Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Московский государственный технический университет имени
Н.Э. Баумана (национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Робототехники и комплексной автоматизации»
КАФЕДРА «Системы автоматизированного проектирования (РК-6)»

ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ

по дисциплине «Вычислительная математика»

Студент:	Новокшанов Евгений Андреевич
Группа:	РК6-66Б
Тип задания:	лабораторная работа
Тема:	Метод конечных разностей

Студент ______________ Новокшанов Е.А.
подпись, дата Фамилия, И.О.

Преподаватель ______________ ______________
подпись, дата Фамилия, И.О.

Москва, 2023

Содержание

Метод конечных разностей

Задание

Метод конечных разностей объединяет целый класс численных методов для решения дифференциальных уравнений путём аппроксимации производных.

В данной задаче необходимо найти распределение температуры в двумерной области (пластине) Ω[1], представленной на рис. 1. Пластина изготовлена из однородного материала. Единицы измерения: время — секунды (сек.), пространство — миллиметры (мм), температура — градусы Цельсия (C°).

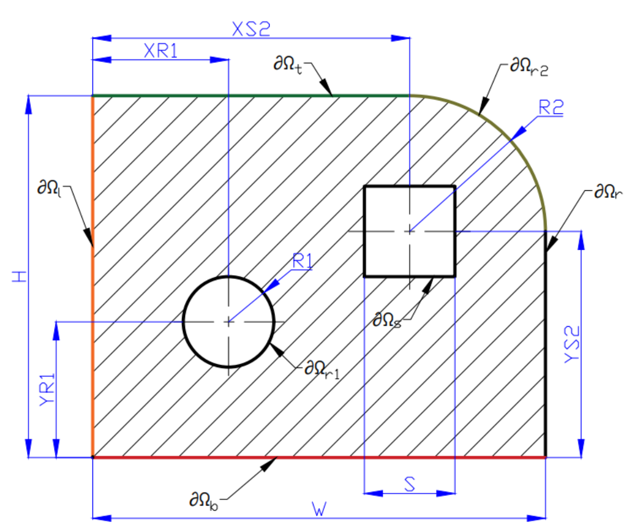

Рис. 1. Чертёж расчётной области пластины Ω

Основные размеры пластины: $H = 400$ мм, $W = 500$ мм, $R2 = 150$ мм, сторона квадратного отверстия $S = 100$ мм, радиус круглого отверстия $R1 = 50$ мм. В каждом варианте задания пластина имеет лишь одно отверстие из двух изображенных на рис. 1. Тип отверстия определяется в соответсвии с параметром $\gamma_1 \in \Gamma_1 = \{\partial\Omega_{r1}, \partial\Omega_s\}$. Координаты центра отверстия $(XR1, YR1)$ или $(XS2, YS2)$ определяются параметром $\gamma_2 \in \Gamma_2 = \{(155,155),(155,255),(355,255),(355,155),(255,205)\}$.

Пусть граница пластины $\partial\Omega$ представлена несколькими участками (рис. 1):

$$\Omega = \Omega_l \cup \Omega_r \cup \Omega_t \cup \Omega_b \cup \Omega_{r1} \cup \Omega_s \cup \Omega_{r2};$$

$$\Omega_{ex} = \Omega_l \cup \Omega_r \cup \Omega_t \cup \Omega_b \cup \Omega_{r2}, \text{ — внешняя граница области } \Omega;$$

$$\Omega_{in} = \Omega_s \cup \Omega_{r1}, \text{ — внутренняя граница области } \Omega.$$

Узловые точки, расположенные на границах $(x,y) \in \partial\Omega$ или их окрестности $(x,y) \in \Omega_{near}$ требуют особого внимания, поскольку в этих областях шаг сетки неравномерный, формулы для расчёта производных можно вывести используя материалы лекций

[1] Ω — область пластины, не включая её границу, $\bar{\Omega} = \Omega \cup \partial\Omega$ — область пластины, включая её границу $\partial\Omega$, Ω_{near} — внутренняя область в h-окрестности границы $\partial\Omega$, где h-шаг сетки.

по Вычислительной математике[2] или материалы БИГОР. Напомним, что внутренняя область пластины обозначена просто Ω и рассматривается двумерная постановка задачи. Известно, что температура T в точке с координатами $(x,y) \in \Omega$ в момент времени t есть отображение $T{:}\mathbb{R} \times \mathbb{R}_{+} \to \mathbb{R}$, которое вычисляется в результате решения дифференциального уравнения теплопроводности:

$$\frac{\partial T}{\partial t} = \Delta T, \qquad (x,y) \in \Omega, \quad t \geq 0, \quad T = T(x,y,t), \tag{1}$$

где Δ — оператор Лапласа, т.е. $\Delta T = frac\partial T \partial x + \frac{\partial T}{\partial y}$.

Начальное значение температуры для всех вариантов имеет вид:

$$T\Big|_{t=0} = 0, (x,y) \in \Omega. \tag{2}$$

При этом, граничные условия для сторон пластины Ω_x будут иметь различный вид:

$$T\Big|_{(x,y)\in\Omega_x} = 100 \text{ — ГУ 1-о рода, задает источник нагрева;} \tag{3}$$

$$\nabla_{\bar{n}} T\Big|_{(x,y)\in\Omega_x} = 0 \text{ —ГУ 3-о рода, теплоизоляция,} \tag{4}$$

где $\nabla_{\bar{n}} T = \frac{\partial T}{\partial \bar{n}} = \nabla_{x,y} T \cdot \bar{n}$ — градиент температуры вдоль внешней нормали $\bar{n}$ к Ω_x;

$$\nabla_{\bar{n}} T\Big|_{(x,y)\in\Omega_x} = T \text{ — ГУ 3-о рода, для конвективного теплообмена на границе } \Omega_x. \tag{5}$$

Физический смысл ГУ 1-о рода может означать, например, соприкосновение стороны поверхности Ω_x,если рассматиривать пластину как сечение тела, с некоторой средой высокой температуры (100 C°). Соотвествие Ω_x конкретному ГУ задает параметр
$\gamma_3 \in \Gamma_3 = \{1,2,3,4,5,6,7,8,9,10,11,12,13\}$, исходя из таблицы 1.

