad (1.4)$$

Рассматривается дифференциальное уравнение Пуассона с потенциалом

$$-\Delta u + q(x, y)u = F(x, y), \quad (1.5)$$

в котором оператор Лапласа

$$\Delta u = \frac{\partial}{\partial x} \left(k(x, y) \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(k(x, y) \frac{\partial u}{\partial y} \right). \quad (1.6)$$

Для выделения единственного решения уравнение (1.5) дополняется граничными условиями. На каждом отрезке границы прямоугольника Π задаются следующие условия:

1. граница γ_R : $\left(k \frac{\partial u}{\partial n} \right) (x, y) + u(x, y) = \psi(x, y);$

2. граница γ_L : $\left(k \frac{\partial u}{\partial n} \right) (x, y) + u(x, y) = \psi(x, y);$

3. граница γ_T : $\left(k \frac{\partial u}{\partial n} \right) (x, y) = \psi(x, y);$

4. граница γ_B : $\left(k \frac{\partial u}{\partial n} \right) (x, y) = \psi(x, y).$

Функции $F(x, y)$, $\psi(x, y)$, $k(x, y)$, $q(x, y)$ — известные. Необходимо найти функцию $u(x, y)$, удовлетворяющую уравнению (1.5) и введённым граничным условиям.

Замечание 1. Нормаль n не определена в угловых точках прямоугольника. Краевые условия следует рассматривать лишь в тех точках границы, где нормаль существует.

Требуется методом конечных разностей приближенно решить краевую задачу для уравнения Пуассона с потенциалом в прямоугольной области. Задание необходимо выполнить на следующих ПВС Московского университета:

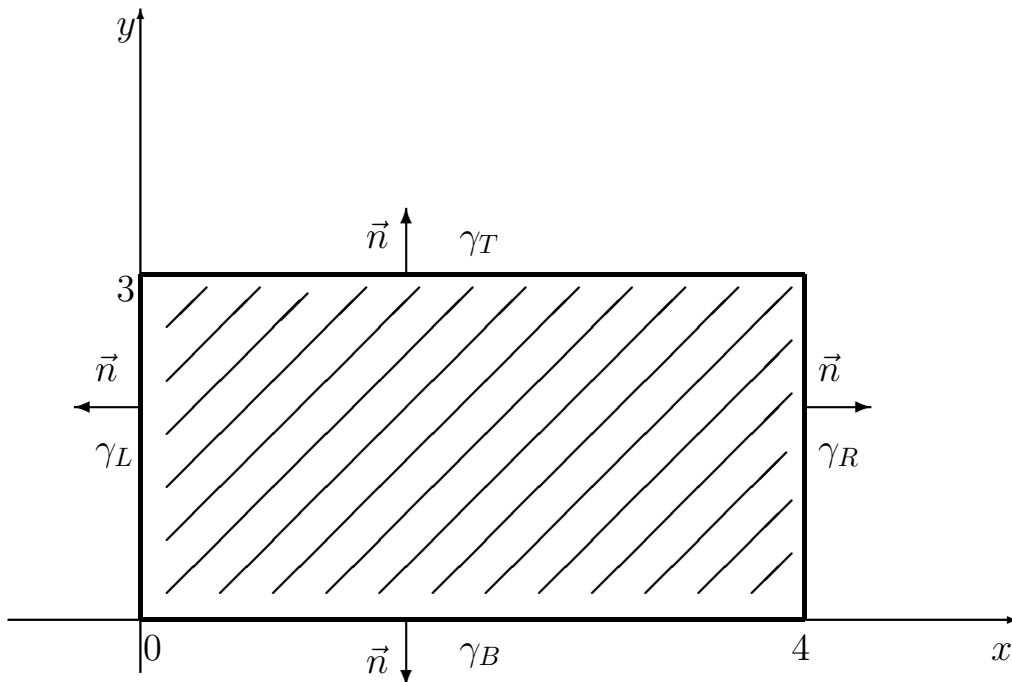
1. IBM Blue Gene/P;
2. IBM Polus.

В соответствии с вариантом задания, рассматриваются следующие данные:

- $A_1 = 0, A_2 = 4, B_1 = 0, B_2 = 3$;
- $u(x, y) = \sqrt{4 + xy}$;
- $k(x, y) = 4 + x + y$;
- $q(x, y) = (x + y)_+ = \max\{x + y, 0\} = x + y$;
- граничные условия представлены выше.

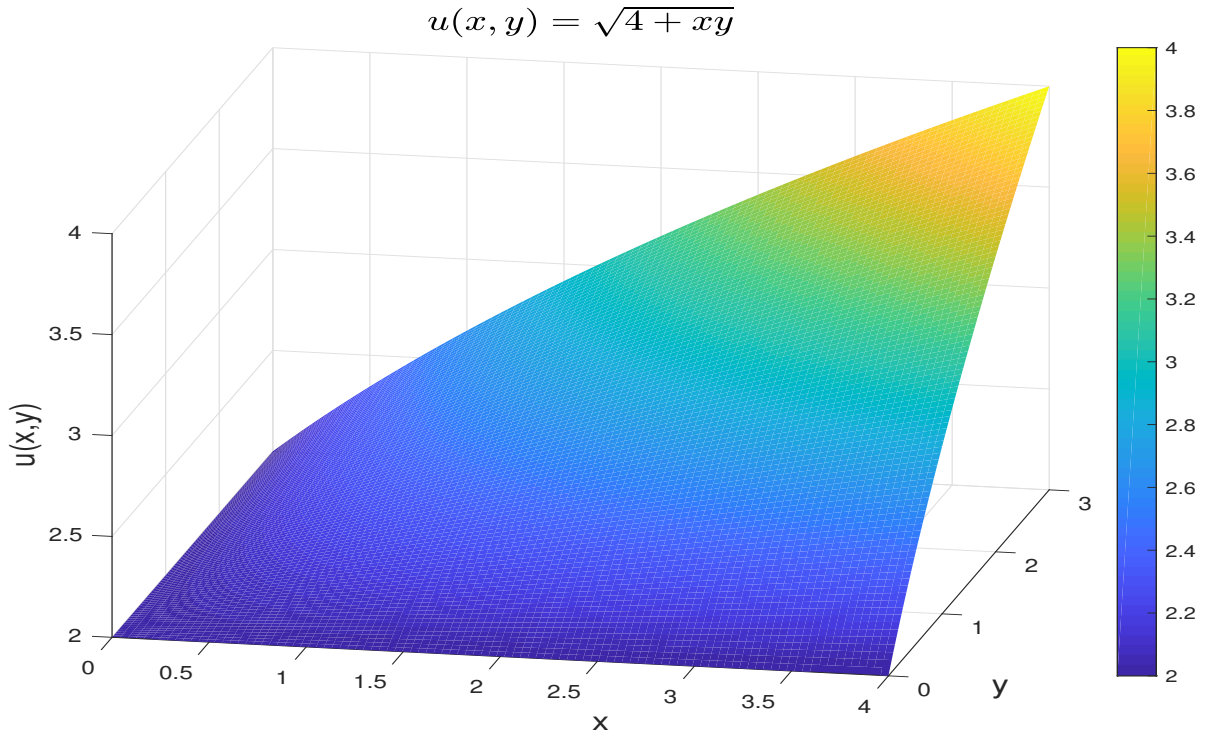
Задача 1. Задача практикума заключается в восстановлении известной гладкой функции $u(x, y)$ по её образу $F(x, y) = -\Delta u + q(x, y)u$ и её граничным значениям.

Схематично, можно представить следующую интерпретацию задачи:



где прямоугольник Π характеризует тонкую однородную пластину.

Требуется восстановить следующий вид функции $u(x, y)$:



2 Определение функций $F(x, y)$, $\psi(x, y)$

Определим функцию $F(x, y)$. Для этого вычислим оператор Лапласа, используя явные вид функций $u(x, y)$, $k(x, y)$:

$$\begin{aligned} \Delta u &= \frac{\partial}{\partial x} \left((4 + x + y) \cdot \frac{y}{2\sqrt{4 + xy}} \right) + \frac{\partial}{\partial y} \left((4 + x + y) \cdot \frac{x}{2\sqrt{4 + xy}} \right) = \\ &= \frac{y}{2\sqrt{4 + xy}} + (4 + x + y) \cdot \frac{y^2}{2} \cdot \left(-\frac{1}{2} \right) \cdot (4 + xy)^{-\frac{3}{2}} + \frac{x}{2\sqrt{4 + xy}} + \\ &+ (4 + x + y) \cdot \frac{x^2}{2} \cdot \left(-\frac{1}{2} \right) \cdot (4 + xy)^{-\frac{3}{2}} = \frac{1}{2\sqrt{4 + xy}}(x + y) - \frac{1}{4} \cdot \frac{4 + x + y}{(4 + xy)^{\frac{3}{2}}} (x^2 + y^2). \end{aligned} \quad (2.1)$$

Учитывая, что $q(x, y)u(x, y) = (x + y)\sqrt{4 + xy}$, правая часть уравнения (1.5) имеет следующий вид:

$$F(x, y) = \frac{1}{4} \cdot \frac{4 + x + y}{(4 + xy)^{\frac{3}{2}}} (x^2 + y^2) - \frac{1}{2\sqrt{4 + xy}}(x + y) + (x + y)\sqrt{4 + xy}. \quad (2.2)$$

Чтобы определить функцию $\psi(x, y)$, вычислим частные производные $\frac{\partial u}{\partial x}$, $\frac{\partial u}{\partial y}$:

$$\frac{\partial u}{\partial x} = \frac{y}{2\sqrt{4 + xy}}, \quad (2.3)$$

$$\frac{\partial u}{\partial y} = \frac{x}{2\sqrt{4+xy}}. \quad (2.4)$$

Из определения производной по направлению, несложно вычислить $\psi(x, y)$:

$$\psi(x, y) = \begin{cases} (8 + \tilde{y}) \cdot \frac{\tilde{y}}{4\sqrt{1+\tilde{y}}} + 2\sqrt{1+\tilde{y}}, & x = 4, \tilde{y} \in (0, 3); \\ -(4 + \tilde{y}) \cdot \frac{\tilde{y}}{4} + 2, & x = 0, \tilde{y} \in (0, 3); \\ (7 + \tilde{x}) \cdot \frac{\tilde{x}}{2\sqrt{4+3\tilde{x}}}, & \tilde{x} \in (0, 4), y = 3; \\ -(4 + \tilde{x}) \cdot \frac{\tilde{x}}{4}, & \tilde{x} \in (0, 4), y = 0. \end{cases} \quad (2.5)$$

3 Разностная схема задачи

Краевые задачи для уравнения Пуассона с потенциалом (1.5) предлагается численно решать методом конечных разностей. В расчётной области Π определяется равномерная прямоугольная сетка $\bar{\omega}_h = \bar{\omega}_2 \times \bar{\omega}_1$, где

$$\bar{\omega}_1 = \{x_i = ih_1, i = \overline{0, M}\}; \quad (3.1)$$

$$\bar{\omega}_2 = \{y_j = jh_2, j = \overline{0, N}\}. \quad (3.2)$$

Здесь $h_1 = \frac{4}{M}$, $h_2 = \frac{3}{N}$. Через ω_h обозначим множество внутренних узлов сетки $\bar{\omega}_h$, т.е. множество узлов сетки прямоугольника, не лежащих на границе Γ .

