

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Долгих Т. Ф., Ширяева Е. В.

**Электронное учебное пособие
«Проекционно-разностные методы»**

(для направлений подготовки 01.03.01 «Математика»,
01.03.02 «Прикладная математика и информатика»,
01.03.03 «Механика и математическое моделирование»,
02.03.02 «Фундаментальная информатика и информационные технологии»)

Ростов–на–Дону
2017

Институт математики, механики и компьютерных наук им. И. И. Воровича
ФГАОУВО «Южный федеральный университет»



Пособие подготовлено сотрудниками кафедры вычислительной математики и математической физики Института математики, механики и компьютерных наук им. И. И. Воровича ФГАОУВО «Южный федеральный университет» Татьяной Фёдоровной Долгих, Еленой Владимировной Ширяевой

Работа поддержана Базовой частью государственного задания
Министерства образования и науки РФ № 1.5169.2017/8.9

В пособии приведены необходимые сведения о проекционных методах, достаточные для их численной реализации. Изложение материала даётся на простых примерах, позволяющих получить ясное представление о методах, среди которых классический метод Ритца, метод Бубнова–Галеркина, метод Петрова–Галеркина, метод наименьших квадратов и метод конечных элементов (МКЭ), последнему в пособии уделяется наибольшее внимание — указаны способы выбора базисных функций, позволяющих упростить решение систем линейных уравнений, возникающих при реализации метода.

Наряду с проекционными методами рассмотрены и конечно-разностные методы, совпадающие с проекционно-разностными схемами. Коротко описан пакет FreeFem++, позволяющий эффективно решать пространственно двухмерные стационарные краевые задачи.

Рекомендуется студентам естественных факультетов университета.

© Т. Ф. Долгих, Е. В. Ширяева

Электронное пособие подготовлено в системе X₃TEX

Содержание

1	Проекционные методы	5
1.1	Варианты проекционных методов	6
1.1.1	Пример. Метод Бубнова–Галеркина	7
1.2	Классический метод Ритца	9
1.3	Метод наименьших квадратов	11
1.4	Индивидуальное задание № 1	12
1.5	Введение в метод конечных элементов	16
1.5.1	Вариационная формулировка задачи	16
1.5.2	Численная реализация	17
1.6	Метод конечных элементов. Пример	18
1.6.1	Построение приближенного слабого решения задачи	19
1.6.2	Построение приближенного сильного решения задачи. Метод Галеркина	21
1.6.3	Конечно-разностный метод	22
1.7	Индивидуальное задание № 2	23
1.8	Выбор базисных функций	28
1.8.1	Краевые условия и сохранение свойств исходной задачи	28
1.8.2	Финитные базисные функции	31
1.8.3	Вычисление элементов матрицы A_{ik} и вектора b_i	34
1.9	Естественные и главные краевые условия	37
1.10	Кубические конечные элементы	42
1.10.1	Пример	44
2	Метод конечных элементов в двумерном случае	46
2.1	Задача Дирихле для уравнения Лапласа. Слабое решение	46
2.2	Построение приближенного решения	48
2.3	Конечно-разностные методы в двумерном случае	52
2.4	Индивидуальное задание № 3	54

3	Использование пакета FreeFem++ для решения задач	58
3.1	Задача о стационарном распределении температуры .	58
3.1.1	Постановка задачи	58
3.1.2	Слабая формулировка задачи	59
3.1.3	Слабая формулировка задачи на языке FreeFem++	61
3.1.4	Задание области	63
3.1.5	Полный код программы	64
3.2	Задача о распределении температуры в областях сложной формы	66
3.2.1	Решение задачи для круга	66
3.2.2	Распределение температуры в четырехугольнике	68
3.2.3	Решение задачи в криволинейной области . . .	69
3.2.4	Решение задачи в области с отверстием	70
3.2.5	Решение задачи с переменной теплопроводностью	70
3.3	Индивидуальное задание № 4	72
	Литература	73
	Предметный указатель	76
	Об авторах	77
	Интерфейс пользователя	78

1 Проекционные методы

Проекционные методы являются популярными и эффективным методом численного и аналитического решения различных задач математической физики. Привести список огромного количества литературы, посвященной основам метода и проблемам, связанным с его применением, — практически невыполнимая задача. Перечислим лишь некоторые из монографий и учебников по численным методам [4–8, 10–19]. Сравнительная простота реализации проекционных методов, в частности, метода конечных элементов (МКЭ) привела к созданию большого количества программного обеспечения (см. например, [1–3, 20, 21]), позволяющего использовать методы без специальной математической подготовки.

В пособии даются лишь самые необходимые сведения о проекционных методах, достаточные для их численной реализации. Читателю достаточно лишь знакомства с основами уравнений математической физики, линейной алгебры и численных методов. При необходимости все специальные понятия приводятся непосредственно в тексте. Изложение материала даётся на простых примерах, позволяющих получить ясное представление о методах, и, в основном, следует [1, 2, 4, 5, 7]. Конечно же рассмотрены лишь некоторые проекционные методы, среди которых классический метод Ритца, метод Бубнова–Галеркина, метод Петрова–Галеркина, метод наименьших квадратов и метод конечных элементов. Именно МКЭ в пособии уделяется наибольшее внимание. Собственно говоря, наиболее важной особенностью МКЭ является выбор в качестве базисных функций, так называемых, финитных базисных функций, что, в свою очередь, позволяет существенно упростить решение систем линейных уравнений, возникающих при реализации метода.

Скажем несколько слов об используемой терминологии. Несмотря на то что метод конечных элементов известен очень давно, до сих пор в литературе используются различные термины. Так, сам метод называется — метод конечных элементов, вариационно-разностный метод,

проекционно-сеточный метод и т. д. Для названия базисных функций используются термины: тестовые функции, пробные функции и т. п. Это связано с широкой применимостью метода для решения различного рода задач — задач теории упругости, гидродинамики и переноса, электродинамики, инженерных задач и пр. В каждой из упомянутых областей принята своя специфическая терминология, зачастую не очень математически строгая.

1.1 Варианты проекционных методов

Пусть E, F — некоторые гильбертовы пространства. Рассмотрим уравнение

$$Lu = f, \quad (1.1)$$

где L — линейный оператор, с областью определения $D(L) \subset E$ и областью значений $R(L) \subset F$.

Зададим две последовательности подпространств $\{E_n\}, \{F_n\}$

$$E_n \subset D(L) \subset E, \quad F_n \subset R(L) \subset F, \quad n = 1, 2, \dots \quad (1.2)$$

Определим операторы проектирования P_n , которые проектируют F на F_n , то есть

$$P_n^2 = P_n, \quad P_n F = F_n, \quad n = 1, 2, \dots \quad (1.3)$$

Проекционный метод заключается в замене уравнения (1.1) уравнением

$$P_n(Lu_n - f) = 0, \quad u_n \in E_n. \quad (1.4)$$

Здесь $u_n \in E_n$ означает, что решение уравнения (1.4) разыскивается в подпространстве E_n .

Наиболее распространенный метод построения подпространств E_n, F_n и проектора P_n , по крайней мере, при численной реализации метода, заключается в следующем (см., например, [6, С. 190]) — задаются две так называемые координатные последовательности ϕ_i, ψ_i

$$\phi_i \in D(L) \subset E, \quad \psi_i \in F, \quad i = 1, 2, \dots \quad (1.5)$$

Приближенное решение u_n разыскивается в виде

$$u_n = \sum_{i=1}^n c_i \phi_i, \quad (1.6)$$

где c_i — некоторые неизвестные постоянные коэффициенты.