Требуется (базовая часть):

1 Определить форму пластины и ГУ в соответствии с $id_{\Gamma_i}[\gamma_i], \forall i \in [1:3]$.

2 Задать равномерный шаг дискретизации $h \in \{5,10\}$[3] по координатам x и y. Построить расчётную сетку на множестве $\bar{\Omega}$ и рассчитать позиции узлов на границах.

3 Для каждого варианта шага h явным и неявным методом решить нестационарное уравнение теплопроводности (1) при заданных ГУ, определив значения температуры в узлах сетки в диапазоне времени $t \in (0;100]$ сек, с шагом $h_t = 1$ сек.

[2]Глава «Метод разложения функции в ряд Тейлора»

[3]Каждому значению шага h соответсвует один расчет; шаг по координатам x и y одинаков.

Таблица 1. Варианты граничных условий в зависимости от γ_3

	«нагрев» (3)	«теплоизоляция» (4)	«конвекция» (5)
$\gamma_3 = 1$	$\partial\Omega_{ex}$	$\varnothing$	$\partial\Omega_{in}$
$\gamma_3 = 2$	$\partial\Omega_{in}$	$\varnothing$	$\partial\Omega_{ex}$
$\gamma_3 = 3$	$\partial\Omega_l$	$\partial\Omega_{in} \cup \partial\Omega_t \cup \partial\Omega_b$	$\partial\Omega_r$
$\gamma_3 = 4$	$\partial\Omega_r$	$\partial\Omega_{in} \cup \partial\Omega_t \cup \partial\Omega_b$	$\partial\Omega_l$
$\gamma_3 = 5$	$\partial\Omega_t$	$\partial\Omega_{in} \cup \partial\Omega_l \cup \partial\Omega_r$	$\partial\Omega_b$
$\gamma_3 = 6$	$\partial\Omega_b$	$\partial\Omega_{in} \cup \partial\Omega_l \cup \partial\Omega_r$	$\partial\Omega_t$
$\gamma_3 = 7$	$\partial\Omega_l \cup \partial\Omega_r$	$\partial\Omega_{in}$	$\partial\Omega_t \cup \partial\Omega_b$
$\gamma_3 = 8$	$\partial\Omega_l \cup \partial\Omega_r \cup \partial\Omega_{in}$	$\varnothing$	$\partial\Omega_t \cup \partial\Omega_b$
$\gamma_3 = 9$	$\partial\Omega_t \cup \partial\Omega_b$	$\partial\Omega_{in}$	$\partial\Omega_l \cup \partial\Omega_r$
$\gamma_3 = 10$	$\partial\Omega_t \cup \partial\Omega_b \cup \partial\Omega_{in}$	$\varnothing$	$\partial\Omega_l \cup \partial\Omega_r$
$\gamma_3 = 11$	$\partial\Omega_l \cup \partial\Omega_r$	$\varnothing$	$\partial\Omega_t \cup \partial\Omega_b \cup \partial\Omega_{in}$
$\gamma_3 = 12$	$\partial\Omega_t \cup \partial\Omega_b$	$\varnothing$	$\partial\Omega_l \cup \partial\Omega_r \cup \partial\Omega_{in}$
$\gamma_3 = 13$	$\partial\Omega_l \cup \partial\Omega_b$	$\partial\Omega_{in}$	$\partial\Omega_t \cup \partial\Omega_r$

4 Результаты необходимо сохранить в 4-х текстовых файлах[4], имя каждому следует задавать в формате согласно материалу инструкции по выполнению лабораторных работ [1][5]. Содержание каждого файла с результатами расчётов должно соответствовать следующему формату[6]:

```
t x y T
ti xj yk Tijk
...
```

5 Для неявного метода в отчёте должна быть приведена информация о СЛАУ: количество неизвестных (уравнений), число ненулевых элементов матрицы.

6 Сравнить полученные результаты вычислений с результатом моделирования аналогичной задачи в ANSYS.

Требуется (продвинутая часть):

7. Реализовать функцию кубической интеполярции значений температуры для произвольных точек пластины $(x, y) \in \mathbb{R}^2$ по известным значениям в узлах, в дискретные моменты времени;

8. Визуализировать результаты вычислений: функцию поля температуры $f(t_i, x, y)$ по всей пластине в виде цветовой диаграммы в требуемый момент времени (во время защиты).

[4]Рассматривается 2 шага и по 2 метода для каждого.

[5]Пример: edumma$_lab_2023_rk6_63b_ivanovpp_lab1_res2.pdf^-21..6-63, 2023$.

[6]В первой строке наименования параметров, а далее в каждой строке значения аргументов t_i, x_j, y_k и функции $f(t_i, x_j, y_k) = f_{ijk}$. В качестве разделителя могут использоваться: пробел или точка с запятой.

Цель выполнения лабораторной работы

Познакомиться с методом конечных разностей и продемонстрировать с помощью компилируемого языка программирования пример распределения температуры в двумерной области.

