



МИНИСТЕРСТВО НАУКИ  
И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Новосибирский государственный технический университет»

**НГТУ**



**НЭТИ**

Кафедра прикладной математики

Курсовой проект  
по дисциплине «Численные методы»



**ФПМИ**

ГРУППА

ПМ-92

ВАРИАНТ

21

СТУДЕНТ

ГЛУШКО ВЛАДИСЛАВ

ПРЕПОДАВАТЕЛЬ

СОЛОВЕЙЧИК ЮРИЙ ГРИГОРЬЕВИЧ

Новосибирск

# 1 Условие задачи

## Формулировка задачи

МКЭ для двумерной краевой задачи для эллиптического уравнения в декартовой системе координат. Базисные функции линейные на треугольниках. Краевые условия всех типов. Коэффициент разложить по линейным базисным функциям. Матрицу СЛАУ генерировать в разреженном строчном формате. Для решения СЛАУ использовать МСГ или ЛОС с неполной факторизацией.

## Постановка задачи

Эллиптическая краевая задача для функции  $u$  определяется дифференциальным уравнением

$$-div(\lambda \text{gradu}) + \gamma u = f$$

заданным в некоторой области  $\Omega$  с границей  $S = S_1 \cup S_2 \cup S_3$  и краевыми условиями:

$$\begin{aligned} u|_{S_1} &= u_g \\ \lambda \frac{\partial u}{\partial n} \Big|_{S_2} &= \theta \\ \lambda \frac{\partial u}{\partial n} \Big|_{S_3} + \beta(u|_{S_3} - u_\beta) &= 0 \end{aligned}$$

В декартовой системе координат  $x, y$  это уравнение может быть записано в виде

$$-\frac{\partial}{\partial x} \left( \lambda \frac{\partial u}{\partial x} \right) - \frac{\partial}{\partial y} \left( \lambda \frac{\partial u}{\partial y} \right) + \gamma u = f$$

## Конечноэлементная дискретизация

Так как для решения задачи используются линейные базисные функции, то на каждом конечном элементе  $\Omega_k$  - треугольнике эти функции будут совпадать с функциями  $L_1(x, y)$ ,  $L_2(x, y)$ ,  $L_3(x, y)$ , такими, что  $L_1(x, y)$  равна единице в вершине  $(x_1, y_1)$  и нулю во всех остальных вершинах,  $L_2(x, y)$  равна единице в вершине  $(x_2, y_2)$  и нулю во всех остальных вершинах,  $L_3(x, y)$  равна единице в вершине  $(x_3, y_3)$  и нулю во всех остальных вершинах. Любая линейная на  $\Omega_k$  функция представима в виде линейной комбинации этих базисных линейных функций, коэффициентами будут значения функции в каждой из вершин треугольника  $\Omega_k$ . Таким образом, на каждом конечном элементе нам понадобятся три узла – вершины треугольника.

$$\psi_1 = L_1(x, y)$$

$$\psi_2 = L_2(x, y)$$

$$\psi_3 = L_3(x, y)$$

Учитывая построение  $L$ -функций, получаем следующие соотношения:

$$\begin{cases} L_1 + L_2 + L_3 = 1 \\ L_1 x_1 + L_2 x_2 + L_3 x_3 = x \\ L_1 y_1 + L_2 y_2 + L_3 y_3 = y \end{cases}$$

Т.е. имеем систему:

$$\begin{pmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{pmatrix} \cdot \begin{pmatrix} L_1 \\ L_2 \\ L_3 \end{pmatrix} = \begin{pmatrix} 1 \\ x \\ y \end{pmatrix}$$

Отсюда находим коэффициенты линейных функций  $L_1(x, y), L_2(x, y), L_3(x, y)$

$$L_i = a_0^i + a_1^i x + a_2^i y, i = \overline{1, 3}$$

$$\begin{pmatrix} \alpha_0^1 & \alpha_1^1 & \alpha_2^1 \\ \alpha_0^2 & \alpha_1^2 & \alpha_2^2 \\ \alpha_0^3 & \alpha_1^3 & \alpha_2^3 \end{pmatrix} = D^{-1} = \begin{pmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{pmatrix}^{-1}$$

$$D^{-1} = \frac{1}{|\det D|} \begin{pmatrix} x_2 y_3 - x_3 y_2 & y_2 - y_3 & x_3 - x_2 \\ x_3 y_1 - x_1 y_3 & y_3 - y_1 & x_1 - x_3 \\ x_1 y_2 - x_2 y_1 & y_1 - y_2 & x_2 - x_1 \end{pmatrix}$$