Проектированию (1.4) соответствует ортогональность невязки $Lu_n - f$ элементам координатной последовательности ψ_k

$$(Lu_n - f, \psi_k) = 0, \quad k = 1, 2, \dots, n. \quad (1.7)$$

Разумеется, что скобки здесь означают скалярное произведение в гильбертовом пространстве. После подстановки (1.6) в (1.7) имеем

$$\sum_{i=1}^n (L\phi_i, \psi_k) c_i = (f, \psi_k), \quad k = 1, 2, \dots, n. \quad (1.8)$$

Таким образом для определения коэффициентов c_i и, следовательно, решения u_n имеем систему линейных алгебраических уравнений.

Выбор ϕ_i, ψ_i можно осуществлять различными способами. Конечно, эти функции должны быть линейно независимыми и образовывать полные системы. Принято различать следующие способы выбора функций (см., например, [4–6]).

1. Пространства E и F совпадают, то есть $E = F$, и $\phi_i = \psi_i$. В этом случае проекционный метод называется методом Бубнова–Галеркина (или Галеркина).

2. Последовательности ϕ_i, ψ_i различны. В этом случае проекционный метод называется методом Петрова–Галеркина.

3. $\psi_i = L\phi_i$. В этом случае проекционный метод называется методом наименьших квадратов.

Теоремы о сходимости проекционных методов приведены, например, в [4–6]). Некоторые конкретные способы выбора ϕ_i, ψ_i указаны ниже.

1.1.1 Пример. Метод Бубнова–Галеркина

Рассмотрим первую краевую задачу для уравнения

$$-\frac{d^2 u(x)}{dx^2} + u(x) = 1, \quad 0 < x < 1, \quad (1.9)$$

с краевыми условиями

$$u(0) = 1, \quad u(1) = 0. \quad (1.10)$$

В данном случае, пространство E — это пространство дважды дифференцируемых функции на отрезке $x \in [0, 1]$, удовлетворяющих условиям (1.10). Оператор L и функция f имеют вид

$$L \equiv -\frac{d^2}{dx^2} + 1, \quad f(x) \equiv 1. \quad (1.11)$$

Определим скалярное произведение

$$(u, v) = \int_0^1 u(x)v(x) dx. \quad (1.12)$$

Выберем набор координатных (базисных) функций

$$\phi_k(x) = \sin k\pi x, \quad k = 1, \dots, n, \quad (1.13)$$

задающих пространство E_n , то есть,

$$u_n(x) = \sum_{k=1}^n c_k \phi_k(x) \in E_n. \quad (1.14)$$

Полагая $\psi_k(x) = \phi_k(x)$, то есть используя метод Бубнова–Галеркина, вычислим соотношения (см.(1.8))

$$L\phi_k(x) = (\pi^2 k^2 + 1)\phi_k, \quad (1.15)$$

$$(L\phi_i, \phi_k) = \int_0^1 \phi_i(x)\phi_k(x) dx = 0, \quad k \neq i, \quad (1.16)$$

$$(L\phi_k(x), \phi_k) = \int_0^1 \phi_k^2(x) dx = \frac{1}{2}(\pi^2 k^2 + 1), \quad (1.17)$$

$$(f, \phi_k) = \int_0^1 \phi_k(x) dx = \frac{1 - (-1)^k}{\pi k}. \quad (1.18)$$

Таким образом, матрица $(L\phi_i, \psi_k)$ системы (1.8) является диагональной и решение уравнений (1.8) имеет вид

$$c_k = 2 \frac{1 - (-1)^k}{\pi k (\pi^2 k^2 + 1)}, \quad k = 1, \dots, n. \quad (1.19)$$

Окончательно, приближенное решение задачи (1.9), (1.10) записывается в форме

$$u_n(x) = 2 \sum_{k=1}^n \frac{1 - (-1)^k}{\pi k (\pi^2 k^2 + 1)} \sin k\pi x. \quad (1.20)$$

Заметим, что при вычислениях решения по формуле (1.20), конечно, следует учитывать, что коэффициенты $c_{2m} = 0$ (с четными индексами).

1.2 Классический метод Ритца

Рассмотрим функционал $J(u)$

$$J(u) = (Lu, u) - 2(f, u), \quad L: E \rightarrow R, \quad (1.1)$$

где L — линейный самосопряженный положительный оператор ($L > 0$, $L = L^*$), f — заданная функция, E — некоторое (для определенности) гильбертово пространство, R — пространство вещественных чисел.

Решение задачи о нахождении минимума функционала $J(u)$ эквивалентно решению уравнения

$$Lu = f. \quad (1.2)$$

Действительно, пусть u_0 является функцией, в которой функционал имеет минимум

$$\min_{u \in E} J(u) = J(u_0). \quad (1.3)$$

Используя методы вариационного исчисления, рассмотрим функцию вещественного переменного β

$$\Phi(\beta) \equiv J(u_0 + \beta v), \quad v \in E. \quad (1.4)$$

Вычислим производные

$$\frac{d\Phi(\beta)}{d\beta} = (Lv, u_0 + \beta v) + (Lu_0 + \beta Lv, v) - 2(f, v), \quad (1.5)$$

$$\frac{d^2\Phi(\beta)}{d\beta^2} = 2(Lv, v).$$

Полагая $\beta = 0$, с учетом самосопряженности оператора (и вещественности скалярного произведения)

$$(Lv, u_0) = ((v, L^*u_0) = (v, Lu_0) = (Lu_0, v), \quad (1.6)$$

получим

$$\frac{d\Phi(0)}{d\beta} = (Lv, u_0) + (Lu_0, v) - 2(f, v) = 2(Lu_0 - f, v). \quad (1.7)$$

В силу положительности оператора L , имеем

$$\frac{d^2\Phi(0)}{d\beta^2} = 2(Lv, v) > 0. \quad (1.8)$$

Условие экстремума (минимума с учетом (1.8)) имеет вид

$$\frac{d\Phi(0)}{d\beta} = 2(Lu_0 - f, v) = 0. \quad (1.9)$$

Таким образом, u_0 является решением уравнения (1.2).

Метод Ритца заключается в поиске функции $u_n \in E_n$, минимизирующий функционал $J(u)$ в пространстве E_n

$$\min_{u_n \in E_n} J(u_n). \quad (1.10)$$

Пусть $\{\phi_k\}$ — базисные функции в E_n , то есть

$$u_n = \sum_{k=1}^n c_k \phi_k. \quad (1.11)$$

Тогда имеем функцию переменных c_1, c_1, \dots, c_n ,

$$\Phi(c_1, \dots, c_n) = J\left(\sum_{k=1}^n c_k \phi_k\right) \quad (1.12)$$

или с учетом (1.1)

$$\Phi(c_1, \dots, c_n) = \left(L \sum_{k=1}^n c_k \phi_k, \sum_{k=1}^n c_k \phi_k\right) - 2\left(f, \sum_{k=1}^n c_k \phi_k\right). \quad (1.13)$$

Вычисляя производные и приравнивая их нулю, выводим

$$\frac{\partial \Phi}{\partial c_i} = 2 \sum_{k=1}^n (L\phi_k, \phi_i) c_k - 2(f, \phi_i) = 0. \quad (1.14)$$

Таким образом, для определения c_k имеем систему линейных уравнений

$$\sum_{k=1}^n (L\phi_k, \phi_i) c_k = (f, \phi_i), \quad i = 1, \dots, n, \quad (1.15)$$

которая для самосопряженного оператора L в случае $\phi_k = \psi_k$ полностью совпадает с (1.8).