1 Метод конечных разностей в общем виде

В первую очередь для реализации МКР требуется дискретизировать время и пространство. Дискретизация пространства происходит путем наложения на пластину сетки с шагом h. Начало координат сетки совпадает с левым нижним углом пластины. Ось X параллельна нижней грани пластины. Ось Y параллельна левой грани пластины. Начало отсчета производится с нуля. На пересечении вертикальных и горизонтальных "полос"сетки находятся узлы, в которых будут вычисляться значения температур. Исключением являются граничные узлы, не попадающие на сетку. Они находятся на пересечении границ пластины с одной из "полос"сетки. Количество узлов в сетке по горизонтали, считая с граничными: $n = \frac{W}{h} + 1$, по вертикали: $m = \frac{H}{h} + 1$. Время дискретизируется путем разделения задачи на шаги, в каждом из которых производится вычисление температуры во всех узлах при фиксированном времени с шагом по времени $h_t = 1$ сек. Начальный момент времени $k = 0$.

Значение темературы в граничных узлах может быть записано формулой (6):

$$T_{ij}^k = \begin{cases} 100,\ \text{ГУ} = (3); \\ 200, (ih, jh) \in \Omega_{r2}; \\ T^t_{i+\Delta i_1, j+\Delta j_1},\ \text{ГУ} = (4); \\ \frac{T^t_{i+\Delta i_1, j+\Delta j_1}}{\mu h+1},\ \text{ГУ} = (5), \end{cases} \tag{6}$$

где $i = \{0, n\}$ при $j = 1..m-1$; $j = \{0, m\}$ при $i = 1..n-1$; $\Delta i_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \bar{n}, \Delta j_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \bar{n}$, μ - соотношение расстояния между узлом (i, j) и $(i + \Delta i_1, j + \Delta j_1)$ к h.

2 Метод дробления шага

Если граничный узел не совпадает с реальной границей рассматриваемого объекта, то для вычисления 1-й производной необходимо применять метод дробления шага. Суть данного метода заключается в дроблении шага сетки на граничном узле так, что бы новый узел совпадал с реальной границей объекта. Формула представлена ниже:

Формула для второй производной:

$$\frac{\partial^2 T}{\partial x^2} = 2\frac{\mu V_C + V_A - (\mu + 1)V_B}{(\mu^2 + \mu) \vartriangle x^2} \tag{7}$$

Формула для первой производной:

$$\frac{\partial T}{\partial x} = \frac{\mu^2 V_C - V_A - (\mu^2 - 1)V_B}{(\mu^2 - \mu) \vartriangle x} \tag{8}$$

Для узлов, которые не являются граничными, мы принимаем $\mu = 1$ и вычисляем 2-ю производную.

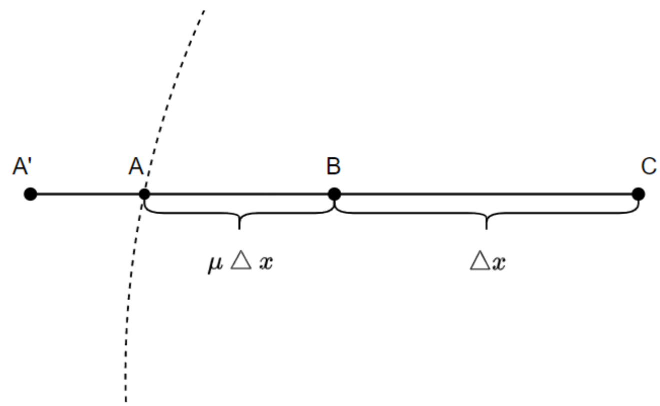

Рис. 2. Иллюстрация схемы с дроблением шага

3 Метод конечных элементов в явном виде

Явная разностная схема, из которой мы находим температуру на k+1 шаге, имеет следующий вид:

$$\frac{T_{i,j}^{k+1} - T_{i,j}^{k}}{\triangle t} = \frac{\partial^2 T_j^k}{\partial x^2} + \frac{\partial^2 T_i^k}{\partial y^2} \tag{9}$$

Схема расположения узлов и их индексации представлена ниже:

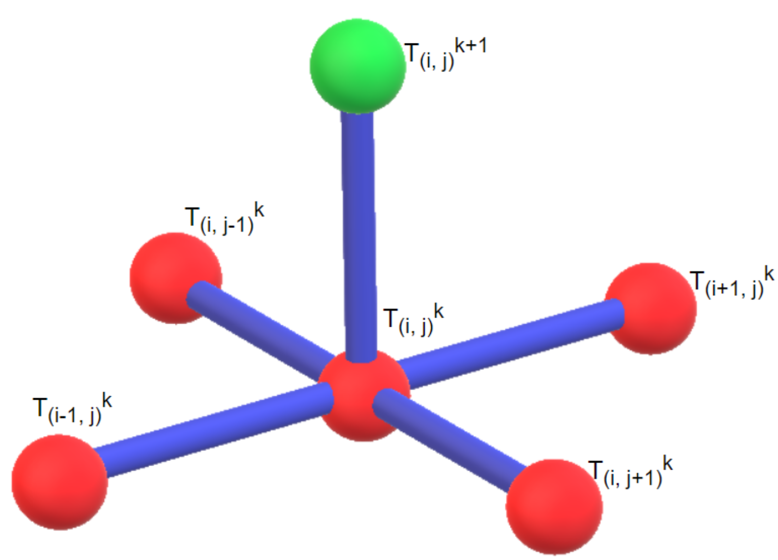

Рис. 3. Визуальное представление сетки на k-м и k+1-м шаге

4 Метод конечных элементов в неявном виде

Для решения задачи неявным МКР применяется метод расщепления.