Рассмотрим линейное пространство H функций, заданных на сетке $\bar{\omega}_h$. Обозначим через ω_{ij} значение сеточной функции $\omega \in H$ в узле сетки $(x_i, y_j) \in \bar{\omega}_h$. Будем считать, что в пространстве H задано скалярное произведение и евклидова норма следующим образом

$$\langle u, v \rangle = h_1 h_2 \sum_{i=0}^M \sum_{j=0}^N \rho_{ij} u_{ij} v_{ij}, \quad (3.3)$$

где $\rho_{ij} = \rho^1(x_i) \rho^2(y_j)$:

$$\rho^1(x_i) = \begin{cases} 1, & i = \overline{1, M-1}; \\ \frac{1}{2}, & i = \{0, M\}. \end{cases} \quad (3.4)$$

$$\rho^2(y_j) = \begin{cases} 1, & j = \overline{1, N-1}; \\ \frac{1}{2}, & j = \{0, N\}. \end{cases} \quad (3.5)$$

$$\|u\|_E = \sqrt{\langle u, u \rangle}. \quad (3.6)$$

В методе конечных разностей дифференциальная задача математической физики заменяется конечно-разностной операторной задачей вида

$$A\omega = B, \quad (3.7)$$

где $A : H \rightarrow H$ — оператор, действующий в пространстве сеточных функций, а $B \in H$ — известная правая часть. Задача (3.7) называется разностной схемой. Решение этой задачи считается численным решением исходной дифференциальной задачи.

При построении разностной схемы следует аппроксимировать все уравнения краевой задачи их разностными аналогами — сеточными уравнениями, связывающими значения искомой сеточной функции в узлах сетки. Полученные таким образом уравнения должны быть функционально независимыми, а их общее количество — совпадать с числом неизвестных, т.е. с количеством узлов сетки.

Уравнение (1.5) во всех внутренних точка сетки аппроксимируется разностным уравнением

$$-\Delta_h \omega_{ij} + q_{ij} \omega_{ij} = F_{ij}, \quad i = \overline{1, M-1}, j = \overline{1, N-1}, \quad (3.8)$$

в котором $F_{ij} = F(x_i, y_j)$, $q_{ij} = q(x_i, y_j)$, разностный оператор Лапласа

$$\begin{aligned} \Delta_h \omega_{ij} = & \frac{1}{h_1} \left(k \left(x_i + \frac{1}{2} h_1, y_j \right) \frac{\omega_{i+1,j} - \omega_{ij}}{h_1} - k \left(x_i - \frac{1}{2} h_1, y_j \right) \frac{\omega_{ij} - \omega_{i-1,j}}{h_1} \right) + \\ & + \frac{1}{h_2} \left(k \left(x_i, y_j + \frac{1}{2} h_2 \right) \frac{\omega_{i,j+1} - \omega_{ij}}{h_2} - k \left(x_i, y_j - \frac{1}{2} h_2 \right) \frac{\omega_{ij} - \omega_{i,j-1}}{h_2} \right). \end{aligned} \quad (3.9)$$

Введём обозначения правой и левой разностных производных по переменным x, y соответственно:

$$\omega_{x,ij} = \frac{\omega_{i+1,j} - \omega_{ij}}{h_1}, \quad (3.10)$$

$$\omega_{\bar{x},ij} = \omega_{x,i-1,j} = \frac{\omega_{i,j} - \omega_{i-1,j}}{h_1}, \quad (3.11)$$

$$\omega_{y,ij} = \frac{\omega_{i,j+1} - \omega_{ij}}{h_2}, \quad (3.12)$$

$$\omega_{\bar{y},ij} = \omega_{y,i,j-1} = \frac{\omega_{i,j} - \omega_{i,j-1}}{h_2}. \quad (3.13)$$

Пусть также

$$a_{ij} = k \left(x_i - \frac{1}{2} h_1, y_j \right), \quad (3.14)$$

$$b_{ij} = k \left(x_i, y_j - \frac{1}{2} h_2 \right). \quad (3.15)$$

С учётом введённых обозначений (3.10)-(3.15), разностный оператор Лапласа (3.9) можно записать в более компактном виде

$$\Delta_h \omega_{ij} = (a\omega_{\bar{x}})_{x,ij} + (b\omega_{\bar{y}})_{y,ij}. \quad (3.16)$$

В самом деле, из (3.9) с учётом (3.10)-(3.15), получаем

$$\begin{aligned} \Delta_h \omega_{ij} &= \frac{1}{h_1} (a_{i+1,j} \omega_{x,ij} - a_{ij} \omega_{\bar{x},ij}) + \frac{1}{h_2} (b_{i,j+1} \omega_{y,ij} - b_{ij} \omega_{\bar{y},ij}) = \\ &= \frac{1}{h_1} (a_{i+1,j} \omega_{\bar{x},i+1,j} - a_{ij} \omega_{\bar{x},ij}) + \frac{1}{h_2} (b_{i,j+1} \omega_{\bar{y},ij+1} - b_{ij} \omega_{\bar{y},ij}) = \\ &= (a\omega_{\bar{x}})_{x,ij} + (b\omega_{\bar{y}})_{y,ij}. \end{aligned} \quad (3.17)$$

Аппроксимация граничных условий третьего рода на правой и левой сторонах прямоугольника имеет вид

$$\frac{2}{h_1} (a\omega_{\bar{x}})_{Mj} + \left(q_{Mj} + \frac{2}{h_1} \right) \omega_{Mj} - (b\omega_{\bar{y}})_{y,Mj} = F_{Mj} + \frac{2}{h_1} \psi_{Mj}, \quad j = \overline{1, N-1}; \quad (3.18)$$

$$-\frac{2}{h_1} (a\omega_{\bar{x}})_{1j} + \left(q_{0j} + \frac{2}{h_1} \right) \omega_{0j} - (b\omega_{\bar{y}})_{y,0j} = F_{0j} + \frac{2}{h_1} \psi_{0j}, \quad j = \overline{1, N-1}. \quad (3.19)$$

На верхней и нижней сторонах соответственно имеем

$$\frac{2}{h_2} (b\omega_{\bar{y}})_{iN} + q_{iN} \cdot \omega_{iN} - (a\omega_{\bar{x}})_{x,iN} = F_{iN} + \frac{2}{h_2} \psi_{iN}, \quad i = \overline{1, M-1}; \quad (3.20)$$

$$-\frac{2}{h_2} (b\omega_{\bar{y}})_{i1} + q_{i0} \cdot \omega_{i0} - (a\omega_{\bar{x}})_{x,i0} = F_{i0} + \frac{2}{h_2} \psi_{i0}, \quad i = \overline{1, M-1}. \quad (3.21)$$

Сеточных уравнений (3.8)-(3.21) недостаточно, чтобы определить разностную схему для задачи с граничными условиями Неймана и Дирихле. Требуются сеточные уравнения для угловых точек прямоугольника П. Они имеют следующий вид:

• В точке $(0, 0)$:

$$-\frac{2}{h_1} (a\omega_{\bar{x}})_{10} - \frac{2}{h_2} (b\omega_{\bar{y}})_{01} + \left(q_{00} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{00} = F_{00} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{00}; \quad (3.22)$$

• В точке $(4, 0)$:

$$\frac{2}{h_1} (a\omega_{\bar{x}})_{M0} - \frac{2}{h_2} (b\omega_{\bar{y}})_{M1} + \left(q_{M0} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{M0} = F_{M0} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{M0}; \quad (3.23)$$

• В точке $(4, 3)$:

$$\frac{2}{h_1} (a\omega_{\bar{x}})_{MN} + \frac{2}{h_2} (b\omega_{\bar{y}})_{MN} + \left(q_{MN} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{MN} = F_{MN} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{MN}; \quad (3.24)$$

• В точке $(0, 3)$:

$$-\frac{2}{h_1} (a\omega_{\bar{x}})_{1N} + \frac{2}{h_2} (b\omega_{\bar{y}})_{0N} + \left(q_{0N} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{0N} = F_{0N} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{0N}. \quad (3.25)$$

Замечание 2. Разностные схемы (3.7), аппроксимирующие все описанные выше краевые задачи для уравнения Пуассона с положительным потенциалом, обладают самосопряжённым и положительно определённым оператором A и имеют единственное решение при любой правой части.

Соберём все уравнения, получим систему линейных алгебраических уравнений (СЛАУ), состоящую из $(M+1) \times (N+1)$ уравнений и $(M+1) \times (N+1)$ неизвестной:

$$\left\{ \begin{array}{l} -\Delta_h \omega_{ij} + q_{ij} \omega_{ij} = F_{ij}, \quad i = \overline{1, M-1}, j = \overline{1, N-1}; \\ \frac{2}{h_2} (b\omega_{\bar{y}})_{iN} + q_{iN} \cdot \omega_{iN} - (a\omega_{\bar{x}})_{x,iN} = F_{iN} + \frac{2}{h_2} \psi_{iN}, \quad i = \overline{1, M-1}; \\ -\frac{2}{h_2} (b\omega_{\bar{y}})_{i1} + q_{i0} \cdot \omega_{i0} - (a\omega_{\bar{x}})_{x,i0} = F_{i0} + \frac{2}{h_2} \psi_{i0}, \quad i = \overline{1, M-1}; \\ \frac{2}{h_1} (a\omega_{\bar{x}})_{Mj} + \left(q_{Mj} + \frac{2}{h_1} \right) \omega_{Mj} - (b\omega_{\bar{y}})_{y,Mj} = F_{Mj} + \frac{2}{h_1} \psi_{Mj}, \quad j = \overline{1, N-1}; \\ -\frac{2}{h_1} (a\omega_{\bar{x}})_{1j} + \left(q_{0j} + \frac{2}{h_1} \right) \omega_{0j} - (b\omega_{\bar{y}})_{y,0j} = F_{0j} + \frac{2}{h_1} \psi_{0j}, \quad j = \overline{1, N-1}; \\ -\frac{2}{h_1} (a\omega_{\bar{x}})_{10} - \frac{2}{h_2} (b\omega_{\bar{y}})_{01} + \left(q_{00} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{00} = F_{00} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{00}; \\ \frac{2}{h_1} (a\omega_{\bar{x}})_{M0} - \frac{2}{h_2} (b\omega_{\bar{y}})_{M1} + \left(q_{M0} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{M0} = F_{M0} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{M0}; \\ \frac{2}{h_1} (a\omega_{\bar{x}})_{MN} + \frac{2}{h_2} (b\omega_{\bar{y}})_{MN} + \left(q_{MN} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{MN} = F_{MN} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{MN}; \\ -\frac{2}{h_1} (a\omega_{\bar{x}})_{1N} + \frac{2}{h_2} (b\omega_{\bar{y}})_{0N} + \left(q_{0N} + \frac{2}{h_1} + \frac{2}{h_2} \right) w_{0N} = F_{0N} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{0N}. \end{array} \right. \quad (3.26)$$