Другими словами, в случае самосопряженного оператора метод Ритца совпадает с методом Бубнова–Галеркина.

1.3 Метод наименьших квадратов

Метод метод наименьших квадратов заключается в поиске функции $u_n \in E_n$, минимизирующий функционал $J(u)$

$$J(u) = \|Lu - f\|^2 = (Lu - f, Lu - f) \quad (1.1)$$

в пространстве E_n , то есть (сравни с (1.10))

$$\min_{u_n \in E_n} J(u_n). \quad (1.2)$$

Аналогично (1.10)–(1.14) имеем

$$F(c_1, \dots, c_n) = J \left(\sum_{k=1}^n c_k \phi_k \right), \quad (1.3)$$

$$F(c_1, \dots, c_n) = \left(\sum_{k=1}^n c_k L\phi_k - f, \sum_{k=1}^n c_k L\phi_k - f \right), \quad (1.4)$$

$$\frac{\partial F}{\partial c_i} = \left(L\phi_i, \sum_{k=1}^n c_k L\phi_k - f \right) + \left(\sum_{k=1}^n c_k L\phi_k - f, L\phi_i \right). \quad (1.5)$$

Если пространство вещественное, то условие экстремума функционала принимает форму

$$\frac{\partial F}{\partial c_i} = 2 \left(\sum_{k=1}^n c_k L\phi_k - f, L\phi_i \right) = 0, \quad i = 1, \dots, n, \quad (1.6)$$

что в точности соответствует (1.8) при выборе $\psi_k = L\phi_k$, то есть методу Петрова–Галеркина.

1.4 Индивидуальное задание № 1

Рассматривается линейная краевая задача для дифференциального уравнения второго порядка

$$\begin{cases} y''(x) + p(x)y'(x) + q(x)y(x) = f(x), & a < x < b, \\ y(a) = 0, & y(b) = 0. \end{cases} \quad (1.7)$$

Требуется найти приближённо–аналитическое решение

$$y_n(x) = \sum_{k=1}^n C_k \varphi_k(x)$$

для задачи (1.7) на отрезке $[a, b]$ указанными в индивидуальном варианте проекционными методами.

Указание 1. Для определения точного решения краевой задачи можно воспользоваться встроенными командами пакета Maple:

```
p := x -> ... ;
q := x -> ... ;
ode := diff(y(x), x$2) + p(x)*diff(y(x), x) + q(x)*y(x) = f(x);
ics := y(a) = 0, y(b) = 0;
dsolve([ode, ics], y(x));
```

Указание 2. При решении заданий использовать базисные функции в виде

$$\varphi_k(x) = x^k(x - a)(b - x), \quad k = 1, 2, \dots, n,$$

или в виде

$$\varphi_k(x) = \sin\left(\frac{\pi k(x - a)}{b - a}\right), \quad k = 1, 2, \dots, n,$$

где $a \leq x \leq b$.

Указание 3. Для сравнения полученных приближённых решений $y_1^n(x)$ и $y_2^n(x)$ с точным $y^t(x)$ нужно построить в одной системе координат три графика функций и таблицу с данными в виде

x_i	$y^t(x_i)$	$y_1^n(x_i)$	Δ_1	$y_2^n(x_i)$	Δ_2
...

где

$$x_i = a + i \frac{b-a}{m}, \quad m = 20, 30, \dots, \quad i = 0, 1, \dots, m,$$

$$\Delta_1 = |y^t(x_i) - y_1^n(x_i)|, \quad \Delta_2 = |y^t(x_i) - y_2^n(x_i)|.$$

Вариант 1. Решить первую краевую задачу

$$\begin{cases} -y'' + y = 2x, & 0 < x < 1, \\ y(0) = 0, & y(1) = 0, \end{cases}$$

используя методы Рунге и Петрова–Галёркина.

Вариант 2. Решить первую краевую задачу

$$\begin{cases} y'' - 3y = 8 - 9x^2, & 0 < x < 1, \\ y(0) = 0, & y(1) = 0, \end{cases}$$

используя метод Бунднова–Галёркина и метод наименьших квадратов.

Вариант 3. Решить первую краевую задачу

$$\begin{cases} y'' + y = -x, & 1 < x < 2, \\ y(1) = 0, & y(2) = 0, \end{cases}$$

используя методы Рунге и Петрова–Галёркина.

Вариант 4. Решить первую краевую задачу

$$\begin{cases} y'' = x - 1, & 0 < x < 2, \\ y(0) = 0, & y(2) = 0, \end{cases}$$

используя метод Бунднова–Галёркина и метод наименьших квадратов.

Вариант 5. Решить первую краевую задачу

$$\begin{cases} y'' + y = 4xe^x, & 0 < x < 1, \\ y(0) = 0, & y(1) = 0, \end{cases}$$

используя методы Ритца и Петрова–Галёркина.

Вариант 6. Решить первую краевую задачу

$$\begin{cases} -y'' + y = 4 \sin x, & 0 < x < 1, \\ y(0) = 0, & y(1) = 0, \end{cases}$$

используя метод Бубнова–Галёркина и метод наименьших квадратов.

Вариант 7. Решить первую краевую задачу

$$\begin{cases} y'' + y' + y = xe^{-x}, & 0 < x < 1, \\ y(0) = 0, & y(1) = 0, \end{cases}$$

используя методы Ритца и Петрова–Галёркина.

Вариант 8. Решить первую краевую задачу

$$\begin{cases} y'' + 9y = \cos 2x, & 1 < x < 2, \\ y(1) = 0, & y(2) = 0, \end{cases}$$

используя метод Бубнова–Галёркина и метод наименьших квадратов.

Вариант 9. Решить первую краевую задачу

$$\begin{cases} y'' + 3y' + 2y = \frac{1}{e^x + 1}, & 0 < x < 1, \\ y(0) = 0, & y(1) = 0, \end{cases}$$

используя методы Ритца и Петрова–Галёркина.

Вариант 10. Решить первую краевую задачу

$$\begin{cases} -y'' + y = -x \cos x, & 1 < x < 2, \\ y(1) = 0, & y(2) = 0, \end{cases}$$

используя метод Бубнова–Галёркина и метод наименьших квадратов.

Вариант 11. Решить первую краевую задачу

$$\begin{cases} y'' - y' = 2e^x + x, & 0 < x < 1, \\ y(0) = 0, & y(1) = 0, \end{cases}$$

используя методы Ритца и Петрова–Галёркина.

Вариант 12. Решить первую краевую задачу

$$\begin{cases} y'' - 5y' = x^2 \sin x, & 1 < x < 3, \\ y(1) = 0, & y(3) = 0, \end{cases}$$

используя метод Бубнова–Галёркина и метод наименьших квадратов.