Первым этапом по значениям температур предыдущего шага находятся температуры w_{ij}^{t+1}, по значениям которых на втором этапе находятся температуры на новом шаге T_{ij}^{t+1}. При этом $i = 1..n-1, j = 1..m-1$.

Формула на первом этапе следующая:

$$w_{ij}^{t+1} - T_{ij}^{t} = 2\frac{\mu_x w_{i+\Delta i_2,j}^{t+1} - (1+\mu_x)w_{ij}^{t+1} + w_{i-\Delta i_2,j}^{t+1}}{\mu_x(1+\mu_x)h^2}, \tag{10}$$

(10) можно представить в виде:

$$-T_{ij}^{t} = \frac{2}{(1+\mu_x)h^2}w_{i+\Delta i_2,j}^{t+1} + \frac{-2-\mu_x h^2}{\mu_x h^2}w_{ij}^{t+1} + \frac{2}{\mu_x(1+\mu_x)h^2}w_{i-\Delta i_2,j}^{t+1}, \tag{11}$$

При фиксации j, (11) можно представить ввиде СЛАУ. При $i = \{1, n-1\}$ при этом в СЛАУ будут фигурировать ГУ. От них можно избавиться путем подстановки (6). Такая подстановка влияет либо на коэффициент перед T_{ij}^{t}, либо перед w_{ij}^{t+1}. Дополнительно можно подставить $\mu_x = 1$ для $i = 2..n-2$.

Результирующая СЛАУ в матричном виде:

$$\begin{bmatrix} \frac{-2-\mu_{x1}h^2}{\mu_{x1}h^2} + \Delta w_1 & \frac{2}{(1+\mu_{x1})h^2} & 0 & 0 & \dots & 0 & 0 & 0 \\ \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} & 0 & \dots & 0 & 0 & 0 \\ 0 & \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} \\ 0 & 0 & 0 & 0 & \dots & 0 & \frac{2}{(1+\mu_{x2})h^2} + \Delta w_2 & \frac{-2-\mu_{x2}h^2}{\mu_{x2}h^2} \end{bmatrix} \begin{bmatrix} w_{1j}^{t+1} \\ w_{2j}^{t+1} \\ w_{3j}^{t+1} \\ \vdots \\ w_{n-2,j}^{t+1} \\ w_{n-1,j}^{t+1} \end{bmatrix} = \tag{12}$$

$$= \begin{bmatrix} -T_{1j}^{t} + \Delta T_1 \\ -T_{2j}^{t} \\ -T_{3j}^{t} \\ \vdots \\ -T_{n-2,j}^{t} \\ -T_{n-1,j}^{t} + \Delta T_2 \end{bmatrix},$$

где μ_{x1} - отношение расстояния между узлом $(0, j)$ и узлом $(1, j)$ к h; μ_{x2} - отношение расстояния между узлом $(n-1, j)$ и узлом (n, j) к h;

$$\Delta w_1 = \begin{cases} 0, \text{ГУ в } (0,j) \text{ 1-го рода;} \\ \frac{2}{\mu_{x1}(1+\mu_{x1})h^2}, \text{ГУ в } (0,j) = (4); \\ \frac{2}{\mu_{x1}(1+\mu_{x1})h^2(\mu_{x1}h+1)}, \text{ГУ в } (0,j) = (5). \end{cases}, \Delta w_2 = \begin{cases} 0, \text{ГУ в } (n,j) \text{ 1-го рода;} \\ \frac{2}{\mu_{x2}(1+\mu_{x2})h^2}, \text{ГУ в } (n,j) = (4); \\ \frac{2}{\mu_{x2}(1+\mu_{x2})h^2(\mu_{x2}h+1)}, \text{ГУ в } (n,j) = (5). \end{cases},$$

$$\Delta T_1 = \begin{cases} -\frac{100}{\mu_{x1}(1+\mu_{x1})h^2}, \text{ГУ в } (0,j) : T_{ij}^{t+1} = 100; \\ -\frac{200}{\mu_{x1}(1+\mu_{x1})h^2}, \text{ГУ в } (0,j) : T_{ij}^{t+1} = 200; \\ 0, \text{в остальных случаях.} \end{cases}, \Delta T_2 = \begin{cases} -\frac{100}{\mu_{x2}(1+\mu_{x2})h^2}, \text{ГУ в } (n,j) : T_{ij}^{t+1} = 100; \\ -\frac{200}{\mu_{x2}(1+\mu_{x2})h^2}, \text{ГУ в } (n,j) : T_{ij}^{t+1} = 200; \\ 0, \text{в остальных случаях.} \end{cases}.$$

При решении СЛАУ (12) для всех $j = 1..m-1$ получается массив значений w_{ij}^{t+1}, $i = 1..n-1$. На основе него аналогичным первому этапу производится второй этап вычислений:

$$T_{ij}^{t+1} - w_{ij}^{t+1} = 2\frac{\mu_y T_{i,j+\Delta j_2}^{t+1} - (1+\mu_y)T_{ij}^{t+1} + T_{i,j-\Delta j_2}^{t+1}}{\mu_y(1+\mu_y)h^2}, \tag{13}$$

(13) можно представить в виде:

$$-w_{ij}^{t+1} = \frac{2}{(1+\mu_y)h^2}T_{i,j+\Delta j_2}^{t+1} + \frac{-2-\mu_y h^2}{\mu_y h^2}T_{ij}^{t+1} + \frac{2}{\mu_y(1+\mu_y)h^2}T_{i,j-\Delta j_2}^{t+1}, \tag{14}$$

При фиксации i, (14) можно представить ввиде СЛАУ. При $j = \{1, m-1\}$ в СЛАУ будут фигурировать ГУ. От них можно избавиться путем подстановки (6). Такая подстановка влияет либо на коэффициент перед w_{ij}^{t+1}, либо перед T_{ij}^{t+1}. Дополнительно можно подставить $\mu_y = 1$ для $j = 2..m-2$.