## Переход к локальным матрицам

Чтобы получить выражения для локальных матриц жёсткости  $G$  и массы  $M$  каждого конечного элемента  $\Omega_K$ , перейдём к решению локальной задачи на каждом конечном элементе. Полученное уравнение для области  $\Omega$  представим в виде суммы интегралов по областям  $\Omega_k$  без учёта краевых условий. Тогда на каждом конечном элементе будем решать локальную задачу построения матриц жёсткости, массы и вектора правой части.

$$\int_{\Omega_k} \lambda \left( \frac{\partial \psi_j}{\partial x} \frac{\partial \psi_i}{\partial x} + \frac{\partial \psi_j}{\partial y} \frac{\partial \psi_i}{\partial y} \right) dx dy + \int_{\Omega_k} \gamma \psi_j \psi_i ds dy = \int_{\Omega_k} f \psi_i dx dy$$

Локальная матрица будет представлять собой сумму матриц жёсткости и массы и будет иметь размерность  $3 \times 3$  (по числу узлов на конечном элементе)

## Построение матрицы массы

$$\begin{aligned} M_{ij} &= \int_{\Omega_m} \gamma Y_i Y_j d\Omega_m = \left| \gamma = Y_1 \gamma_1 + Y_2 \gamma_2 + Y_3 \gamma_3 \right| = \int_{\Omega_m} (Y_1 \gamma_1 + Y_2 \gamma_2 + Y_3 \gamma_3) Y_i Y_j d\Omega_m = \\ &= \gamma_1 \int_{\Omega_m} Y_1 Y_i Y_j d\Omega_m + \gamma_2 \int_{\Omega_m} Y_2 Y_i Y_j d\Omega_m + \gamma_3 \int_{\Omega_m} Y_3 Y_i Y_j d\Omega_m = \\ &= \gamma_1 \int_{\Omega_m} L_1 L_i L_j d\Omega_m + \gamma_2 \int_{\Omega_m} L_2 L_i L_j d\Omega_m + \gamma_3 \int_{\Omega_m} L_3 L_i L_j d\Omega_m \end{aligned}$$

## Построение матрицы жёсткости

Рассмотрим первый член в выражении для  $k$ -го конечного элемента:

$$\int_{\Omega_k} \lambda \left( \frac{\partial \psi_j}{\partial x} \frac{\partial \psi_i}{\partial x} + \frac{\partial \psi_j}{\partial y} \frac{\partial \psi_i}{\partial y} \right) dxdy$$
$$B_{i,j} = (\alpha_1^i \alpha_1^j + \alpha_2^i \alpha_2^j) \frac{|det D|}{2} \quad i, j = \overline{0, 2}$$

## Построение вектора правой части

Рассмотрим правую часть выражения для  $k$ -го конечного элемента:

$$\int_{\Omega_k} f \psi_i dxdy$$

представим  $f$  в виде  $f_1 L_1 + f_2 L_2 + f_3 L_3$ , где  $f_i$  - значения в вершинах треугольника. Получим:

$$\int_{\Omega_k} f_q L_q L_i dxdy = f_q \int_{\Omega_k} L_q L_i d\Omega_k$$

Таким образом:

$$G_i = \sum_{q=1}^3 f_q \int_{\Omega_k} L_q L_i d\Omega_k \quad i = \overline{0, 2}$$

## Сборка глобальной матрицы и глобального вектора

При формировании глобальной матрицы из локальных, полученных суммированием соответствующих матриц массы и жесткости, учитываем соответствие локальной и глобальной нумераций каждого конечного элемента. Глобальная нумерация каждого конечного элемента однозначно определяет позиции вклада его локальной матрицы в глобальную. Поэтому, зная глобальные номера соответствующих узлов конечного элемента, определяем и то, какие элементы глобальной матрицы изменятся при учете текущего конечного элемента. Аналогичным образом определяется вклад локального вектора правой части в глобальный. При учете текущего локального вектора изменятся те элементы глобального вектора правой части, номера которых совпадают с глобальными номерами узлов, присутствующих в этом конечном элементе.