1.5 Введение в метод конечных элементов

Рассмотрим уравнение в гильбертовом пространстве

$$Lu = f, \quad L : E \rightarrow F, \quad (1.1)$$

решение которого называется сильным решением.

Пусть задан билинейный (то есть линейный по обоим переменным) функционал

$$A(u, v) \equiv (Lu - f, v). \quad (1.2)$$

Выражение вида

$$(Lu - f, v) = 0, \quad \forall v \in F, \quad (1.3)$$

называется слабой формулировкой уравнения (1.1), а функция, удовлетворяющая (1.3) называется слабым решением уравнения (1.1).

Очевидно, что из существования сильного решения следует существование слабого решения. Обратное утверждение, в общем случае, не является верным. Для того чтобы из существования слабого решения вытекало существование сильного решения требуется, по крайней мере, выполнение условий теоремы Ф. Рисса об ортогональной проекции.

1.5.1 Вариационная формулировка задачи

В случае, когда оператор L в уравнении (1.1) является самосопряженным, положительно определенным и пространства E, F совпадают

$$L = L^*, \quad L > 0, \quad F = E, \quad (1.4)$$

возможна вариационная трактовка появления слабой формы задачи (1.1), то есть (1.3). В связи с этим, часто, взамен термина слабая формулировка используется термин вариационная формулировка.

В п. 1.2 указано, что в случае выполнения условий (1.1) нахождение решения задачи (1.1) эквивалентно отысканию минимума функционала

$$J(u) = (Lu, u) - 2(f, u). \quad (1.5)$$

Пусть u_0 — функция, доставляющая минимум функционалу $J(u)$

$$\min_{u \in E} J(u) = J(u_0). \quad (1.6)$$

Тогда для любых значений параметра β выполнено неравенство

$$J(u_0) \leq J(u_0 + \beta v), \quad v \in E. \quad (1.7)$$

Используя (1.5), с учетом (1.4)

$$\begin{aligned} J(u_0) &\leq (L(u_0 + \beta v), u_0 + \beta v) - 2(f, u_0 + \beta v) = \\ &= (Lu_0, u_0) + 2\beta(Lu_0, v) + \beta^2(Lv, v) - 2(f, u_0) - 2\beta(f, v) = \\ &= J(u_0) + 2\beta(Lu_0 - f, v) + \beta^2(Lv, v) \end{aligned} \quad (1.8)$$

или

$$2\beta(Lu_0 - f, v) + \beta^2(Lv, v) \geq 0. \quad (1.9)$$

В силу положительности оператора $(Lv, v) > 0$ и произвольности функции v и параметра β неравенство (1.9) выполняется при условии

$$(Lu_0 - f, v) = 0, \quad \forall v, \quad (1.10)$$

что совпадает с (1.3).

Напомним, что $(Lu_0 - f, v)$ фактически является первой вариацией функционала $J(u)$ при $u = u_0$.

1.5.2 Численная реализация

Приближенное (не обязательно численное) слабое решение, то есть решение, уравнения (1.3), задавая набор координатных функций ϕ_m , разыскиваем в виде

$$u = \sum_{k=1}^n c_k \phi_k \in E_n, \quad (1.11)$$

Тогда определение слабого решения сводится к нахождению решения системы алгебраических уравнений (сравни с (1.15))

$$\sum_{k=1}^n (L\phi_k, \phi_i) c_k = (f, \phi_i), \quad i = 1, \dots, n, \quad (1.12)$$

или с учетом введенного обозначения (1.2)

$$\sum_{k=1}^n A(\phi_k, \phi_i) c_k = (f, \phi_i), \quad i = 1, \dots, n. \quad (1.13)$$

1.6 Метод конечных элементов. Пример

Рассмотрим первую краевую задачу (задачу Дирихле) для определения функции $u(x)$

$$-\frac{d}{dx} \left(p(x) \frac{du(x)}{dx} \right) + q(x)u(x) = f(x), \quad 0 < x < 1, \quad (1.1)$$

$$u(0) = 0, \quad u(1) = 0, \quad (1.2)$$

где $p(x)$, $q(x)$, $f(x)$ — известные функции.

Предположим, что функции $q(x)$, $f(x)$ непрерывны, а функция $p(x)$ — непрерывно дифференцируема. Решение задачи (1.1), (1.2), являющееся дважды непрерывно дифференцируемой функцией называется, как уже говорилось, **сильным решением**.

Можно попытаться переформулировать задачу (1.1), (1.2) с целью снижения требований, накладываемых на непрерывность функций $p(x)$, $q(x)$, $f(x)$ и функции $u(x)$. Для этого умножим уравнение (1.1) на некоторую функцию $v(x)$ и проинтегрируем на отрезке $[0, 1]$

$$-\int_0^1 \frac{d}{dx} \left(p(x) \frac{du(x)}{dx} \right) v(x) dx + \int_0^1 q(x)u(x)v(x) dx = \int_0^1 f(x)v(x) dx.$$

Используя формулу интегрирования по частям, имеем

$$\int_0^1 p \frac{du}{dx} \frac{dv}{dx} dx - p \frac{du}{dx} v \Big|_0^1 + \int_0^1 quv dx = \int_0^1 fv dx.$$

Потребуем, чтобы функция $v(x)$ удовлетворяла тем же краевым условиям, что и $u(x)$, т. е.

$$v(0) = 0, \quad v(1) = 0. \quad (1.3)$$

Тогда

$$\int_0^1 \left(p \frac{du}{dx} \frac{dv}{dx} + quv - fv \right) dx = 0, \quad v(0) = 0, \quad v(1) = 0. \quad (1.4)$$

Если соотношение (1.4) выполнено для любых $v(x)$, то оно, как уже говорилось, называется **слабой формой** (или вариационной формой)

задачи (1.1), (1.2). На самом деле, следует дополнительно указать какому классу принадлежат функции $v(x)$, например, $v \in C^1$.

Сам способ получения соотношения (1.4) показывает, что если функция $u(x)$ является решением задачи (1.1), (1.2), то эта же функция будет решением (1.4). Обратное же утверждение в общем случае неверно. Это видно уже из того, что для функций $p(x)$, $q(x)$, $f(x)$ и $u(x)$, входящих в (1.4), можно требовать гораздо меньших ограничений на гладкость, чем для соответствующих функций задачи (1.1), (1.2). Так, например, можно считать функции $p(x)$, $q(x)$, $f(x)$ непрерывными

$$(p, q, f \in C^0),$$

а функции $u(x)$, $v(x)$ непрерывно дифференцируемыми

$$(u, v \in C^1).$$

На самом деле, можно даже требовать от $p(x)$, $q(x)$, $f(x)$ кусочной непрерывности, а от $u(x)$, $v(x)$ — кусочной гладкости. Заметим также, что для функции $u(x)$, входящей в (1.4), не требуется даже выполнения краевых условий (1.2).

1.6.1 Построение приближенного слабого решения задачи

Функция $u(x)$, являющаяся решением задачи (1.4), как уже говорилось, называется **слабым решением** задачи (1.1), (1.2). Разыскиваем приближенное решение задачи (1.4) в виде

$$u^h(x) = \sum_{k=1}^{n-1} c_k \varphi_k(x), \quad (1.5)$$

где $u^h(x)$ — приближенное решение, функции $\varphi_k(x)$ предполагаются известными, линейно независимыми и называются **базисными функциями** (другие названия — **тестовые функции**, **пробные функции**), c_k — коэффициенты, подлежащие определению.