Результирующая СЛАУ в матричном виде:

$$\begin{bmatrix} \frac{-2-\mu_{y1}h^2}{\mu_{y1}h^2} + \Delta x_1 & \frac{2}{(1+\mu_{y1})h^2} & 0 & 0 & \dots & 0 & 0 & 0 \\ \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} & 0 & \dots & 0 & 0 & 0 \\ 0 & \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} \\ 0 & 0 & 0 & 0 & \dots & 0 & \frac{2}{(1+\mu_{y2})h^2} + \Delta x_2 & \frac{-2-\mu_{y2}h^2}{\mu_{y2}h^2} \end{bmatrix} \begin{bmatrix} T_{i1}^{t+1} \\ T_{i2}^{t+1} \\ T_{i3}^{t+1} \\ \vdots \\ T_{i,m-2}^{t+1} \\ T_{i,n-1}^{t+1} \end{bmatrix} = \tag{15}$$

$$= \begin{bmatrix} -w_{i1}^{t+1} + \Delta T_3 \\ -w^{t+1t}{}_{i2} \\ -w^{t+1t}{}_{i3} \\ \vdots \\ -w^{t+1t}{}_{i,n-2} \\ -w^{t+1t}{}_{i,m-1} + \Delta T_4 \end{bmatrix},$$

где μ_{y1} - отношение расстояния между узлом $(i,0)$ и узлом $(i,1)$ к h; μ_{y2} - отношение расстояния между узлом $(i,m-1)$ и узлом (i,m) к h;

$$\Delta x_1 = \begin{cases} 0, \text{ГУ в } (i,0) \text{ 1-го рода}; \\ \frac{2}{\mu_{y1}(1+\mu_{y1})h^2}, \text{ГУ в } (i,0) = (4); \\ \frac{2}{\mu_{y1}(1+\mu_{y1})h^2(\mu_{y1}h+1)}, \text{ГУ в } (i,0) = (5). \end{cases}, \Delta x_2 = \begin{cases} 0, \text{ГУ в } (i,m) \text{ 1-го рода}; \\ \frac{2}{\mu_{y2}(1+\mu_{y2})h^2}, \text{ГУ в } (i,m) = (4); \\ \frac{2}{\mu_{y2}(1+\mu_{y2})h^2(\mu_{y2}h+1)}, \text{ГУ в } (i,m) = (5). \end{cases},$$

$$\Delta T_3 = \begin{cases} -\frac{100}{\mu_{y1}(1+\mu_{y1})h^2}, \text{ГУ в } (i,0) : T_{ij}^{t+1} = 100; \\ -\frac{200}{\mu_{y1}(1+\mu_{y1})h^2}, \text{ГУ в } (i,0) : T_{ij}^{t+1} = 200; \\ 0, \text{в остальных случаях}. \end{cases}, \Delta T_4 = \begin{cases} -\frac{100}{\mu_{y2}(1+\mu_{y2})h^2}, \text{ГУ в } (i,m) : T_{ij}^{t+1} = 100; \\ -\frac{200}{\mu_{y2}(1+\mu_{y2})h^2}, \text{ГУ в } (i,m) : T_{ij}^{t+1} = 200; \\ 0, \text{в остальных случаях}. \end{cases}.$$

При решении (15) для всех i находятся значения во всех внутренних узлах на новом шаге. Для нахождения значений температур в граничных узлах используется формула (6) после вычисления значения во внутренних узлах.

Цикл нахождения повторяется до достижения поставленного задачей времени выполнения.
В итоге, в неявном методе составляется и решается 2 СЛАУ, где количество уравнений равно количеству внутренних узлов пластины, а в матричном виде мы получаем $3 \cdot N - 2$ ненулевых элементов матрицы.
Схема расположения узлов и их индексации представлена ниже:

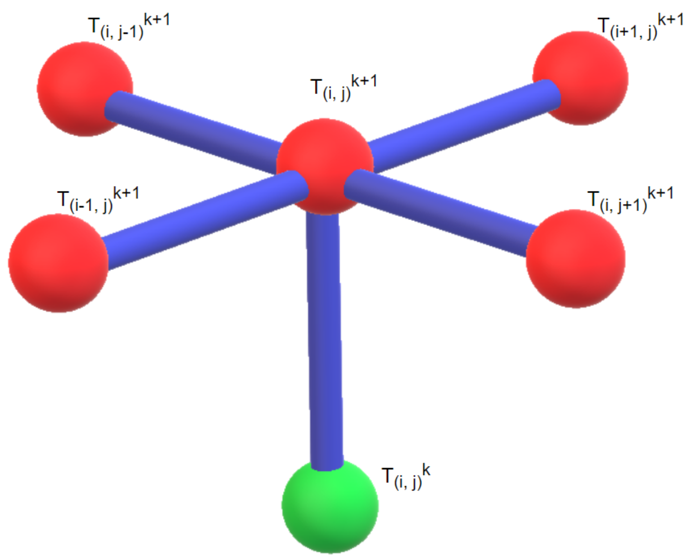

Рис. 4. Визуальное представление сетки на k-м и k+1-м шаге

5 Описание структур данных

Для выполнения поставленных задач, а именно программной реализации метода конечных разностей в явном и неявном виде, были разработаны следующие классы:

- Object;
- Hole;
- Circle - public наследование класса Hole;
- Square - public наследование класса Hole;
- Point.