## Учёт первых краевых условий

Для учета первых краевых условий, в глобальной матрице и глобальном векторе находим соответствующую глобальному номеру краевого узла строку и зануляем всё кроме диагонального элемента, которому присваиваем 1, а вместо элемента с таким номером в векторе правой части - значение краевого условия, заданное в исходной задаче.

## Учёт вторых и третьих краевых условий

Рассмотрим краевые условия второго и третьего рода:

$$\lambda \frac{\partial u}{\partial n} \Big|_{S_2} = \theta$$

$$\lambda \frac{\partial u}{\partial n} \Big|_{S_3} + \beta(u|_{S_3} - u_\beta) = 0$$

Отсюда получаем, что для учёта краевых условий необходимо вычислить интегралы:

$$\int_{S_2} \theta \psi_j dx dy, \quad \int_{S_3} \beta u_\beta \psi_j dx dy, \quad \int_{S_3} \beta \psi_i \psi_j dx dy$$

Краевые условия второго и третьего рода задаются на рёбрах, т.е. определяются двумя узлами, лежащими на ребре. Будем считать, что параметр  $\beta$  на  $S_3$  постоянен, тогда параметр  $\beta$  будем раскладывать по двум базисным функциям, определённым на этом ребре:

$$u_\beta = u_{\beta 1} \phi_1 + u_{\beta 2} \phi_2$$

где  $\phi_i$ ,  $i = \overline{0, 1}$  - локально занумерованные линейные базисные функции, которые имеют также свои глобальные номера во всей расчетной области, а  $u_{\beta i}$  - значение функции  $u_\beta$  в узлах ребра.

Аналогично поступаем и при учете вторых краевых условий, раскладывая по базису ребра функцию  $\theta = \theta_0 \phi_0 + \theta_1 \phi_1$ .

Тогда приведенные выше интегралы примут вид:

$$I_1 = \int_{S_2} (\theta_0 \phi_0 + \theta_1 \phi_1) \phi_i dx dy$$
$$I_2 = \beta \int_{S_3} (u_{\beta 1} \phi_0 + u_{\beta 2} \phi_1) \phi_i dx dy$$
$$I_3 = \beta \int_{S_3} \phi_i \phi_j dx dy$$

Фактически, решая задачу учета краевых условий второго и третьего рода, мы переходим к решению одномерной задачи на ребре для того, чтобы занести соответствующие результаты в глобальную матрицу и вектор.

Базисными функциями ребра являются две ненулевые на данном ребре базисные функции из  $\phi_i$ ,  $i = \overline{0, 1}$  конечного элемента.

Для учёта вклада вторых и третьих краевых условий рассчитываются 2 матрицы  $2 \times 2$ .

Интегралы  $I_1, I_2, I_3$  будем вычислять по формуле:

$$\int (L_i)^{v_i} (L_j)^{v_j} dS = \frac{v_i! v_j!}{(v_i + v_j + 1)!} \text{mes} \Gamma, \quad i \neq j$$

где  $mes\Gamma$  длина ребра. При этом независимо от того, что на каждом из ребер присутствуют свои функции, интегралы, посчитанные по приведенным выше формулам, будут равны.

$$I_1 = \begin{pmatrix} \int_{S_2} L_1 L_1 dx dy & \int_{S_2} L_1 L_2 dx dy \\ \int_{S_2} L_2 L_1 dx dy & \int_{S_2} L_2 L_2 dx dy \end{pmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} = \frac{1}{6} mes S_2 \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}$$

Этот вектор поправок в правую часть позволяет учесть не только вторые краевые условия, но и часть  $\beta u_\beta$  из третьих. Осталось рассмотреть матрицу поправок в левую часть:

$$I_3 = \beta \int_{S_3} \phi_i \phi_j dx dy$$

Очевидно, что получится та же матрица, только не умноженная на вектор констант.

$$I_3 = \frac{1}{6} mes S_3 \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

Добавляя эту матрицу в левую часть, на места соответствующие номерам узлов, получаем учет третьих краевых условий.