Формула (1.5) задает аппроксимацию функции $u(x)$ в виде некоторого сужения (проектирования) на **конечномерное** пространство,

определяемое базисом $\varphi_k(x)$, $k = 1, \dots, n - 1$. Заметим, что величина u^h при заданных $\varphi_k(x)$ полностью определяется набором чисел

$$(c_1, c_2, \dots, c_{n-1}),$$

т. е. можно считать, что $u^h = (c_1, c_2, \dots, c_{n-1})$ является вектором, принадлежащим \mathbb{R}^{n-1} .

Подставляя (1.5) в (1.4), получим

$$\int_0^1 \left(p \sum_{k=1}^{n-1} c_k \varphi'_k v' + q \sum_{k=1}^{n-1} c_k \varphi_k v - f v \right) dx = 0.$$

Учитывая, что (1.4) должно выполняться для любых $v(x)$, принадлежащих некоторому классу, выберем в качестве $v(x)$ некоторый набор функций, а именно $v(x) = \varphi_i(x)$, $i = 1, \dots, n - 1$. В соответствии с требованиями $v(0) = 0$, $v(1) = 0$ (см. (1.4)) будем считать, что

$$\varphi_k(0) = 0, \quad \varphi_k(1) = 0, \quad k = 1, \dots, n - 1. \quad (1.6)$$

В этом случае имеем

$$\int_0^1 \left(p \sum_{k=1}^{n-1} c_k \varphi'_k \varphi'_i + q \sum_{k=1}^{n-1} c_k \varphi_k \varphi_i - f \varphi_i \right) dx = 0, \quad i = 1, \dots, n - 1.$$

Перепишем это соотношение в виде

$$\sum_{k=1}^{n-1} A_{ik} c_k = b_i, \quad i = 1, \dots, n - 1, \quad (1.7)$$

где введены следующие обозначения

$$A_{ik} = \int_0^1 (p \varphi'_i \varphi'_k + q \varphi_i \varphi_k) dx, \quad b_i = \int_0^1 f \varphi_i dx, \quad i, k = 1, \dots, n - 1. \quad (1.8)$$

Таким образом, для определения c_k имеем систему линейных алгебраических уравнений. Решая эту систему и подставляя c_k в (1.5), получим приближенное решение задачи (1.4). Подчеркнем, что приближенное решение (1.5) в силу условий (1.6) автоматически удовлетворяет краевым условиям (1.2) для исходной задачи.

1.6.2 Построение приближенного сильного решения задачи. Метод Галеркина

Разыскивать решение в виде (1.5) можно и для исходной задачи. Подставляя (1.5) в (1.1), имеем

$$\delta(x) = -\frac{d}{dx} \left(p \sum_{k=1}^{n-1} c_k \varphi'_k \right) + q \sum_{k=1}^{n-1} c_k \varphi_k - f$$

или

$$\delta(x) = -p' \sum_{k=1}^{n-1} c_k \varphi'_k - p \sum_{k=1}^{n-1} c_k \varphi''_k + q \sum_{k=1}^{n-1} c_k \varphi_k - f. \quad (1.9)$$

Здесь $\delta(x)$ — невязка, возникающая после подстановки приближенного решения $u^h(x)$ в уравнение (1.1).

Умножим соотношение (1.9) на φ_i и, проинтегрировав на $[0, 1]$, потребуем выполнения равенств

$$\int_0^1 \delta(x) \varphi_i(x) dx = - \int_0^1 \left(\sum_{k=1}^{n-1} c_k (p' \varphi'_k + p \varphi''_k - q \varphi_k) + f \right) \varphi_i dx = 0$$

или

$$\sum_{k=1}^{n-1} c_k \int_0^1 (-p' \varphi'_k \varphi_i - p \varphi''_k \varphi_i + q \varphi_k \varphi_i) dx - \int_0^1 f \varphi_i dx = 0, \quad (1.10)$$

$$i = 1, \dots, n - 1.$$

Таким образом, вновь получена система линейных алгебраических уравнений для определения c_k

$$\sum_{k=1}^{n-1} \tilde{A}_{ik} c_k = b_i, \quad i = 1, \dots, n - 1,$$

где

$$\tilde{A}_{ik} = \int_0^1 (-p' \varphi'_k \varphi_i - p \varphi''_k \varphi_i + q \varphi_k \varphi_i) dx, \quad b_i = \int_0^1 f \varphi_i dx. \quad (1.11)$$

Заметим, что функция (1.5) будет являться приближением решения исходной задачи (1.1), (1.2), если на базисные функции $\varphi_k(x)$ будут

наложены дополнительные ограничения (1.6) — требования удовлетворения граничным условиям (1.2).

Укажем на основное различие между матрицами A_{ik} и \tilde{A}_{ik} . Для вычисления матрицы A_{ik} (см. (1.8)) от функций φ_k требуется, чтобы φ'_k была кусочно-непрерывной. В случае матрицы \tilde{A}_{ik} (см. (1.11)) на функции φ_k следует накладывать более сильные ограничения — требуется, чтобы φ''_k была кусочно-непрерывной. Конечно, имеется в виду, что матрица \tilde{A}_{ik} вычисляется непосредственно по приведенной формуле и операция интегрирования по частям в (1.11) не используется. Если же в (1.11) произвести операцию интегрирования по частям, то вновь получится слабая формулировка задачи и соотношения (1.11) совпадут с (1.7).

1.6.3 Конечно-разностный метод

Описанные приемы построения приближенного решения, конечно же, не являются специфическими способами решения исходной задачи. Приведем, в частности, конечно-разностный (сеточный) метод, важный для дальнейших целей. Для простоты ограничимся случаем, когда $p(x) = 1$, $q(x) = 0$, т. е. задачей

$$-u''(x) = f(x), \quad 0 < x < 1, \quad u(0) = 0, \quad u(1) = 0. \quad (1.12)$$

Разобьем отрезок $[0, 1]$ на интервалы с одинаковой длиной

$$[x_k, x_{k+1}], \quad k = 0, \dots, n-1,$$

$$x_i = ih, \quad h = 1/n.$$

Рассмотрим уравнение (1.12) в точке x_k и аппроксимируем производную $u''(x_k)$ центральной конечной разностью (см. [7]). Тогда, с учетом краевых условий, для определения

$$u_k = u(x_k), \quad k = 0, \dots, n$$

получим систему линейных алгебраических уравнений

$$-u_{k-1} + 2u_k - u_{k+1} = h^2 f(x_k), \quad k = 1, \dots, n-1, \quad (1.13)$$

$$u_0 = 0, \quad u_n = 0.$$

Матрица коэффициентов для этой системы имеет трехдиагональный вид

$$\begin{pmatrix} 2 & -1 & 0 & 0 & \dots & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & \dots & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & \dots & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & \dots & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \dots \\ u_{n-2} \\ u_{n-1} \end{pmatrix} = \begin{pmatrix} h^2 f(x_1) \\ h^2 f(x_2) \\ h^2 f(x_3) \\ h^2 f(x_4) \\ \dots \\ h^2 f(x_{n-2}) \\ h^2 f(x_{n-1}) \end{pmatrix}. \quad (1.14)$$

1.7 Индивидуальное задание № 2

Рассматривается линейная краевая задача для дифференциального уравнения второго порядка

$$\begin{cases} y''(x) + p(x)y'(x) + q(x)y(x) = f(x), & a < x < b, \\ \alpha y(a) + \beta y'(a) = \gamma, \\ \xi y(b) + \eta y'(b) = \mu. \end{cases} \quad (1.15)$$

Требуется найти приближённое решение в точках

$$x_i = a + ih, \quad h = \frac{b-a}{n}, \quad i = 0, 1, \dots, n$$

методом прогонки.