UML-диаграмма их взаимодействия представлена ниже:

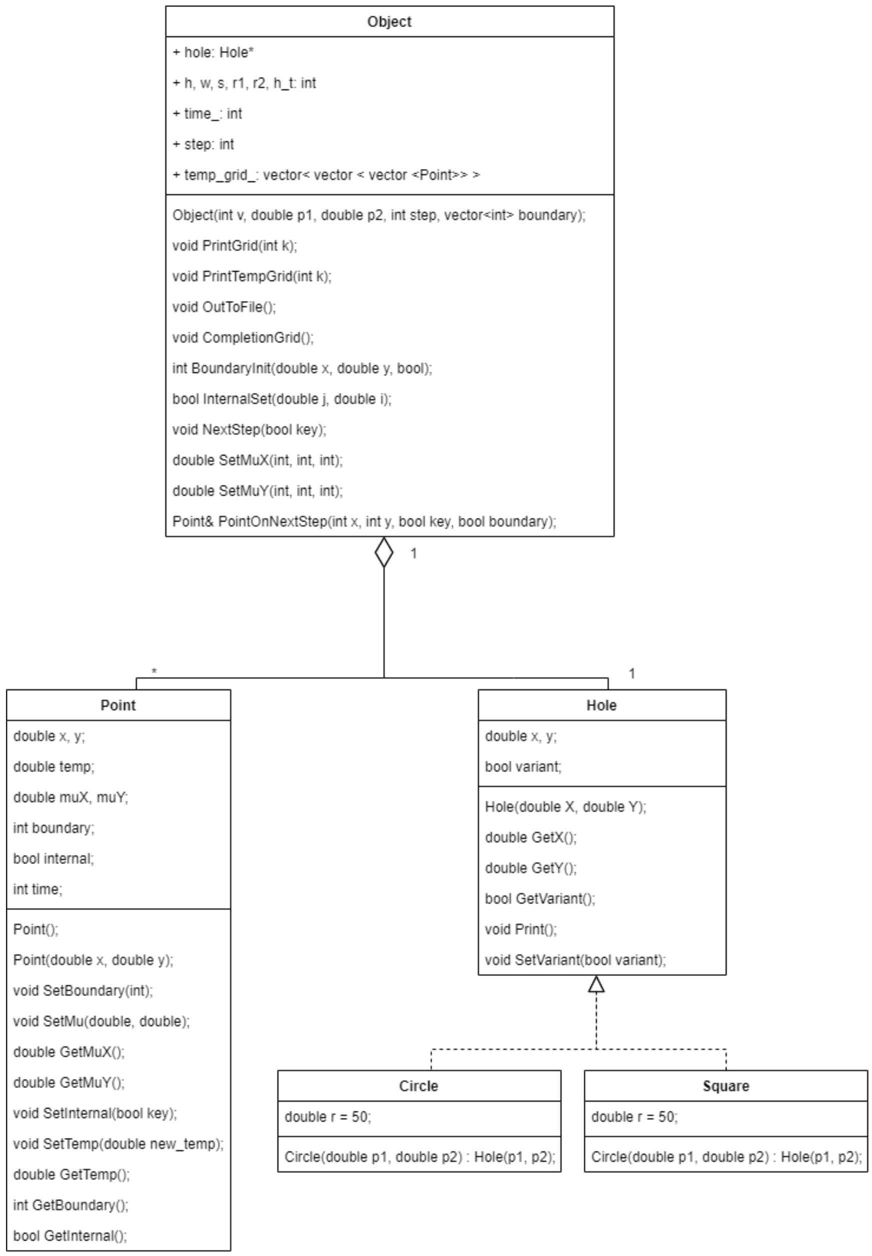

Рис. 5. UML-диаграмма классов

Класс **Object** - объект, который хранит температуру каждого узла в каждый момент времени. В инициализаторе данного класса вызывается метод **CompletionGrid** который формирует сетку в начальный мемент времени и проставляет начальные условия, которые передаются программе с помощью файла **input.txt**. Матрица для храниения сетки реализуется с помощью класса вектора стандартной библиотеки C++, в котором хранятся объекты пользовательского класса Point. Вычисление температуры в каждом узле в следующий момент времени вычисляется с помощью метода **NextStep**, в который передается ключ типа **boolean**, с помощью которого определяется метод. Ниже представлен листинг класса **Object**.

Листинг 1. Листинг файла Object.h

```
class Object {
private:
  int h = 400, w = 500;
  int s = 100;
  int r1 = 50, r2 = 150;
  int h_s, w_s;
  int h_t = 1;
  int time_;
  Hole* hole;
  int step;
  vector<int> boundary_;
  vector< vector < vector <Point>> > temp_grid_;
  vector< pair <char, int> > boundary_conditions_;
public:
  Object(int v, double p1, double p2, int step, vector<int> boundary);
  void PrintGrid(int k);
  void PrintTempGrid(int k);
  void OutToFile();
  void CompletionGrid();
  void SetBoundary(vector<int> boundary);
  int BoundaryInit(double x, double y, bool);
  bool InternalSet(double j, double i);
  Point& PointOnNextStep(int x, int y, bool key, bool boundary);
  void NextStep(bool key);
  double SetMuX(int, int, int);
  double SetMuY(int, int, int);
};
```

Узлы сетки реализованы с помощью объектов пользовательского класса **Point**. В данном классе реализованы основные **set** и **get** методы для изменения параметров объекта и получения данных и области **private**. Ниже приведен листинг класса **Point**.