В самом деле, число уравнений системы (3.26) соответствует числу неизвестным $(M+1) \times (N+1)$:

$$\begin{aligned} & (M-1)(N-1) + (M-1) + (M-1) + (N-1) + (N-1) + 4 = \\ & = (M-1)(N-1) + 2(M-1) + 2(N-1) + 4 = \\ & = (M-1)(N-1+2) + 2(N-1) + 4 = (M-1)(N+1) + 2(N+1) = (M+1)(N+1). \end{aligned} \quad (3.27)$$

Распишем подробнее систему (3.26), выписав линейно ω_{ij} :

$$\left\{ \begin{array}{l}
 -\omega_{0j} \frac{a_{1j}}{h_1^2} - \omega_{1,j-1} \frac{b_{1j}}{h_2^2} + \omega_{1j} c_{1j} - \omega_{1,j+1} \frac{b_{1,j+1}}{h_2^2} - \omega_{2j} \frac{a_{2j}}{h_1^2} = F_{1j}, \quad j = \overline{1, N-1}; \\
 -\omega_{1j} \frac{a_{2j}}{h_1^2} - \omega_{2,j-1} \frac{b_{2j}}{h_2^2} + \omega_{2j} c_{2j} - \omega_{2,j+1} \frac{b_{2,j+1}}{h_2^2} - \omega_{3j} \frac{a_{3j}}{h_1^2} = F_{2j}, \quad j = \overline{1, N-1}; \\
 \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\
 -\omega_{M-2,j} \frac{a_{M-1,j}}{h_1^2} - \omega_{M-1,j-1} \frac{b_{M-1,j}}{h_2^2} + \omega_{M-1,j} c_{M-1,j} - \omega_{M-1,j+1} \frac{b_{M-1,j+1}}{h_2^2} - \\
 \qquad \qquad \qquad - \omega_{Mj} \frac{a_{Mj}}{h_1^2} = F_{M-1,j}, \quad j = \overline{1, N-1}; \\
 -\omega_{0N} \frac{a_{1N}}{h_1^2} - \omega_{1,N-1} \frac{2b_{1N}}{h_2^2} + \omega_{1N} d_{1N} - \omega_{2N} \frac{a_{2N}}{h_1^2} = F_{1N} + \frac{2}{h_2} \psi_{1N}; \\
 \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\
 -\omega_{M-2,N} \frac{a_{M-1,N}}{h_1^2} - \omega_{M-1,N-1} \frac{2b_{M-1,N}}{h_2^2} + \omega_{M-1,N} d_{M-1,N} - \omega_{MN} \frac{a_{MN}}{h_1^2} = \\
 \qquad \qquad \qquad = F_{M-1,N} + \frac{2}{h_2} \psi_{M-1,N}; \\
 -\omega_{00} \frac{a_{10}}{h_1^2} + \omega_{10} e_{10} - \omega_{11} \frac{2b_{11}}{h_2^2} - \omega_{20} \frac{a_{20}}{h_1^2} = F_{10} + \frac{2}{h_2} \psi_{10}; \\
 \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\
 -\omega_{M-2,0} \frac{a_{M-1,0}}{h_1^2} + \omega_{M-1,0} e_{M-1,0} - \omega_{M-1,1} \frac{2b_{M-1,1}}{h_2^2} - \omega_{M0} \frac{a_{M0}}{h_1^2} = \\
 \qquad \qquad \qquad = F_{M-1,0} + \frac{2}{h_2} \psi_{M-1,0}; \\
 -\omega_{M-1,1} \frac{2a_{M1}}{h_1^2} - \omega_{M0} \frac{b_{M1}}{h_2^2} + \omega_{M1} f_{M1} - \omega_{M2} \frac{b_{M2}}{h_2^2} = F_{M1} + \frac{2}{h_1} \psi_{M1}; \\
 \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\
 -\omega_{M-1,N-1} \frac{2a_{M,N-1}}{h_1^2} - \omega_{M,N-2} \frac{b_{M,N-1}}{h_2^2} + \omega_{M,N-1} f_{M,N-1} - \omega_{MN} \frac{b_{MN}}{h_2^2} = \\
 \qquad \qquad \qquad = F_{M,N-1} + \frac{2}{h_1} \psi_{M,N-1}; \\
 -\omega_{00} \frac{b_{01}}{h_2^2} + \omega_{01} g_{01} - \omega_{02} \frac{b_{02}}{h_2^2} - \omega_{11} \frac{2a_{11}}{h_1^2} = F_{01} + \frac{2}{h_1} \psi_{01}; \\
 \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\
 -\omega_{0,N-2} \frac{b_{0,N-1}}{h_2^2} + \omega_{0,N-1} g_{0,N-1} - \omega_{0N} \frac{b_{0N}}{h_2^2} - \omega_{1,N-1} \frac{2a_{1,N-1}}{h_1^2} = F_{0,N-1} + \frac{2}{h_1} \psi_{0,N-1}; \\
 \omega_{00} k_{00} - \omega_{01} \frac{2b_{01}}{h_2^2} - \omega_{10} \frac{2a_{10}}{h_1^2} = F_{00} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{00}; \\
 -\omega_{M-1,0} \frac{2a_{M0}}{h_1^2} + \omega_{M0} \ell_{M0} - \omega_{M1} \frac{2b_{M1}}{h_2^2} = F_{M0} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{M0}; \\
 -\omega_{M-1,N} \frac{2a_{MN}}{h_1^2} - \omega_{M,N-1} \frac{2b_{MN}}{h_2^2} + \omega_{MN} r_{MN} = F_{MN} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{MN}; \\
 -\omega_{0,N-1} \frac{2b_{0N}}{h_2^2} + \omega_{0N} s_{0N} - \omega_{1N} \frac{2a_{1N}}{h_1^2} = F_{0N} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{0N}.
 \end{array} \right.$$

Здесь

$$c_{ij} = \frac{a_{i+1,j} + a_{ij}}{h_1^2} + \frac{b_{i,j+1} + b_{ij}}{h_2^2} + q_{ij}, \quad i = \overline{1, M-1}, j = \overline{1, N-1}; \quad (3.29)$$

$$d_{iN} = \frac{2b_{iN}}{h_2^2} + \frac{a_{i+1,N} + a_{iN}}{h_1^2} + q_{iN}, \quad i = \overline{1, M-1}; \quad (3.30)$$

$$e_{i0} = \frac{2b_{i1}}{h_2^2} + q_{i0} + \frac{a_{i+1,0} + a_{i0}}{h_1^2}, \quad i = \overline{1, M-1}; \quad (3.31)$$

$$f_{Mj} = \frac{2a_{Mj}}{h_1^2} + q_{Mj} + \frac{2}{h_1} + \frac{b_{M,j+1} + b_{Mj}}{h_2^2}, \quad j = \overline{1, N-1}; \quad (3.32)$$

$$g_{0j} = \frac{2a_{1j}}{h_1^2} + q_{0j} + \frac{2}{h_1} + \frac{b_{0,j+1} + b_{0j}}{h_2^2}, \quad j = \overline{1, N-1}; \quad (3.33)$$

$$k_{00} = \frac{2a_{10}}{h_1^2} + \frac{2b_{01}}{h_2^2} + q_{00} + \frac{2}{h_1} + \frac{2}{h_2}; \quad (3.34)$$

$$\ell_{M0} = \frac{2a_{M0}}{h_1^2} + \frac{2b_{M1}}{h_2^2} + q_{M0} + \frac{2}{h_1} + \frac{2}{h_2}; \quad (3.35)$$

$$r_{MN} = \frac{2a_{MN}}{h_1^2} + \frac{2b_{MN}}{h_2^2} + q_{MN} + \frac{2}{h_1} + \frac{2}{h_2}; \quad (3.36)$$

$$s_{0N} = \frac{2a_{1N}}{h_1^2} + \frac{2b_{0N}}{h_2^2} + q_{0N} + \frac{2}{h_1} + \frac{2}{h_2}. \quad (3.37)$$

Теперь представим СЛАУ (3.28) в записе матричного вида $A\omega = B$. Здесь $A \in \mathbb{R}^{(N+1)(M+1) \times (N+1)(M+1)}$, $\omega \in \mathbb{R}^{(N+1)(M+1) \times 1}$, $B \in \mathbb{R}^{(N+1)(M+1) \times 1}$.

Замечание 3. Сеточную функцию ω_{ij} , $i = \overline{0, M}$, $j = \overline{0, N}$ вытjнем в вектор:

$$\vec{\omega}^T = (\omega_{00}, \omega_{01}, \dots, \omega_{0N}, \omega_{10}, \omega_{11}, \dots, \omega_{1N}, \dots, \dots, \omega_{M0}, \omega_{M1}, \dots, \omega_{MN}) \quad (3.38)$$

Определим следующие матрицы A_i , B_i , $C_i \in \mathbb{R}^{(N-1) \times (N+1)}$:

$$A_i = \begin{pmatrix} 0 & -\frac{a_{i1}}{h_1^2} & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & -\frac{a_{i2}}{h_1^2} & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{a_{i3}}{h_1^2} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & -\frac{a_{i,N-1}}{h_1^2} & 0 \end{pmatrix}, \quad i = \overline{1, M-1}; \quad (3.39)$$

$$B_i = \begin{pmatrix} -\frac{b_{i1}}{h_2^2} & c_{i1} & -\frac{b_{i2}}{h_2^2} & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\frac{b_{i2}}{h_2^2} & c_{i2} & -\frac{b_{i3}}{h_2^2} & \cdots & 0 & 0 & 0 \\ 0 & 0 & -\frac{b_{i3}}{h_2^2} & c_{i3} & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & -\frac{b_{i,N-1}}{h_2^2} & c_{i,N-1} & -\frac{b_{iN}}{h_2^2} \end{pmatrix}, \quad i = \overline{1, M-1}; \quad (3.40)$$

$$C_i = \begin{pmatrix} 0 & -\frac{a_{i+1,1}}{h_1^2} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & -\frac{a_{i+1,2}}{h_1^2} & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{a_{i+1,3}}{h_1^2} & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{a_{i+1,N-1}}{h_1^2} & 0 \end{pmatrix}, \quad i = \overline{1, M-1}. \quad (3.41)$$

Тогда матрица, характеризующая область действия Лапласиана (внутренние точки области, первые $(N-1)(M-1)$ уравнений (3.28)) имеет вид

$$A_1) = \begin{pmatrix} A_1 & B_1 & C_1 & \Theta & \Theta & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & A_2 & B_2 & C_2 & \Theta & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & A_3 & B_3 & C_3 & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & A_4 & B_4 & C_4 & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & A_5 & B_5 & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & A_6 & \cdots & \Theta & \Theta & \Theta & \Theta \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \cdots & A_{M-2} & B_{M-2} & C_{M-2} & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \cdots & \Theta & A_{M-1} & B_{M-1} & C_{M-1} \end{pmatrix}, \quad (3.42)$$

где Θ — нулевые матрицы размерами $(N-1) \times (N+1)$.