## 2 Текст программы

### main.cpp

```

1  #include "argparse/argparse.hpp"
2  #include "timer/cxxtimer.hpp"
3  #include "LOS/LOS.hpp"
4  #include "FEM.hpp"
5
6  #include <iostream>
7  #include <optional>
8  #include <fstream>
9
10 int main(int argc, char* argv[]) {
11     using namespace      ::Log;
12     using ::std::chrono::milliseconds;
13
14     argparse::ArgumentParser _program("FEM", "1.0.0");
15     _program.add_argument("-i", "--input")
16         .help("path to input files" )
17         .required();
18
19     _program.add_argument("-o", "--output")
20         .help("path to output files");

```

```

21
22 try {
23     _program.parse_args(argc, argv);
24
25     std::optional _opt          = _program.present("-o");
26     std::filesystem::path _input = _program.get<std::string>("-i");
27     std::filesystem::path _output =
28         _opt.has_value() ?
29             _program.get<std::string>("-o") :
30             _input / "sparse";
31
32     Function::setFunction(_input.string());
33
34     cxxtimer::Timer _timer(true);      /// start timer
35     FEM _FEM(_input);                 /// start FEM
36     LOS<double> _LOS (
37         _FEM.takeDate(),               /// data
38         _FEM.getNodes(),               /// count nodes
39         1E-16, 1000);                  /// epsilon and max iteration
40     _LOS.solve(Cond::HOLLESKY, true);   /// solve LOS + DIAGONAL
41     _timer.stop();                     /// stop timer
42
43     #if DEBUG != 0
44         _FEM.printAll();                /// print input FEM data
45         _FEM.printSparse();             /// print sparse format
46         _LOS.printX(14);                /// print solution vector
47         _FEM.printAnalytics();          /// print analicals solve
48     #endif
49
50     std::cout << std::scientific << 0.1341234 << std::endl;
51     std::cout << "Milliseconds: " << _timer.count<milliseconds>();
52
53 } catch(const std::runtime_error& err) {
54     Logger::append(getLog("argc != 2 (FEM --input ./input)"));
55     std::cerr << err.what();
56     std::cerr << _program;
57     std::exit(1);                      /// program error
58 }
59 return 0;
60 }

```

### 3 Тестирование

#### Тест №1

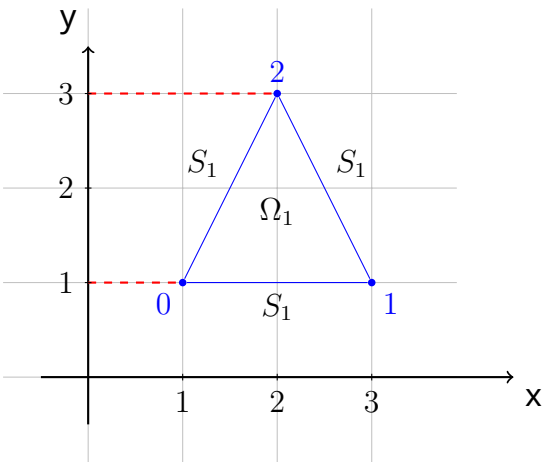
$$u(x, y) = 1$$

$$f(x, y) = 0$$

$$\lambda = 1$$

$$\gamma = 0$$

$$\beta = 0$$



nodes	elems	area	bords
1 1	0 1 2	0	0 0 1 1 0
3 1			0 1 2 1 0
2 3			0 2 0 1 0

$x$	$x^*$	$x^* - x$	$\ x^* - x\ $
1.000	1.000	0.00E+00	
1.000	1.000	0.00E+00	0.00E+00
1.000	1.000	0.00E+00	

### 4 Выводы

gdfgdfgddgdgd