Листинг 2. Листинг файла Point.h

```
#pragma once
class Point {
private:
  double x, y;
  double temp;
  double muX, muY;
  int boundary;
  bool internal;
```

```
  int time;
public:
  Point();
  Point(double x, double y);
  void SetBoundary(int);
  void SetMu(double, double);
  double GetMuX();
  double GetMuY();
  void SetInternal(bool key);
  void SetTemp(double new_temp);
  double GetTemp();
  int GetBoundary();
  bool GetInternal();
};
```

В методе **CompletionGrid** с помощью методов **BoundaryInit**, **InternalSet**, **SetMuX** и **SetMuY** происходит вычисление коэффициентов μ, распределение узлов на внутренние и внешние, а так же на граничные и не граничные, которое используется для правильной реализации МКР. Все это передается в соответствующий объект класса **Point**.

Для задания координат отверстия и его формы были реализованы классы **Hole**(родительский класс), **Circle**(дочерний класс) и **Square**(дочерний класс). Вид отверстия определяется в конструкторе класса **Object** с помощью переопрделения класса **Hole**.

6 Инструкция по работе с программой

Параметры для работы программы записываются в файле input.txt, где на каждой новой строке находится новый параметр:

1. Шаг сетки: 5 или 10;
2. Вариант: определяет то, каким будет отверстие. Если вариант нечетный, то отверстие имеет форму квадрата, если четный, то отверстие имеет форму круга;
3. Указывается X координата центра отверстия;
4. Указывается Y координата центра отверстия;
5. Узазывается граничное условие для 1-й грани;
6. Узазывается граничное условие для 2-й грани;
7. Узазывается граничное условие для 3-й грани;
8. Узазывается граничное условие для 4-й грани;
9. Узазывается граничное условие для 5-й грани;

10. Узазывается граничное условие для граней отверстия.

Граничные условия: 1 - 100 градусов(нагрев) 2 - 200 градусов(нагрев) 3 - Граничное условие второго рода(теплоизоляция) 4 - Граничное условие третьего рода(конвекция)

Нумерация граней:

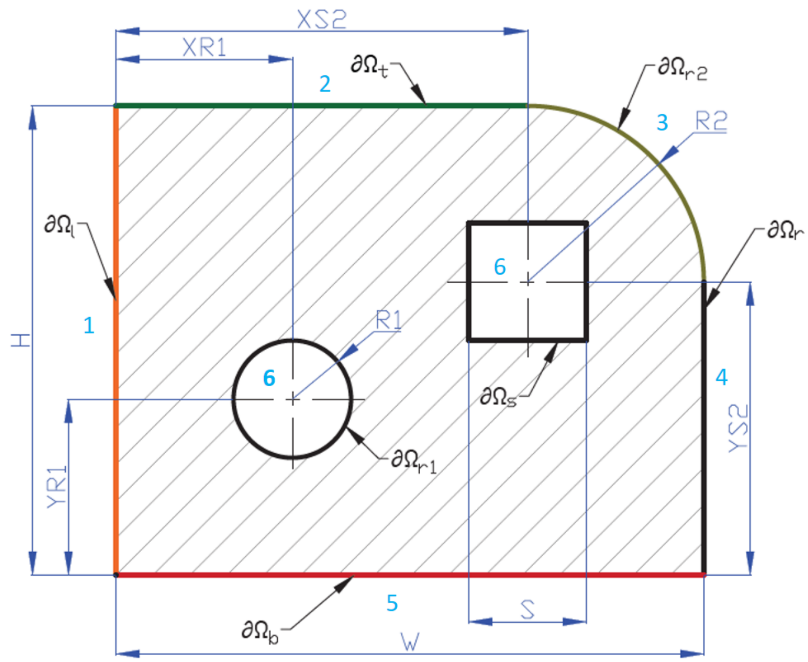

Рис. 6. Сформированная сетка

Системные требования:

1. Для работы явного метода с шагом 10 в среднем требуется 100 MB ОЗУ;
2. Для работы явного метода с шагом 5 в среднем требуется 1 GB ОЗУ;
3. Для работы неявного метода с шагом 10 в среднем требуется 100 MB ОЗУ;
4. Для работы неявного метода с шагом 5 в среднем требуется 1 GB ОЗУ.

7 Сравнение результатов моделирования

В соответствии со своим 17-м вариантом представлены следующие результаты.
Результаты работы программы в программном комплексе Ansys:

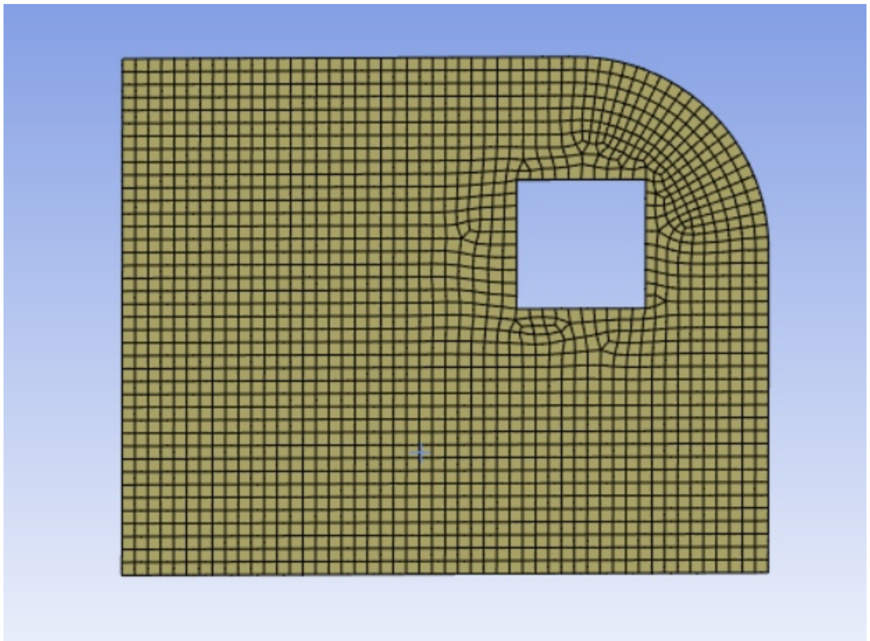

Рис. 7. Сформированная сетка

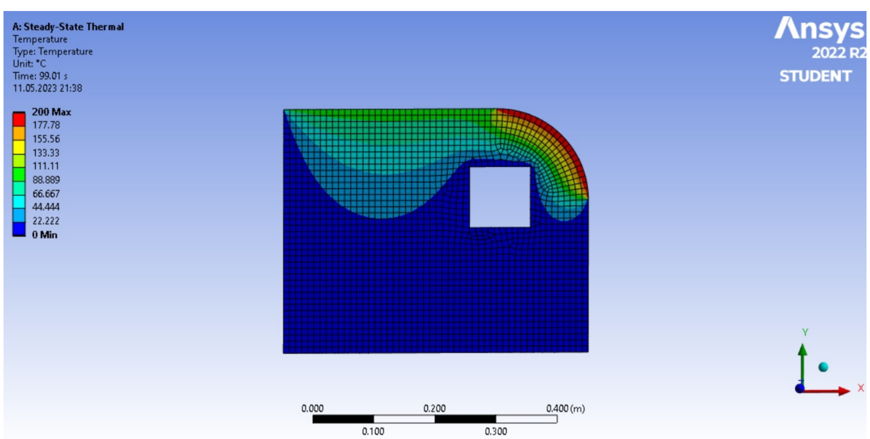

Рис. 8. Температура пластины на 100-й секунде

Результаты работы разработанной программы:

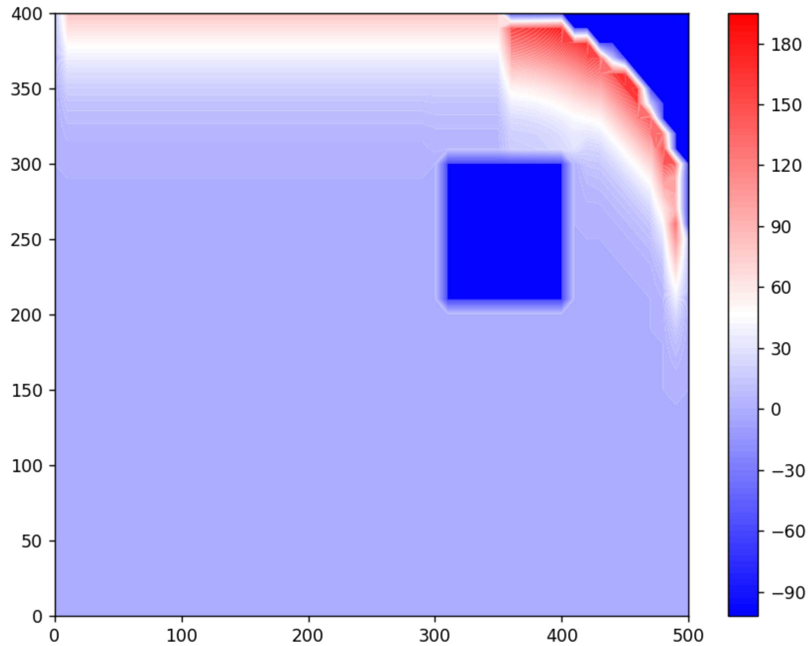

Рис. 9. Температура пластины на 100-й секунде

Результаты работы разработанной программы и моделирования в программе Ansys совпадают с некоторой неточностью, что вызвано тем, что мы не знаем точный материал пластины.

Заключение

При выполнении лабораторной работы была построена модель в ANSYS, заданы необходимые граничные условия, реализован программный аналог решения задачи на языке программирования C++.

Список использованных источников

1. Першин А.Ю. Лекции по курсу «Вычислительная математика». Москва, 2018-2021. С. 140. URL: `https://archrk6.bmstu.ru/index.php/f/810046`.

2. Соколов, А.П. Инструкция по выполнению лабораторных работ (общая). Москва: Соколов, А.П., 2018-2021. С. 9. URL: `https://archrk6.bmstu.ru`. (облачный сервис кафедры РК6).

3. Соколов, А.П. Инструкция по выполнению заданий к семинарским занятиям (общая). Москва: Соколов, А.П., 2018-2022. С. 7. URL: `https://archrk6.bmstu.ru`. (облачный сервис кафедры РК6).

4. Першин А.Ю. Сборник задач семинарских занятий по курсу «Вычислительная математика»: Учебное пособие. / Под редакцией Соколова А.П. [Электронный ресурс]. Москва, 2018-2021. С. 20. URL: https://archrk6.bmstu.ru. (облачный сервис кафедры РК6).

5. Першин А.Ю., Соколов А.П. Сборник постановок задач на лабораторные работы по курсу «Вычислительная математика»: Учебное пособие. [Электронный ресурс]. Москва, 2021. С. 54. URL: https://archrk6.bmstu.ru. (облачный сервис кафедры РК6).

Выходные данные

Новокшанов Е.А.Отчет о выполнении лабораторной работы по дисциплине «Вычислительная математика». [Электронный ресурс] — Москва: 2023. — 17 с. URL: https://sa2systems.ru:88 (система контроля версий кафедры РК6)

2023, весенний семестр