Аналогично, найдём матрицы A_k), $k = \overline{2, 9}$, которые соответствуют оставшимся 8 уравнениям системы (3.26), представленной в расписанном виде (3.28). Таким образом, мы получим 9 матриц $A_1), \dots, A_9)$ с фиксированным количеством столбцов $(N+1)(M+1)$. Итоговый вид матрицы A , соответствующий СЛАУ (3.28) будет иметь вид

$$A = \begin{pmatrix} A_1) \\ A_2) \\ A_3) \\ A_4) \\ A_5) \\ A_6) \\ A_7) \\ A_8) \\ A_9) \end{pmatrix}. \quad (3.43)$$

Определим матрицу A_2). Пусть

$$\psi_{T_1 i} = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & 0 & -\frac{a_{iN}}{h_1^2} \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}, \quad i = \overline{1, M-1}; \quad (3.44)$$

$$\psi_{T_2 i} = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{2b_{iN}}{h_2^2} & d_{iN} \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}, \quad i = \overline{1, M-1}; \quad (3.45)$$

$$\psi_{T_3 i} = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & 0 & -\frac{a_{i+1,N}}{h_1^2} \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}, \quad i = \overline{1, M-1}. \quad (3.46)$$

Тогда

$$A_2) = \begin{pmatrix} \psi_{T_1 1} & \psi_{T_2 1} & \psi_{T_3 1} & \Theta & \Theta & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \psi_{T_1 2} & \psi_{T_2 2} & \psi_{T_3 2} & \Theta & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \psi_{T_1 3} & \psi_{T_2 3} & \psi_{T_3 3} & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \psi_{T_1 4} & \psi_{T_2 4} & \psi_{T_3 4} & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \psi_{T_1 5} & \psi_{T_2 5} & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \psi_{T_1 6} & \cdots & \Theta & \Theta & \Theta & \Theta \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \cdots & \psi_{T_1 M-2} & \psi_{T_2 M-2} & \psi_{T_3 M-2} & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \cdots & \Theta & \psi_{T_1 M-1} & \psi_{T_2 M-1} & \psi_{T_3 M-1} \end{pmatrix}, \quad (3.47)$$

где Θ — нулевые матрицы размерами $1 \times (N+1)$.

Определим матрицу A_3). Пусть

$$\psi_{B_1 i} = \begin{pmatrix} -\frac{a_{i0}}{h_1^2} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}, \quad i = \overline{1, M-1}; \quad (3.48)$$

$$\psi_{B_2 i} = \begin{pmatrix} e_{i0} & -\frac{2b_{i1}}{h_2^2} & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}, \quad i = \overline{1, M-1}; \quad (3.49)$$

$$\psi_{B_3 i} = \begin{pmatrix} -\frac{a_{i+1,0}}{h_1^2} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}, \quad i = \overline{1, M-1}. \quad (3.50)$$

Тогда

$$A_3) = \begin{pmatrix} \psi_{B_1 1} & \psi_{B_2 1} & \psi_{B_3 1} & \Theta & \Theta & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \psi_{B_1 2} & \psi_{B_2 2} & \psi_{B_3 2} & \Theta & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \psi_{B_1 3} & \psi_{B_2 3} & \psi_{B_3 3} & \Theta & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \psi_{B_1 4} & \psi_{B_2 4} & \psi_{B_3 4} & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \psi_{B_1 5} & \psi_{B_2 5} & \cdots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \psi_{B_1 6} & \cdots & \Theta & \Theta & \Theta & \Theta \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \cdots & \psi_{B_1 M-2} & \psi_{B_2 M-2} & \psi_{B_3 M-2} & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \cdots & \Theta & \psi_{B_1 M-1} & \psi_{B_2 M-1} & \psi_{B_3 M-1} \end{pmatrix}, \quad (3.51)$$

где Θ — нулевые матрицы размерами $1 \times (N + 1)$.

Определим матрицу A_4). Пусть

$$\psi_{R_{M-1}} = \begin{pmatrix} 0 & -\frac{2a_{M1}}{h_1^2} & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & -\frac{2a_{M2}}{h_1^2} & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{2a_{M3}}{h_1^2} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & -\frac{2a_{M,N-1}}{h_1^2} & 0 \end{pmatrix} \in \mathbb{R}^{(N-1) \times (N+1)}; \quad (3.52)$$

$$\psi_{R_M} = \begin{pmatrix} -\frac{b_{M1}}{h_2^2} & f_{M1} & -\frac{b_{M2}}{h_2^2} & 0 & \dots & 0 & 0 & 0 \\ 0 & -\frac{b_{M2}}{h_2^2} & f_{M2} & -\frac{b_{M3}}{h_2^2} & \dots & 0 & 0 & 0 \\ 0 & 0 & -\frac{b_{M1}}{h_2^2} & f_{M1} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & -\frac{b_{M,N-1}}{h_2^2} & f_{M,N-1} & -\frac{b_{MN}}{h_2^2} \end{pmatrix} \in \mathbb{R}^{(N-1) \times (N+1)}. \quad (3.53)$$

Тогда

$$A_4) = (\Theta \quad \psi_{R_{M-1}} \quad \psi_{R_M}), \quad (3.54)$$

где Θ — нулевая матрица размером $(N - 1) \times (M - 1)(N + 1)$.

Определим матрицу A_5). Пусть

$$\psi_{L_0} = \begin{pmatrix} -\frac{b_{01}}{h_2^2} & g_{01} & -\frac{b_{02}}{h_2^2} & 0 & \dots & 0 & 0 & 0 \\ 0 & -\frac{b_{02}}{h_2^2} & g_{02} & -\frac{b_{03}}{h_2^2} & \dots & 0 & 0 & 0 \\ 0 & 0 & -\frac{b_{01}}{h_2^2} & g_{01} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & -\frac{b_{0,N-1}}{h_2^2} & g_{0,N-1} & -\frac{b_{0N}}{h_2^2} \end{pmatrix} \in \mathbb{R}^{(N-1) \times (N+1)}; \quad (3.55)$$

$$\psi_{L_1} = \begin{pmatrix} 0 & -\frac{2a_{11}}{h_1^2} & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & -\frac{2a_{12}}{h_1^2} & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{2a_{13}}{h_1^2} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & -\frac{2a_{1,N-1}}{h_1^2} & 0 \end{pmatrix} \in \mathbb{R}^{(N-1) \times (N+1)}; \quad (3.56)$$

Тогда

$$A_5) = (\psi_{L_0} \quad \psi_{L_1} \quad \Theta), \quad (3.57)$$

где Θ — нулевая матрица размером $(N - 1) \times (M - 1)(N + 1)$.

Определим матрицу A_6). Пусть

$$P_{00}^0 = \begin{pmatrix} k_{00} & -\frac{2b_{01}}{h_2^2} & 0 & 0 & \dots & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.58)$$

$$P_{00}^1 = \begin{pmatrix} -\frac{2a_{10}}{h_1^2} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.59)$$

Тогда

$$A_6) = (P_{00}^0 \ P_{00}^1 \ \Theta), \quad (3.60)$$

где Θ — нулевая матрица размером $1 \times (M-1)(N+1)$.

Определим матрицу A_7). Пусть

$$P_{40}^{M-1} = \begin{pmatrix} -\frac{2a_{M0}}{h_1^2} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.61)$$

$$P_{40}^M = \begin{pmatrix} \ell_{M0} & -\frac{2b_{M1}}{h_2^2} & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.62)$$

Тогда

$$A_7) = (\Theta \ P_{40}^{M-1} \ P_{40}^M), \quad (3.63)$$

где Θ — нулевая матрица размером $1 \times (M-1)(N+1)$.

Определим матрицу A_8). Пусть

$$P_{43}^{M-1} = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & 0 & -\frac{2a_{MN}}{h_1^2} \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.64)$$

$$P_{43}^M = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{2b_{MN}}{h_2^2} & r_{MN} \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.65)$$

Тогда

$$A_8) = (\Theta \ P_{43}^{M-1} \ P_{43}^M), \quad (3.66)$$

где Θ — нулевая матрица размером $1 \times (M-1)(N+1)$.

Определим матрицу A_9). Пусть

$$P_{03}^0 = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{2b_{0N}}{h_2^2} & s_{0N} \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.67)$$

$$P_{03}^1 = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & 0 & -\frac{2a_{1N}}{h_1^2} \end{pmatrix} \in \mathbb{R}^{1 \times (N+1)}; \quad (3.68)$$

Тогда

$$A_9) = (P_{03}^0 \ P_{03}^1 \ \Theta), \quad (3.69)$$

где Θ — нулевая матрица размером $1 \times (M-1)(N+1)$.

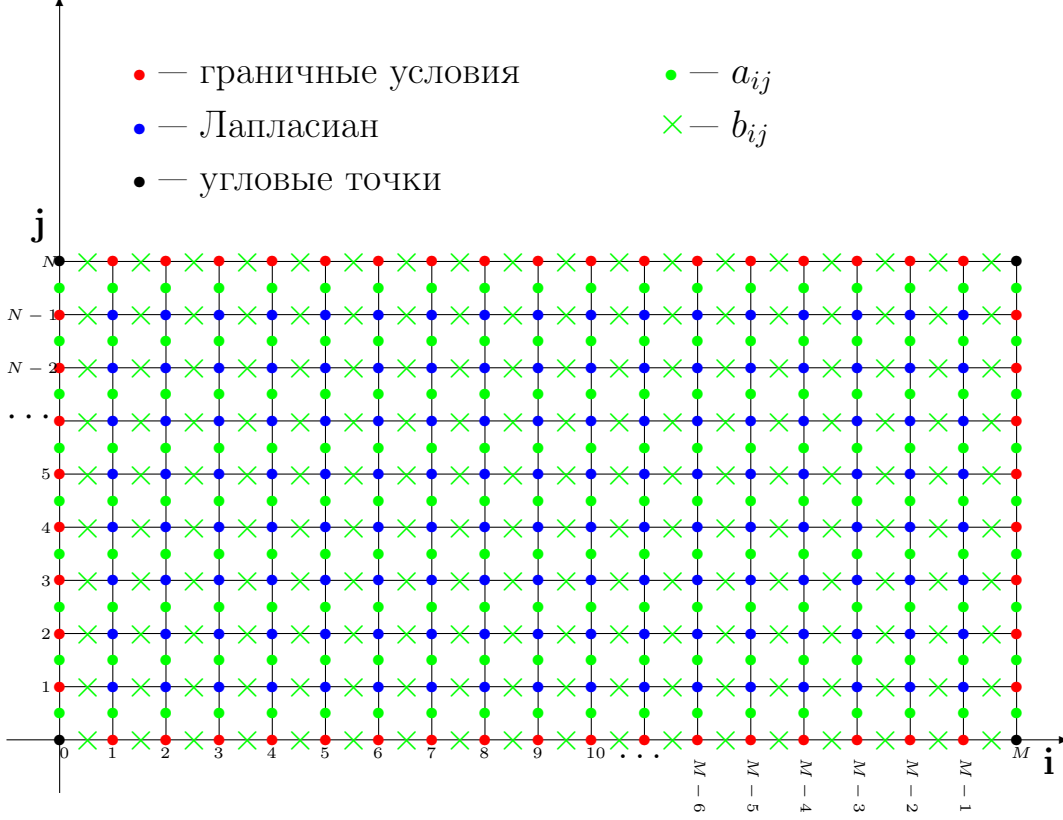
Выпишем итоговую матрицу A . Пусть $\hat{\Theta}$ обозначает нулевую матрицу размерами $(N-1) \times (N+1)$, а Θ обозначает нулевую матрицу размерами $1 \times (N+1)$. Утомительное объединение матриц $A_1, \dots, A_9$ позволяет получить следующий вид матрицы A :