Указание 1. Для решения (1.15) можно воспользоваться встроенными командами пакета Maple (см. п. 1.4). Краевые условия в этом случае будут задаваться в виде:

```
ics := alpha*y(a) + beta*D(y)(a) = gamma,
      xi*y(b) + eta*D(y)(b) = mu;
```

Указание 2. Метод прогонки используется для решения разностного уравнения вида

$$A_i y_{i-1} - B_i y_i + C_i y_{i+1} = F_i, \quad i = 1, 2, \dots, n-1. \quad (1.16)$$

Рассмотрим дифференциальное уравнение и краевые условия (1.15) в точках $x = x_i$. Значения производных в уравнении заменим разделёнными разностями вида

$$y_i'' \approx \frac{y_{i-1} - 2y_i + y_{i+1}}{h^2}, \quad y_i' \approx \frac{y_{i+1} - y_{i-1}}{2h}, \quad i = 1, 2, \dots, n-1,$$

где $y_i = y(x_i)$.

Производные в краевых условиях можно аппроксимировать отношениями

$$y_i' \approx \frac{y_{i+1} - y_i}{h}.$$

Здесь $i = 0$ для первого краевого условия и $i = n - 1$ для второго краевого условия.

Таким образом, от системы (1.15) переходим к системе линейных алгебраических уравнений (1.16) с трёхдиагональной матрицей.

Указание 3. Метод прогонки состоит из двух этапов: **прямой прогонки** и **обратной прогонки**.

На этапе прямой прогонки определяются **прогоночные коэффициенты** L_i и K_i из соотношений

$$K_0 = \frac{C_0}{B_0}, \quad L_0 = -\frac{F_0}{B_0},$$

$$K_i = -\frac{C_i}{A_i K_{i-1} - B_i}, \quad L_i = \frac{F_i - A_i L_{i-1}}{A_i K_{i-1} - B_i}, \quad i = 1, 2, \dots, n.$$

На этапе обратной прогонки вычисляются решения y_i с помощью **прогоночных формул**

$$y_n = \frac{F_n - A_n L_{n-1}}{A_n K_{n-1} - B_n}, \quad y_{i-1} = K_{i-1} y_i + L_{i-1}, \quad i = n, n-1, \dots, 1.$$

Указание 4. Расчёты по методу прогонки зависят от выбора шага h . Подбор оптимального шага выполняем по следующему алгоритму: выбираем начальный шаг \tilde{h} и вычисляем решение y_i^1 в точках $a + i\tilde{h}$, где $i = 0, 1, \dots, \tilde{n}$. Затем полагаем, что $h = \frac{\tilde{h}}{2}$ и находим значения y_i^2 . Сравниваем решения y_i^1 и y_i^2 в одних и тех же точках. Если для некоторого

заданного значения ε выполняется условие

$$\delta = \|y^1 - y^2\| < \varepsilon,$$

то решение найдено. Иначе делим шаг ещё раз пополам, переписываем y^2 в y^1 , а y^2 пересчитываем с новым шагом. Дробить шаг следует не более четырёх раз.

Указание 5. Полученное приближённое решение y^2 сравнить с точным решением y_t . Для этого нужно построить графики функций в одной системе координат и проверить условие

$$\delta = \|y_t - y^2\| < \varepsilon.$$

Для построения полученного численного решения (X, Y) , $X = \{x_i\}$, $Y = \{y_i^2\}$, $i = 0, 1, \dots, n$ можно воспользоваться командами в Maple

```
xy := zip((x,y)->[x,y],convert(X,list),convert(Y,list));
plot(xy);
```

Вариант 1. Методом прогонки найти с точностью $\varepsilon = 0.1$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' - 3y' + 2y = x \sin x, & -2 < x < -1, \\ y(-2) - y'(-2) = 0, & y(-1) = 0. \end{cases}$$

Вариант 2. Методом прогонки найти с точностью $\varepsilon = 0.05$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' + 9y = \cos 2x, & 0 < x < \frac{\pi}{2}, \\ y(0) - y'(0) = -1, & y\left(\frac{\pi}{2}\right) + y'\left(\frac{\pi}{2}\right) = 1. \end{cases}$$

Вариант 3. Методом прогонки найти с точностью $\varepsilon = 0.015$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' - 5y' = 3x^2 + \sin 5x, & -\frac{\pi}{2} < x < \frac{\pi}{2}, \\ y\left(-\frac{\pi}{2}\right) = 0.02, & y'\left(\frac{\pi}{2}\right) = -0.03. \end{cases}$$

Вариант 4. Методом прогонки найти с точностью $\varepsilon = 0.04$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' + 9y = \sin 2x, & 0 < x < \frac{\pi}{4}, \\ y(0) = -1, & y' \left(\frac{\pi}{4} \right) = 0. \end{cases}$$

Вариант 5. Методом прогонки найти с точностью $\varepsilon = 0.005$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' + y = x^2, & 0 < x < 1, \\ y(0) + y'(0) = -1, & y'(1) = 0. \end{cases}$$

Вариант 6. Методом прогонки найти с точностью $\varepsilon = 0.001$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' = x - 1, & 0 < x < 1, \\ y(0) = 0, & y'(1) = 1. \end{cases}$$

Вариант 7. Методом прогонки найти с точностью $\varepsilon = 0.005$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} -y'' + y' = 2x, & 0 < x < 1, \\ y(0) = 0, & y'(1) = 0. \end{cases}$$

Вариант 8. Методом прогонки найти с точностью $\varepsilon = 0.005$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' - y' = 0, & 1 < x < 2, \\ 2y(1) = 1, & 4y(2) = 1. \end{cases}$$

Вариант 9. Методом прогонки найти с точностью $\varepsilon = 0.05$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' - 3y' + 2y = \sin x, & 0 < x < \pi, \\ y(0) + 0.1 = 0, & -y(\pi) + y'(\pi) - 0.1 = 0. \end{cases}$$

Вариант 10. Методом прогонки найти с точностью $\varepsilon = 0.05$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' - y' = 2e^x - x, & 0 < x < 2, \\ y(0) + y'(0) = -2, & -y(2) = e. \end{cases}$$

Вариант 11. Методом прогонки найти с точностью $\varepsilon = 0.05$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y'' + 4y = 2 \operatorname{tg} x, & 0 < x < \frac{\pi}{4}, \\ y(0) = 0, & y\left(\frac{\pi}{4}\right) = \ln\left(\frac{\sqrt{2}}{2}\right). \end{cases}$$

Вариант 12. Методом прогонки найти с точностью $\varepsilon = 0.05$ решение краевой задачи для дифференциального уравнения второго порядка

$$\begin{cases} y''(x) + y(x) = 2 \cos x, & 0 < x < \frac{\pi}{2}, \\ y(0) = 0, & y\left(\frac{\pi}{2}\right) - y'\left(\frac{\pi}{2}\right) = \frac{\pi}{2} - 1. \end{cases}$$

1.8 Выбор базисных функций

Как видно из предыдущего, в конечном итоге, все методы сводятся к решению системы линейных уравнений (см. (1.7), (1.12), (1.15), (1.8)). Очевидно, что структура матрицы линейной системы уравнений сильно зависит от выбора базисных функций, и именно эти функции в первую очередь определяют эффективность того или иного метода. Приведенный в п. 1.1.1 пример, показывает, что удачный выбор базисных функций приводит к диагональной матрице для системы линейных уравнений. В случае использования конечно-разностного метода (см. п. 1.6.3) такая матрица является трехдиагональной и система (1.13) может быть решена, например, при помощи метода прогонки. Ниже указаны некоторые варианты выбора базисных функций.