$$A = \begin{pmatrix} A_1 & B_1 & C_1 & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \hat{\Theta} & A_2 & B_2 & C_2 & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \hat{\Theta} & \hat{\Theta} & A_3 & B_3 & C_3 & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & A_4 & B_4 & C_4 & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & A_5 & B_5 & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & A_6 & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & A_{M-2} & B_{M-2} & C_{M-2} & \hat{\Theta} \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & A_{M-1} & B_{M-1} & C_{M-1} \\ \psi_{T_1 1} & \psi_{T_2 1} & \psi_{T_3 1} & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \psi_{T_1 2} & \psi_{T_2 2} & \psi_{T_3 2} & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \psi_{T_1 3} & \psi_{T_2 3} & \psi_{T_3 3} & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \psi_{T_1 4} & \psi_{T_2 4} & \psi_{T_3 4} & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \psi_{T_1 5} & \psi_{T_2 5} & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \psi_{T_1 6} & \dots & \Theta & \Theta & \Theta & \Theta \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \psi_{T_1 M-2} & \psi_{T_2 M-2} & \psi_{T_3 M-2} & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \psi_{T_1 M-1} & \psi_{T_2 M-1} & \psi_{T_3 M-1} \\ \psi_{B_1 1} & \psi_{B_2 1} & \psi_{B_3 1} & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \psi_{B_1 2} & \psi_{B_2 2} & \psi_{B_3 2} & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \psi_{B_1 3} & \psi_{B_2 3} & \psi_{B_3 3} & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \psi_{B_1 4} & \psi_{B_2 4} & \psi_{B_3 4} & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \psi_{B_1 5} & \psi_{B_2 5} & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \psi_{B_1 6} & \dots & \Theta & \Theta & \Theta & \Theta \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \psi_{B_1 M-2} & \psi_{B_2 M-2} & \psi_{B_3 M-2} & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \psi_{B_1 M-1} & \psi_{B_2 M-1} & \psi_{B_3 M-1} \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \psi_{R_{M-1}} & \psi_{R_M} \\ \psi_{L_0} & \psi_{L_1} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ P_{00}^0 & P_{00}^1 & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & P_{40}^{M-1} & P_{40}^M \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & P_{43}^{M-1} & P_{43}^M \\ P_{03}^0 & P_{03}^1 & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \end{pmatrix}. \quad (3.70)$$

Осталось выписать матрицу B . Пусть $\hat{F}_i = (F_{i1}, \dots, F_{i,N-1})^T$, $i = \overline{1, M-1}$. Пусть, также, $\hat{\psi}_M = (\psi_{M1}, \dots, \psi_{M,N-1})^T$; $\hat{\psi}_0 = (\psi_{01}, \dots, \psi_{0,N-1})^T$. Тогда матрица B имеет следующий вид:

$$B = \begin{pmatrix} \hat{F}_1 \\ \hat{F}_2 \\ \hat{F}_3 \\ \hat{F}_4 \\ \hat{F}_5 \\ \hat{F}_6 \\ \dots \\ \hat{F}_{M-2} \\ \hat{F}_{M-1} \\ F_{1N} + \frac{2}{h_2} \psi_{1N} \\ F_{2N} + \frac{2}{h_2} \psi_{2N} \\ F_{3N} + \frac{2}{h_2} \psi_{3N} \\ F_{4N} + \frac{2}{h_2} \psi_{4N} \\ F_{5N} + \frac{2}{h_2} \psi_{5N} \\ F_{6N} + \frac{2}{h_2} \psi_{6N} \\ \dots \\ F_{M-2,N} + \frac{2}{h_2} \psi_{M-2,N} \\ F_{M-1,N} + \frac{2}{h_2} \psi_{M-1,N} \\ F_{10} + \frac{2}{h_2} \psi_{10} \\ F_{20} + \frac{2}{h_2} \psi_{20} \\ F_{30} + \frac{2}{h_2} \psi_{30} \\ F_{40} + \frac{2}{h_2} \psi_{40} \\ F_{50} + \frac{2}{h_2} \psi_{50} \\ F_{60} + \frac{2}{h_2} \psi_{60} \\ \dots \\ F_{M-2,0} + \frac{2}{h_2} \psi_{M-2,0} \\ F_{M-1,0} + \frac{2}{h_2} \psi_{M-1,0} \\ \hat{F}_M + \frac{2}{h_1} \hat{\psi}_M \\ \hat{F}_0 + \frac{2}{h_1} \hat{\psi}_0 \\ F_{00} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{00} \\ F_{M0} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{M0} \\ F_{MN} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{MN} \\ F_{0N} + \left(\frac{2}{h_1} + \frac{2}{h_2} \right) \psi_{0N} \end{pmatrix}. \quad (3.71)$$

Формально, сетка обретает следующий вид



4 Метод решения СЛАУ

Точным решением СЛАУ $A\omega = B$ является вектор $\omega = A^{-1}B$. Однако, из-за больших размеров матрицы A поиск обратной крайне затруднителен, поэтому необходим алгоритм, позволяющий решить СЛАУ приближённо. Приближённое решение СЛАУ $A\omega = B$ может быть получено итерационным методом наименьших невязок. Этот метод позволяет получить последовательность сеточных функций $\omega^{(k)} \in H$, $k = 1, 2, \dots$, сходящуюся по норме пространства H к решению разностной схемы, т.е.

$$\left\| \omega - \omega^{(k)} \right\|_E \rightarrow 0, \quad k \rightarrow \infty. \quad (4.1)$$

Начальное приближение $\omega^{(0)}$ можно выбрать любым способом, например, равным нулю во всех точках расчётной сетки.

Итерационным методом является одношаговым. Итерация $\omega^{(k+1)}$ вычисляется по итерации $\omega^{(k)}$ в соответствии с

$$\omega_{ij}^{(k+1)} = \omega_{ij}^{(k)} - \tau_{k+1} \gamma_{ij}^{(k)}, \quad (4.2)$$

где невязка $\gamma^{(k)} = A\omega^{(k)} - B$, итерационный параметр

$$\tau_{k+1} = \frac{\langle A\gamma^{(k)}, \gamma^{(k)} \rangle}{\|A\gamma^{(k)}\|_E^2}. \quad (4.3)$$

Критерием остановки итерационного алгоритма является выполнение неравенства

$$\left\| \omega^{(k+1)} - \omega^{(k)} \right\|_E \leq \varepsilon, \quad (4.4)$$

где $\varepsilon > 0$ — фиксированная точность итерационного метода.

Для того, чтобы представленный итерационный метод работал корректно, необходимо переставить блоки матрицы (3.70) так, чтобы имело место диагональное преобладание:

$$A = \begin{pmatrix} P_{00}^0 & P_{00}^1 & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \psi_{L_0} & \psi_{L_1} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ P_{03}^0 & P_{03}^1 & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \psi_{B_{11}} & \psi_{B_{21}} & \psi_{B_{31}} & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ A_1 & B_1 & C_1 & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \psi_{T_{11}} & \psi_{T_{21}} & \psi_{T_{31}} & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \psi_{B_{12}} & \psi_{B_{22}} & \psi_{B_{32}} & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \hat{\Theta} & A_2 & B_2 & C_2 & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \Theta & \psi_{T_{12}} & \psi_{T_{22}} & \psi_{T_{32}} & \Theta & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \psi_{B_{13}} & \psi_{B_{23}} & \psi_{B_{33}} & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \hat{\Theta} & \hat{\Theta} & A_3 & B_3 & C_3 & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \Theta & \Theta & \psi_{T_{13}} & \psi_{T_{23}} & \psi_{T_{33}} & \Theta & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \psi_{B_{14}} & \psi_{B_{24}} & \psi_{B_{34}} & \dots & \Theta & \Theta & \Theta & \Theta \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & A_4 & B_4 & C_4 & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \Theta & \Theta & \Theta & \psi_{T_{14}} & \psi_{T_{24}} & \psi_{T_{34}} & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \psi_{B_{15}} & \psi_{B_{25}} & \dots & \Theta & \Theta & \Theta & \Theta \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & A_5 & B_5 & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \Theta & \Theta & \Theta & \Theta & \psi_{T_{15}} & \psi_{T_{25}} & \dots & \Theta & \Theta & \Theta & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \psi_{B_{16}} & \dots & \Theta & \Theta & \Theta & \Theta \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & A_6 & \dots & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} \\ \Theta & \Theta & \Theta & \Theta & \Theta & \psi_{T_{16}} & \dots & \Theta & \Theta & \Theta & \Theta \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \psi_{B_{1M-2}} & \psi_{B_{2M-2}} & \psi_{B_{3M-2}} & \Theta \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & A_{M-2} & B_{M-2} & C_{M-2} & \hat{\Theta} \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \psi_{T_{1M-2}} & \psi_{T_{2M-2}} & \psi_{T_{3M-2}} & \Theta \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \psi_{B_{1M-1}} & \psi_{B_{2M-1}} & \psi_{B_{3M-1}} \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & A_{M-1} & B_{M-1} & C_{M-1} \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \psi_{T_{1M-1}} & \psi_{T_{2M-1}} & \psi_{T_{3M-1}} \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & P_{40}^{M-1} & P_{40}^M \\ \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \hat{\Theta} & \dots & \hat{\Theta} & \hat{\Theta} & \psi_{R_{M-1}} & \psi_{R_M} \\ \Theta & \Theta & \Theta & \Theta & \Theta & \Theta & \dots & \Theta & \Theta & P_{43}^{M-1} & P_{43}^M \end{pmatrix}. \quad (4.5)$$

В связи с этим, матрица B будет иметь вид

$$B = \begin{pmatrix} F_{00} + \left(\frac{2}{h_1} + \frac{2}{h_2}\right) \psi_{00} \\ \hat{F}_0 + \frac{2}{h_1} \hat{\psi}_0 \\ F_{0N} + \left(\frac{2}{h_1} + \frac{2}{h_2}\right) \psi_{0N} \\ F_{10} + \frac{2}{h_2} \psi_{10} \\ \hat{F}_1 \\ F_{1N} + \frac{2}{h_2} \psi_{1N} \\ F_{20} + \frac{2}{h_2} \psi_{20} \\ \hat{F}_2 \\ F_{2N} + \frac{2}{h_2} \psi_{2N} \\ F_{30} + \frac{2}{h_2} \psi_{30} \\ \hat{F}_3 \\ F_{3N} + \frac{2}{h_2} \psi_{3N} \\ F_{40} + \frac{2}{h_2} \psi_{40} \\ \hat{F}_4 \\ F_{4N} + \frac{2}{h_2} \psi_{4N} \\ F_{50} + \frac{2}{h_2} \psi_{50} \\ \hat{F}_5 \\ F_{5N} + \frac{2}{h_2} \psi_{5N} \\ F_{60} + \frac{2}{h_2} \psi_{60} \\ \hat{F}_6 \\ F_{6N} + \frac{2}{h_2} \psi_{6N} \\ \dots \\ F_{M-2,0} + \frac{2}{h_2} \psi_{M-2,0} \\ \hat{F}_{M-2} \\ F_{M-2,N} + \frac{2}{h_2} \psi_{M-2,N} \\ F_{M-1,0} + \frac{2}{h_2} \psi_{M-1,0} \\ \hat{F}_{M-1} \\ F_{M-1,N} + \frac{2}{h_2} \psi_{M-1,N} \\ F_{M0} + \left(\frac{2}{h_1} + \frac{2}{h_2}\right) \psi_{M0} \\ \hat{F}_M + \frac{2}{h_1} \hat{\psi}_M \\ F_{MN} + \left(\frac{2}{h_1} + \frac{2}{h_2}\right) \psi_{MN} \end{pmatrix}. \quad (4.6)$$

5 Численная реализация на сгущающихся сетках

Последовательный код программы находится в Приложении 1.

5.1 $M = 20, N = 20$

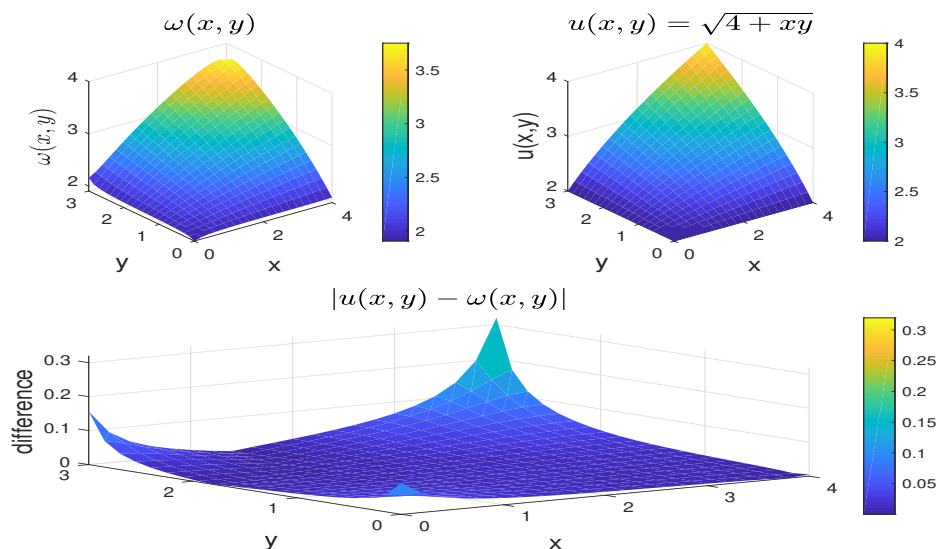


Рис. 1: Число итераций = 3884. $\varepsilon = 10^{-6}$.

5.2 $M = 40, N = 40$

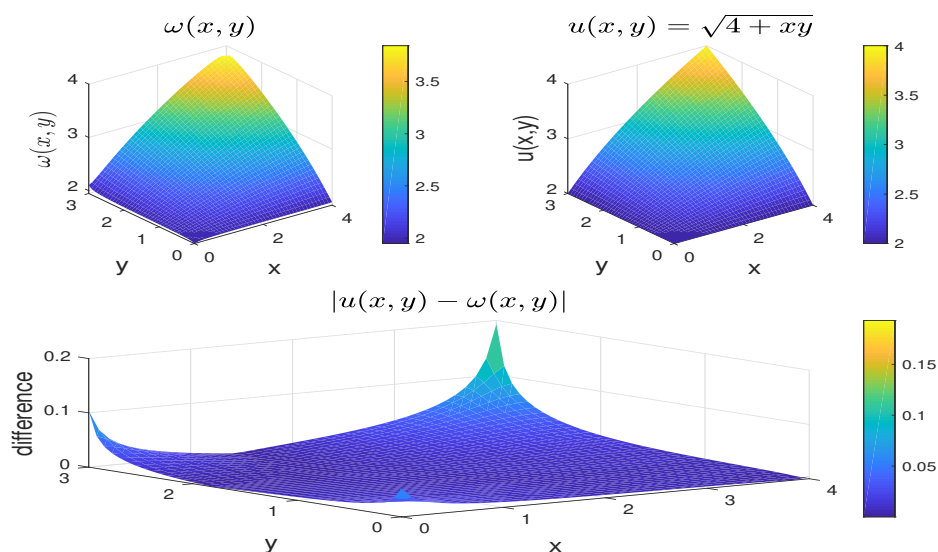


Рис. 2: Число итераций = 13786. $\varepsilon = 10^{-6}$.

5.3 $M = 80, N = 80$

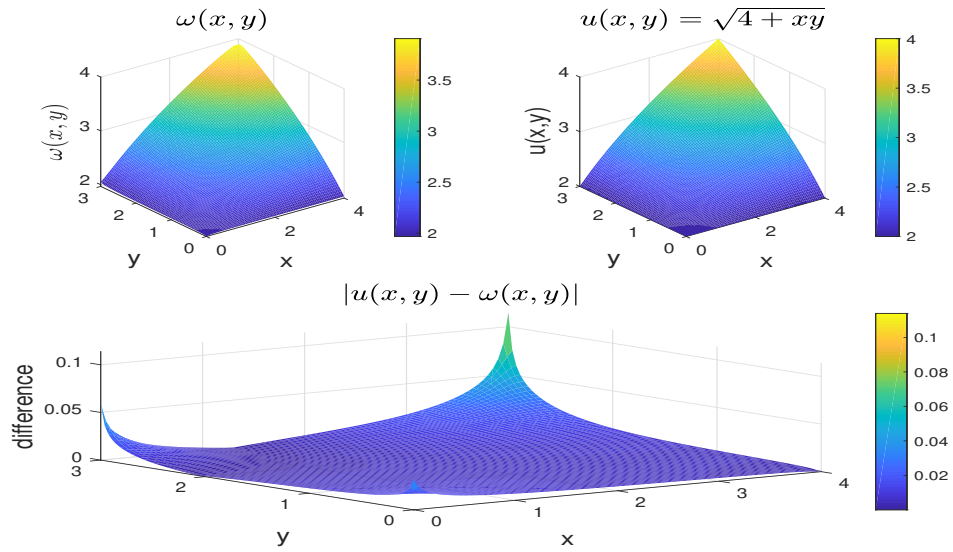


Рис. 3: Число итераций = 46974. $\varepsilon = 10^{-6}$.

5.4 $M = 160, N = 160$

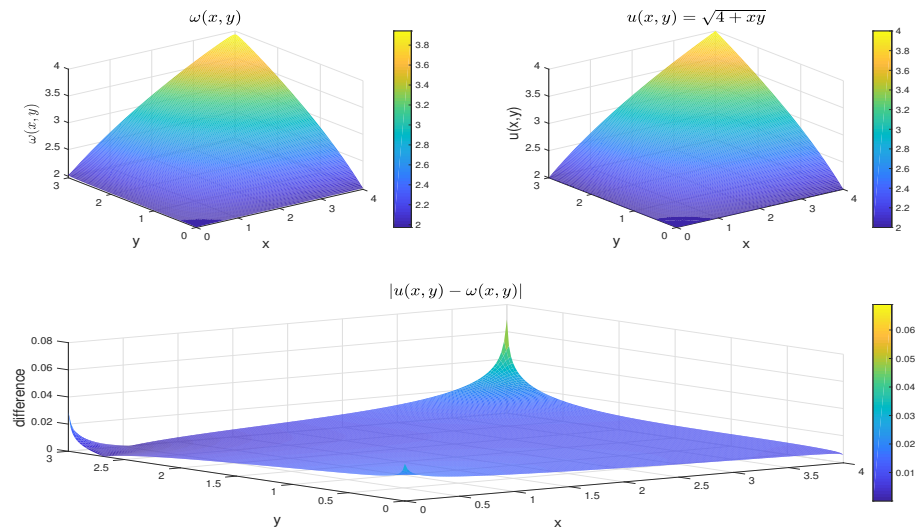


Рис. 4: Число итераций = 152258. $\varepsilon = 10^{-6}$.

5.5 Приложение 1

Код программы. Чтобы эффективно затрачивать вычислительные ресурсы, достаточно знать не всю матрицу A целиком, а только ненулевые элементы её блоков. Так, например, для блоков A_i нужно знать не $(N-1) \times (N+1)$ элементов, а всего лишь $N-1$.

```
1  #include <iostream>
2  #include <cmath>
3  #include <vector>
4  #include <fstream>
5  #include <sys/time.h>
6  #include <mpi.h>
7
8  using namespace std;
9
10
11 double Scalar(vector<double> uVec, vector<double> vVec, int M, int N, double h1, double h2)
12 {
13     double rho1, rho2, rho, res;
14     for(int i = 0; i < M+1; i++)
15     {
16         for(int j = 0; j < N+1; j++)
17         {
18             if((i == 0) || (i == M))
19                 rho1 = 0.5;
20             else
21                 rho1 = 1;
22
23             if((j == 0) || (j == N))
24                 rho2 = 0.5;
25             else
26                 rho2 = 1;
27
28             rho = rho1*rho2;
29             res += rho*uVec[i*(N+1) + j]*vVec[i*(N+1) + j];
30         }
31     }
32     res *= h1*h2;
33
34     return res;
35 }
36
37
38 int main (int argc, char* argv[])
39 {
40     cout << "START\n";
41     int M = 160;
42     int N = 160;
43     double eps = pow(10,-6);
44
45     struct timeval start;
46     struct timeval stop;
47     double time;
48     const double micro = 1.0e-06;
49
50     gettimeofday(&start, NULL);
51
52     double h1 = (double)4/M; //+
53     double h2 = (double)3/M; //+
54
55     vector<double> xVec; //+
56     vector<double> yVec; //+
57     vector<double> x_shiftVec;
58     vector<double> y_shiftVec;
59     vector<double> wVec;
60     vector<double> w_newVec;
61     vector<double> w_diffVec;
62
63     vector<vector<double> > FMat(N+1); //+
64     vector<vector<double> > qMat(N+1); //+
65
66     vector<vector<double> > aMat(N+1); //+
67     vector<vector<double> > bMat(N+1); //+
```

```

68     vector<vector<double>> > cMat(N+1);
69
70     vector<double> dVec;
71     vector<double> eVec;
72     vector<double> fVec;
73     vector<double> gVec;
74
75     vector<double> gammaTopVec; //+
76     vector<double> gammaBottomVec; //+
77     vector<double> gammaLeftVec; //+
78     vector<double> gammaRightVec; //+
79
80     double psi_00, psi_M0, psi_MN, psi_ON;
81     double k_00, ell_M0, r_MN, s_ON;
82
83     vector<double> BVec; //+
84
85     vector<double> AwVec;
86     vector<double> gammaVec;
87     vector<double> AgammaVec;
88     double tau, numerator, denominator, norm_val;
89
90     //Initialize w
91     for(int i = 0; i < (N+1)*(M+1); i++)
92     {
93         wVec.push_back(0);
94         AwVec.push_back(0);
95         gammaVec.push_back(0);
96         AgammaVec.push_back(0);
97         w_newVec.push_back(0);
98         w_diffVec.push_back(0);
99     }
100
101     for(int i = 0; i < M+1; i++) //CYCLE FROM 0 TO M. REMEMBER.
102         xVec.push_back(i*h1);
103     for(int j = 0; j < N+1; j++)
104         yVec.push_back(j*h2);
105
106     for(int i = 1; i < M+1; i++)
107         x_shiftVec.push_back((i-0.5)*h1);
108     for(int j = 1; j < N+1; j++)
109         y_shiftVec.push_back((j-0.5)*h2);
110
111     for(int j = 0; j < N+1; j++)
112     {
113         // kMat[j].resize(M+1);
114         qMat[j].resize(M+1);
115         FMat[j].resize(M+1);
116         aMat[j].resize(M+1);
117         bMat[j].resize(M+1);
118
119         for(int i = 0; i < M+1; i++)
120         {
121             qMat[j][i] = xVec[i] + yVec[j];
122             FMat[j][i] = 1.0/4.0*(4 + xVec[i] + yVec[j])/(pow(4 + xVec[i]*yVec[j],3.0/2.0))*
123             (pow(xVec[i],2.0) + pow(yVec[j],2.0)) - 1/(2*pow(4 + xVec[i]*yVec[j],0.5))*
124             (xVec[i] + yVec[j]) + (xVec[i] + yVec[j])*pow(4 + xVec[i]*yVec[j],0.5);
125
126             if((i > 0))
127                 aMat[j][i] = 4 + x_shiftVec[i-1] + yVec[j];
128             else
129                 aMat[j][i] = 0;
130
131             if((j > 0))
132                 bMat[j][i] = 4 + xVec[i] + y_shiftVec[j-1];
133             else
134                 bMat[j][i] = 0;
135
136             if((j == 0)&&(i > 0)&&(i < M))
137             {
138                 gammaTopVec.push_back((7 + xVec[i])*xVec[i]/(2*sqrt(4 + 3*xVec[i])));
139                 gammaBottomVec.push_back(-(4 + xVec[i])*xVec[i]/4.0);
140             }
141         }
142     }