1.8.1 Краевые условия и сохранение свойств исходной задачи

При построении приближенного решения u^h по формуле (1.5) необходимо задавать набор базисных функций $\varphi_1, \varphi_2, \dots, \varphi_{n-1}$, удовлетворяющих условиям (1.6). Если какая-либо дополнительная информация о задаче неизвестна, то обычно в качестве такого набора выбирают полиномиальные или тригонометрические функции. Например, для рассматриваемой задачи это могут быть

$$\varphi_1(x) = x(1-x), \quad \varphi_2(x) = x^2(1-x)^2, \dots$$

или

$$\varphi_k(x) = \sin \pi k x.$$

Вычислительная практика показывает, что при удачном выборе базисных функций бывает достаточно ограничиться их небольшим количеством. В общем же случае, когда число n велико, для определения коэффициентов c_k придется решать систему линейных уравнений (1.7) с большой размерностью $(n-1) \times (n-1)$ и сильно заполненной матрицей A_{ik} , определенной формулами (1.8). При решении такой системы могут возникнуть значительные трудности — большие погрешности при вычислении, увеличение времени расчетов и т. п.

Напротив, при использовании конечно-разностной схемы (1.13) матрица в (1.14) трехдиагональна и для нахождения решения

$$u(x_0), u(x_1), \dots, u(x_n)$$

существуют эффективные алгоритмы. Более того, значения функции $u(x)$ непосредственно определяются в точках x_k сразу же после решения системы (1.13), тогда как для нахождения приближенного решения по формуле (1.5) следует сначала, решая (1.7), найти c_k , и лишь затем использовать формулу (1.5).

На самом деле такие преимущества конечно-разностный метод имеет лишь для сравнительно простых уравнений, краевых условий и областей, в которых решается задача. Уже в двумерном случае использование конечно-разностного метода приводит к серьезным проблемам при конструировании краевых условий в областях сложной формы (например, отличных от прямоугольника). Кроме того, часто возникают проблемы сохранения различных свойств исходных задач.

Поясним это на следующем примере. Рассмотрим связанный с задачей (1.1), (1.2) дифференциальный оператор L (строго говоря, в определении следует еще включать требования гладкости)

$$(Lu)(x) \stackrel{def}{=} -\frac{d}{dx} \left(p(x) \frac{du(x)}{dx} \right) + q(x)u(x), \quad 0 < x < 1, \quad (1.17)$$

$$u(0) = 0, \quad u(1) = 0.$$

Дифференциальный оператор называется **симметричным**, если для любых $u(x), v(x)$ из допустимого класса функций, в частности, удовлетворяющих краевым условиям (1.2) и требуемым условиям гладкости, выполнены соотношения

$$(Lu, v) = (u, Lv), \quad (1.18)$$

$$(Lu, v) \stackrel{def}{=} \int_0^1 \left\{ -\frac{d}{dx} \left(p(x) \frac{du(x)}{dx} \right) + q(x)u(x) \right\} v(x) dx. \quad (1.19)$$

Используем ранее введенное понятие билинейной формы (см. (1.2))

$$A(u, v) \stackrel{def}{=} \int_0^1 \left(p \frac{du}{dx} \frac{dv}{dx} + quv \right) dx. \quad (1.20)$$

Очевидно, что условие симметричности (1.18) для задачи (1.1), (1.2) выполнены

$$(Lu, v) = A(u, v), \quad (u, Lv) = A(v, u), \quad A(u, v) = A(v, u).$$

Аналогом свойства симметричности в конечномерном случае, естественно, является симметричность матрицы системы линейных уравнений (1.7). С учетом введенного обозначения (1.20) легко убедиться, что матрица (1.8) симметрична

$$A_{ik} = A(\varphi_i, \varphi_k), \quad A_{ki} = A(\varphi_k, \varphi_i), \quad A_{ik} = A_{ki}.$$

Таким образом, при построении приближенного решения такое важное свойство исходной задачи, как симметричность, сохраняется. Особенно подчеркнем, что матрица A_{ik} будет симметрична при любом выборе базисных функций.

По иному обстоит дело в случае конечно-разностного метода. Опуская довольно утомительные выкладки, приведем систему линейных уравнений для приближенного решения задачи (1.12) в случае, когда при аппроксимации не делается предположение об одинаковой длине отрезков $[x_k, x_{k-1}]$, $k = 1, \dots, n-1$ (ср. с (1.13))

$$-\alpha_k u_{k-1} + 2u_k - \beta_k u_{k+1} = (x_k - x_{k-1})(x_{k+1} - x_k)f(x_k),$$

$$u_0 = 0, \quad u_n = 0, \quad \alpha_k = \frac{2(x_k - x_{k-1})}{(x_{k+1} - x_{k-1})}, \quad \beta_k = \frac{2(x_{k+1} - x_k)}{(x_{k+1} - x_{k-1})}.$$

Теперь соответствующая матрица коэффициентов не является симметричной (ср. с (1.14)), хотя, по-прежнему, остается трехдиагональной

$$\begin{pmatrix} 2 & -\beta_1 & 0 & 0 & \dots & 0 & 0 & 0 \\ -\alpha_2 & 2 & -\beta_2 & 0 & \dots & 0 & 0 & 0 \\ 0 & -\alpha_3 & 2 & -\beta_3 & \dots & 0 & 0 & 0 \\ 0 & 0 & -\alpha_4 & 2 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & -\alpha_{n-2} & 2 & -\beta_{n-2} \\ 0 & 0 & 0 & 0 & \dots & 0 & -\alpha_{n-1} & 2 \end{pmatrix}.$$

Приведенный пример показывает, что проекционный метод построения слабого решения является предпочтительнее конечно-разностного метода, т. к. он позволяет автоматически сохранять такое важное свойство задачи, как симметричность.

1.8.2 Финитные базисные функции

Существует эффективный способ устранения «главного недостатка» проекционных методов — сильной заполненности матрицы системы линейных уравнений. Специальный выбор базисных функций позволяет сделать матрицу A_{ik} сильно разреженной, а во многих случаях трехдиагональной, блочно-диагональной или ленточной. Более того, это можно сделать не только для одномерных задач, но и в случаях многомерных задач в области с достаточно сложной геометрической формой.

Иными словами, возможно сохранить все преимущества проекционных методов и конечно-разностных методов (разреженность матриц).

Напомним некоторые определения.