```

```

143         if((j > 0)&&(j < N))
144         {
145             gammaLeftVec.push_back(-(4 + yVec[j])*yVec[j]/4.0 + 2);
146             gammaRightVec.push_back((8 + yVec[j])*yVec[j]/(4*sqrt(1 + yVec[j])) +
147             2*sqrt(1 + yVec[j]));
148         }
149     }
150
151
152     psi_00 = 2*gammaBottomVec[0]/h1 + 2*gammaLeftVec[0]/h2;
153     psi_M0 = 2*gammaBottomVec[M-1]/h1 + 2*gammaRightVec[0]/h2;
154     psi_MN = 2*gammaTopVec[0]/h1 + 2*gammaLeftVec[N-1]/h2;
155     psi_ON = 2*gammaTopVec[M-1]/h1 + 2*gammaRightVec[N-1]/h2;
156
157
158     //Initialize B
159     BVec.push_back(FMat[0][0] + psi_00);
160     for(int j = 1; j < N; j++)
161         BVec.push_back(FMat[j][0] + 2/h1 * gammaLeftVec[j-1]);
162     BVec.push_back(FMat[N][0] + psi_ON);
163     for(int i = 1; i < M; i++)
164     {
165         BVec.push_back(FMat[0][i] + 2/h2 * gammaBottomVec[i-1]);
166         for(int j = 1; j < N; j++)
167             BVec.push_back(FMat[j][i]);
168         BVec.push_back(FMat[N][i] + 2/h2 * gammaTopVec[i-1]);
169     }
170
171     BVec.push_back(FMat[0][M] + psi_M0);
172     for(int j = 1; j < N; j++)
173         BVec.push_back(FMat[j][M] + 2/h1 * gammaRightVec[j-1]);
174     BVec.push_back(FMat[N][M] + psi_MN);
175
176
177
178
179
180     //Initialize c
181     for(int j = 0; j < N+1; j++)
182     {
183         cMat[j].resize(M+1);
184         for(int i = 0; i < M+1; i++)
185         {
186             if((i > 0)&&(i < M)&&(j > 0)&&(j < N))
187                 cMat[j][i] = (aMat[j][i+1] + aMat[j][i])/(pow(h1,2)) +
188                 (bMat[j+1][i] + bMat[j][i])/(pow(h2,2)) + qMat[j][i];
189             else
190                 cMat[j][i] = 0;
191         }
192     }
193
194
195
196     //Initialize d,e
197     for(int i = 0; i < M+1; i++)
198     {
199         if((i > 0)&&(i < M))
200         {
201             dVec.push_back((2*bMat[N][i])/(pow(h2,2)) +
202             (aMat[N][i+1] + aMat[N][i])/(pow(h1,2)) + qMat[N][i]);
203             eVec.push_back((2*bMat[1][i])/(pow(h2,2)) + qMat[0][i] +
204             (aMat[0][i+1] + aMat[0][i])/(pow(h1,2)));
205         }
206         else
207         {
208             dVec.push_back(0);
209             eVec.push_back(0);
210         }
211     }
212
213
214
215     //Initialize f,g
216     for(int j = 0; j < N+1; j++)
217     {

```

```

218         if((j > 0)&&(j < N))
219         {
220             fVec.push_back((2*aMat[j][M])/(pow(h1,2)) + qMat[j][M] + 2/h1 +
221                 (bMat[j+1][M] + bMat[j][M])/(pow(h2,2)));
222             gVec.push_back((2*aMat[j][1])/(pow(h1,2)) + qMat[j][0] + 2/h1 +
223                 (bMat[j+1][0] + bMat[j][0])/(pow(h2,2)));
224         }
225         else
226         {
227             fVec.push_back(0);
228             gVec.push_back(0);
229         }
230     }
231
232
233
234     //Initialize k,ell,r,s
235     k_00 = 2*aMat[0][1]/(pow(h1,2)) + 2*bMat[1][0]/(pow(h2,2)) +
236     qMat[0][0] + 2/h1 + 2/h2;
237     ell_M0 = 2*aMat[0][M]/(pow(h1,2)) + 2*bMat[1][M]/(pow(h2,2)) +
238     qMat[0][M] + 2/h1 + 2/h2;
239     r_MN = 2*aMat[N][M]/(pow(h1,2)) + 2*bMat[N][M]/(pow(h2,2)) +
240     qMat[N][M] + 2/h1 + 2/h2;
241     s_ON = 2*aMat[N][1]/(pow(h1,2)) + 2*bMat[N][0]/(pow(h2,2)) +
242     qMat[N][0] + 2/h1 + 2/h2;
243
244     //-----A_1), A_4), A_5)-----
245     vector< vector< vector<double> > > A_block;
246     vector< vector< vector<double> > > B_block;
247     vector< vector< vector<double> > > C_block;
248
249     vector< vector<double> > psiR_Mmin1;
250     vector< vector<double> > psiR_M;
251     vector< vector<double> > psiL0;
252     vector< vector<double> > psiL1;
253     for(int block = 0; block < M-1; block++)
254     {
255         vector< vector<double> > A_i;
256         vector< vector<double> > B_i;
257         vector< vector<double> > C_i;
258         A_block.push_back(A_i);
259         B_block.push_back(B_i);
260         C_block.push_back(C_i);
261         for(int i = 0; i < N-1; i++)
262         {
263             vector<double> A_ij;
264             vector<double> B_ij;
265             vector<double> C_ij;
266             A_block[block].push_back(A_ij);
267             B_block[block].push_back(B_ij);
268             C_block[block].push_back(C_ij);
269
270             if(block == 0) //PSI
271             {
272                 vector<double> psiR_Mmin1_Elem;
273                 vector<double> psiR_M_Elem;
274                 vector<double> psiL0_Elem;
275                 vector<double> psiL1_Elem;
276                 psiR_Mmin1.push_back(psiR_Mmin1_Elem);
277                 psiR_M.push_back(psiR_M_Elem);
278                 psiL0.push_back(psiL0_Elem);
279                 psiL1.push_back(psiL1_Elem);
280             }
281
282
283             A_block[block][i].push_back(-aMat[i+1][block+1]/(pow(h1,2)));
284             B_block[block][i].push_back(-bMat[i+1][block+1]/(pow(h2,2)));
285             B_block[block][i].push_back(cMat[i+1][block+1]);
286             B_block[block][i].push_back(-bMat[i+2][block+1]/(pow(h2,2)));
287             C_block[block][i].push_back(-aMat[i+1][block+2]/(pow(h1,2)));
288             if(block == 0) //PSI
289             {
290                 psiR_Mmin1[i].push_back(-2*aMat[i+1][M]/(pow(h1,2)));
291                 psiR_M[i].push_back(-bMat[i+1][M]/(pow(h2,2)));
292                 psiR_M[i].push_back(fVec[i+1]);

```

```

293         psiR_M[i].push_back(-bMat[i+2][M]/(pow(h2,2)));
294         psiL0[i].push_back(-bMat[i+1][0]/(pow(h2,2)));
295         psiL0[i].push_back(gVec[i+1]);
296         psiL0[i].push_back(-bMat[i+2][0]/(pow(h2,2)));
297         psiL1[i].push_back(-2*aMat[i+1][1]/(pow(h1,2)));
298     }
299 }
300 }
301
302
303
304 //-----A_2), A_3)-----
305 vector< vector<double> > psiT1_block;
306 vector< vector<double> > psiT2_block;
307 vector< vector<double> > psiT3_block;
308
309 vector< vector<double> > psiB1_block;
310 vector< vector<double> > psiB2_block;
311 vector< vector<double> > psiB3_block;
312
313 vector<double> P00_0;
314 vector<double> P00_1;
315 vector<double> P40_Mmin1;
316 vector<double> P40_M;
317 vector<double> P43_Mmin1;
318 vector<double> P43_M;
319 vector<double> P03_0;
320 vector<double> P03_1;
321
322 P00_0.push_back(k_00);
323 P00_0.push_back(-2*bMat[1][0]/(pow(h2,2)));
324 P00_1.push_back(-2*aMat[0][1]/(pow(h1,2)));
325 P40_Mmin1.push_back(-2*aMat[0][M]/(pow(h1,2)));
326 P40_M.push_back(e11_M0);
327 P40_M.push_back(-2*bMat[1][M]/(pow(h2,2)));
328 P43_Mmin1.push_back(-2*aMat[N][M]/(pow(h1,2)));
329 P43_M.push_back(-2*bMat[N][M]/(pow(h2,2)));
330 P43_M.push_back(r_MN);
331 P03_0.push_back(-2*bMat[N][0]/(pow(h2,2)));
332 P03_0.push_back(s_0N);
333 P03_1.push_back(-2*aMat[N][1]/(pow(h1,2)));
334
335 for(int block = 0; block < M-1; block++)
336 {
337     vector<double> psiElem;
338     psiT1_block.push_back(psiElem);
339     psiT2_block.push_back(psiElem);
340     psiT3_block.push_back(psiElem);
341
342     psiB1_block.push_back(psiElem);
343     psiB2_block.push_back(psiElem);
344     psiB3_block.push_back(psiElem);
345
346     psiB1_block[block].push_back(-aMat[0][block+1]/(pow(h1,2)));
347     psiB2_block[block].push_back(eVec[block+1]);
348     psiB2_block[block].push_back(-2*bMat[1][block+1]/(pow(h2,2)));
349     psiB3_block[block].push_back(-aMat[0][block+2]/(pow(h1,2)));
350     psiT1_block[block].push_back(-aMat[N][block+1]/(pow(h1,2)));
351     psiT2_block[block].push_back(-2*bMat[N][block+1]/(pow(h2,2)));
352     psiT2_block[block].push_back(dVec[block+1]);
353     psiT3_block[block].push_back(-aMat[N][block+2]/(pow(h1,2)));
354 }
355
356 //-----
357 //-----Here will be cycle WHILE-----
358 //-----
359 bool flag = true;
360 int count = 0;
361 while(flag)
362 {
363     count++;
364     double sum_row = 0.0;
365
366
367     //New

```

```

368     sum_row = P00_0[0]*wVec[0] + P00_0[1]*wVec[1] +
369     P00_1[0]*wVec[N+1];
370     AwVec[0] = sum_row;
371
372     //psi_L0, psi_L1
373     #pragma omp parallel for
374     for(int i = 0; i < N-1; i++)
375     {
376         sum_row = 0.0;
377
378         sum_row += psiL0[i][0]*wVec[i];
379         sum_row += psiL0[i][1]*wVec[i+1];
380         sum_row += psiL0[i][2]*wVec[i+2];
381         sum_row += psiL1[i][0]*wVec[N+1 + i+1];
382
383         AwVec[1 + i] = sum_row;    //next: Awec[N]
384     }
385
386     sum_row = P03_0[0]*wVec[N-1] + P03_0[1]*wVec[N] +
387     P03_1[0]*wVec[2*(N+1) - 1];
388     AwVec[N] = sum_row; //next: Awec[N+1]
389
390     #pragma omp parallel for
391     for(int block = 0; block < M-1; block++)
392     {
393         sum_row = 0.0;
394
395         sum_row += psiB1_block[block][0]*wVec[block*(N+1) + 0];
396         sum_row += psiB2_block[block][0]*wVec[block*(N+1) + 0 + N+1];
397         sum_row += psiB2_block[block][1]*wVec[block*(N+1) + 1 + N+1];
398         sum_row += psiB3_block[block][0]*wVec[block*(N+1) + 0 + 2*(N+1)];
399
400         AwVec[N+1 + block*(N+1)] = sum_row;
401         for(int i = 0; i < N-1; i++)
402         {
403             sum_row = 0.0;
404             sum_row += A_block[block][i][0]*wVec[block*(N+1) + i+1];
405             sum_row += B_block[block][i][0]*wVec[block*(N+1) + i + N+1];
406             sum_row += B_block[block][i][1]*wVec[block*(N+1) + i+1 + N+1];
407             sum_row += B_block[block][i][2]*wVec[block*(N+1) + i+2 + N+1];
408             sum_row += C_block[block][i][0]*wVec[block*(N+1) + i+1 + 2*(N+1)];
409
410             AwVec[N+1 + block*(N+1) + 1 + i] = sum_row;
411         }
412         sum_row = 0.0;
413
414         sum_row += psiT1_block[block][0]*wVec[block*(N+1) + N];
415         sum_row += psiT2_block[block][0]*wVec[block*(N+1) + N-1 + N+1];
416         sum_row += psiT2_block[block][1]*wVec[block*(N+1) + N + N+1];
417         sum_row += psiT3_block[block][0]*wVec[block*(N+1) + N + 2*(N+1)];
418
419         AwVec[N+1 + block*(N+1) + 1 + N - 1] = sum_row;
420         //next: Awec[2*N + (M-2)*(N+1) + 2]
421     }
422
423     sum_row = P40_Mmin1[0]*wVec[(M-1)*(N+1)] +
424     P40_M[0]*wVec[M*(N+1)] + P40_M[1]*wVec[M*(N+1) + 1];
425     AwVec[2*N + (M-2)*(N+1) + 2] = sum_row;
426     //next: Awec[2*N + (M-2)*(N+1) + 3] = Awec[M*(N+1) + 1]
427
428     //psi_R_M-1, psi_R_M
429     #pragma omp parallel for
430     for(int i = 0; i < N-1; i++)
431     {
432         sum_row = 0.0;
433
434         sum_row += psiR_Mmin1[i][0]*wVec[(M-1)*(N+1) + i+1];
435         sum_row += psiR_M[i][0]*wVec[M*(N+1) + i];
436         sum_row += psiR_M[i][1]*wVec[M*(N+1) + i+1];
437         sum_row += psiR_M[i][2]*wVec[M*(N+1) + i+2];
438
439         AwVec[M*(N+1)+1 + i] = sum_row; //next: Awec[M*(N+1) + N]
440     }
441
442     sum_row = P43_Mmin1[0]*wVec[M*(N+1) - 1] +

```

```

443 P43_M[0]*wVec[M*(N+1) + N-1] + P43_M[1]*wVec[M*(N+1) + N];
444 AwVec[M*(N+1) + N] = sum_row;
445
446 for(int i = 0; i < (N+1)*(M+1); i++)
447     gammaVec[i] = AwVec[i] - BVec[i];
448
449
450
451 sum_row = P00_0[0]*gammaVec[0] + P00_0[1]*gammaVec[1] +
452 P00_1[0]*gammaVec[N+1];
453 AgammaVec[0] = sum_row;
454
455 //psi_L0, psi_L1
456 #pragma omp parallel for
457 for(int i = 0; i < N-1; i++)
458 {
459     sum_row = 0.0;
460
461     sum_row += psiL0[i][0]*gammaVec[i];
462     sum_row += psiL0[i][1]*gammaVec[i+1];
463     sum_row += psiL0[i][2]*gammaVec[i+2];
464     sum_row += psiL1[i][0]*gammaVec[N+1 + i+1];
465
466     AgammaVec[1 + i] = sum_row;    //next: Awec[N]
467 }
468
469 sum_row = P03_0[0]*gammaVec[N-1] + P03_0[1]*gammaVec[N] +
470 P03_1[0]*gammaVec[2*(N+1) - 1];
471 AgammaVec[N] = sum_row; //next: Awec[N+1]
472
473 #pragma omp parallel for
474 for(int block = 0; block < M-1; block++)
475 {
476     sum_row = 0.0;
477
478     sum_row += psiB1_block[block][0]*gammaVec[block*(N+1) + 0];
479     sum_row += psiB2_block[block][0]*gammaVec[block*(N+1) + 0 + N+1];
480     sum_row += psiB2_block[block][1]*gammaVec[block*(N+1) + 1 + N+1];
481     sum_row += psiB3_block[block][0]*gammaVec[block*(N+1) + 0 + 2*(N+1)];
482
483     AgammaVec[N+1 + block*(N+1)] = sum_row;
484     for(int i = 0; i < N-1; i++)
485     {
486         sum_row = 0.0;
487         sum_row += A_block[block][i][0]*gammaVec[block*(N+1) + i+1];
488         sum_row += B_block[block][i][0]*gammaVec[block*(N+1) + i + N+1];
489         sum_row += B_block[block][i][1]*gammaVec[block*(N+1) + i+1 + N+1];
490         sum_row += B_block[block][i][2]*gammaVec[block*(N+1) + i+2 + N+1];
491         sum_row += C_block[block][i][0]*gammaVec[block*(N+1) + i+1 + 2*(N+1)];
492
493         AgammaVec[N+1 + block*(N+1) + 1 + i] = sum_row;
494     }
495     sum_row = 0.0;
496
497     sum_row += psiT1_block[block][0]*gammaVec[block*(N+1) + N];
498     sum_row += psiT2_block[block][1]*gammaVec[block*(N+1) + N + N+1];
499     sum_row += psiT2_block[block][0]*gammaVec[block*(N+1) + N-1 + N+1];
500     sum_row += psiT3_block[block][0]*gammaVec[block*(N+1) + N + 2*(N+1)];
501
502     AgammaVec[N+1 + block*(N+1) + 1 + N - 1] = sum_row;
503     //next: Awec[2*N + (M-2)*(N+1) + 2]
504 }
505
506 sum_row = P40_Mmini1[0]*gammaVec[(M-1)*(N+1)] +
507 P40_M[0]*gammaVec[M*(N+1)] + P40_M[1]*gammaVec[M*(N+1) + 1];
508 AgammaVec[2*N + (M-2)*(N+1) + 2] = sum_row;
509 //next: Awec[2*N + (M-2)*(N+1) + 3] = Awec[M*(N+1) + 1]
510
511 //psi_R_M-1, psi_R_M
512 #pragma omp parallel for
513 for(int i = 0; i < N-1; i++)
514 {
515     sum_row = 0.0;
516
517     sum_row += psiR_Mmini1[i][0]*gammaVec[(M-1)*(N+1) + i+1];

```

```

518         sum_row += psiR_M[i][0]*gammaVec[M*(N+1) + i];
519         sum_row += psiR_M[i][1]*gammaVec[M*(N+1) + i+1];
520         sum_row += psiR_M[i][2]*gammaVec[M*(N+1) + i+2];
521
522         AgammaVec[M*(N+1)+1 + i] = sum_row; //next: Awec[M*(N+1) + N]
523     }
524
525     sum_row = P43_Mmin1[0]*gammaVec[M*(N+1) - 1] +
526     P43_M[0]*gammaVec[M*(N+1) + N-1] + P43_M[1]*gammaVec[M*(N+1) + N];
527     AgammaVec[M*(N+1) + N] = sum_row;
528
529     numerator = Scalar(AgammaVec, gammaVec, M, N, h1, h2);
530     // cout << "numerator = " << numerator << '\n';
531     denominator = Scalar(AgammaVec, AgammaVec, M, N, h1, h2);
532     // cout << "denominator = " << denominator << '\n';
533     tau = numerator/denominator;
534
535
536     for(int i = 0; i < (N+1)*(M+1); i++)
537     {
538         w_newVec[i] = wVec[i] - tau*gammaVec[i];
539         w_diffVec[i] = w_newVec[i] - wVec[i];
540
541         wVec[i] = w_newVec[i];
542     }
543
544
545
546     //-----Euclidian norm-----
547     //-----
548     norm_val = sqrt(Scalar(w_diffVec, w_diffVec, M, N, h1, h2));
549     //-----
550     //-----END Euclidian norm-----
551     //-----
552
553
554
555     //-----C norm-----
556     //-----
557     // norm_val = abs(*max_element(w_diffVec.begin(), w_diffVec.end()));
558     //-----
559     //-----END C norm-----
560     //-----
561
562
563     // cout << "Norm value = " << norm_val << '\n';
564     if(norm_val < eps)
565         flag = false;
566
567
568 }
569
570 //-----
571 //-----END WHILE-----
572 //-----
573
574 //Save matrix to bin file
575 ofstream myfile;
576 myfile.open ("omegaMat.txt");
577 myfile << M << '\n';
578 myfile << N << '\n';
579 for(int i = 0; i < wVec.size(); i++)
580     myfile << wVec[i] << '\n';
581 myfile.close();
582
583 cout << "eps=" << eps << '\n';
584 cout << "wVec.size=" << wVec.size() << '\n';
585 cout << "h1=" << h1 << '\n';
586 cout << "h2=" << h2 << '\n';
587 cout << "Number_of_iterations=" << count << '\n';
588
589 gettimeofday(&stop, NULL);
590 time = (stop.tv_sec-start.tv_sec) + micro*(stop.tv_usec-start.tv_usec);
591
592 cout << "Work_time=" << time << "seconds." << '\n';

```

```
593     cout << "DONE\n";  
594     return 0;  
595 }
```

Список литературы

[1] Система «Полюс»: <http://hpc.cmc.msu.ru/polus>