Определение 1.1. Носителем функции называется замкнутая область, вне которой функция тождественно обращается в нуль. Для носителя функции $\varphi(x)$ используется обозначение: $\text{supp } \varphi$.

Определение 1.2. Функция называется **финитной**, если ее носитель является ограниченной областью (более точно, компактной).

В методе конечных элементов в качестве базисных функций предлагается выбирать финитные функции с размерами носителей, меньшими (как правило, существенно меньшими), чем размеры области, в которой рассматривается решаемая задача.

Для решения задачи (1.4) возможен следующий способ выбора базисных функций. Разобьем отрезок $[0, 1]$ на интервалы $[x_{k-1}, x_k]$, $k = 1, \dots, n$, $x_0 = 0$, $x_n = 1$ (длины интервалов не предполагаются одинаковыми, см. рис. 1).

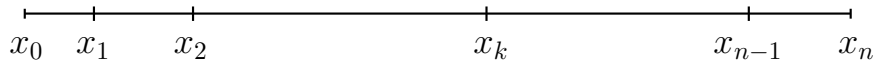
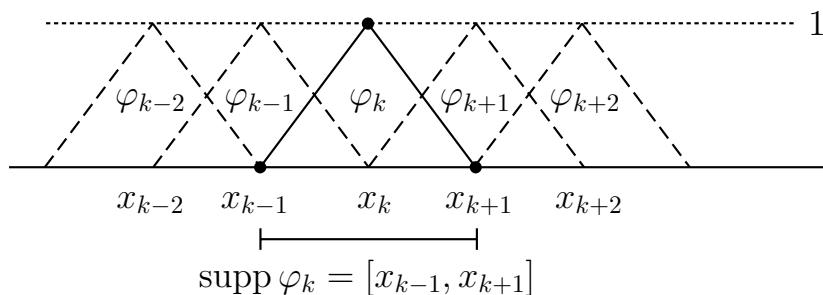


Рис. 1. Разбиение отрезка

В качестве $\varphi_k(x)$, $k = 1, \dots, n-1$ выберем **финитные кусочно-линейные функции**, носителем которых будет отрезок $[x_{k-1}, x_{k+1}]$ (см. рис. 2)

$$\varphi_k(x) = \begin{cases} 0, & x < x_{k-1}, \\ \frac{x - x_{k-1}}{x_k - x_{k-1}}, & x_{k-1} < x < x_k, \\ \frac{x - x_{k+1}}{x_k - x_{k+1}}, & x_k < x < x_{k+1}, \\ 0, & x > x_{k+1}. \end{cases} \quad (1.21)$$

Рис. 2. Набор базисных функций $\varphi_k(x)$

Обычно носители базисных функций называются **конечными элементами**. Часто тот же самый термин используется и для самих базисных функций.

Подчеркнем, что все функции $\varphi_k(x)$ $k = 1, \dots, n-1$ удовлетворяют условиям (1.6).

Сделаем ряд важных замечаний.

1. Носители базисных функций полностью заполняют отрезок $[0, 1]$, т. е.

$$\bigcup_{k=1}^{n-1} \text{supp } \varphi_k = [0, 1]. \quad (1.22)$$

Иными словами, конечные элементы полностью покрывают область, в которой разыскивается решение задачи. В противном случае, на-

шлась бы такая точка отрезка $[0, 1]$, в которой приближенное решение $u^h(x)$, задаваемое формулой (1.5), было бы неопределено.

2. Пересечения носителей различных базисных функций не являются пустыми, но таких пересечений достаточно мало. Более точно

$$\text{supp } \varphi_k \cap \text{supp } \varphi_i \neq \emptyset, \quad i = k - 1, k, k + 1, \quad (1.23)$$

$$\text{supp } \varphi_k \cap \text{supp } \varphi_i \neq \emptyset, \quad i = k - 2, k + 2, \quad (1.24)$$

$$\text{mes}(\text{supp } \varphi_k \cap \text{supp } \varphi_i) = 0,$$

$$\text{supp } \varphi_k \cap \text{supp } \varphi_i = \emptyset, \quad i < k - 2, \quad i > k + 2. \quad (1.25)$$

Соотношения (1.23)–(1.25) являются наиболее важными для эффективности метода конечных элементов. В конечном итоге, именно они определяют структуру матрицы A_{ik} , задаваемой соотношениями (1.8). Благодаря (1.24), (1.25), большинство интегралов в (1.8) обращается в нуль и матрица A_{ik} становится сильно разреженной.

3. Справедливо соотношение

$$u^h(x_s) = \sum_{k=1}^{n-1} c_k \varphi_k(x_s) = c_s. \quad (1.26)$$

Таким образом, коэффициенты c_s — это значения приближенного решения в точках x_s . В частности, это означает, что при решении системы (1.7) приближенное решение $u^h(x)$ **в узлах сетки** будет получено сразу, без использования формулы (1.5). Напомним, что в случае конечно-разностного метода, приближенное решение в узлах сетки также получалось непосредственно после решения системы линейных уравнений (см. (1.13)).

4. Базисные функции $\varphi_k(x)$ выбраны кусочно-линейными из следующих соображений. Во-первых, кусочно-линейные функции допустимо использовать в задаче (1.4), т. к. производные $\varphi'_k(x)$ в этом случае кусочно-постоянны и интегрирование в (1.4) и последующих соотношениях (1.6)–(1.8) может быть выполнено. Во-вторых, кусочно-линейные функции являются наиболее простейшими финитными кусочно-полиномиальными функциями, использование которых позволяет учесть все члены в интегралах (1.8). В случае задачи (1.17)

нельзя выбирать в качестве $\varphi_k(x)$ кусочно-постоянные функции, т. к. в этом случае $\varphi'_k(x)$ обращаются в нуль и члены вида $\varphi'_k(x)\varphi'_i(x)$ будут отсутствовать в интегралах (1.8), что приведет к исчезновению части информации об исходной задаче. По этой же причине, нельзя использовать кусочно-линейные функции непосредственно в выражениях для вычисления матрицы \tilde{A}_{ik} , т. к. интегралы (1.11) содержат вторые производные базисных функций.

1.8.3 Вычисление элементов матрицы A_{ik} и вектора b_i

Уже на основании свойств носителей базисных функций (конечных элементов) (1.23)–(1.25) можно сделать вывод о том, что матрица A_{ik} будет трехдиагональной. Однако, с учетом явных выражений (1.21) для базисных функций полезно привести конкретные формулы для элементов матрицы A_{ik} и вектора b_i системы (1.7).

С учетом (1.25) ясно, что носитель функции φ_k не пересекается с носителями функций φ_i при $i > k + 2$ и $i < k - 2$ (см. также рис. 2). Это означает, что произведение $\varphi_i\varphi_k = 0$ при $i > k + 2$ и $i < k - 2$. Это же относится и к производным функции φ_i , заданной формулой (1.21). На самом деле, из (1.21) следует даже большее — $\varphi_i\varphi_k = 0$ при $i \geq k + 2$ и $i \leq k - 2$.

Учитывая сказанное, запишем формулы для вычисления интегралов, входящих в (1.8). Заметим, что вместо интегрирования по всей области $[0, 1]$ можно ограничиться интегрированием лишь по области, являющейся объединением носителей подинтегральных функций.

Имеем следующие соотношения (пересечение носителей функций либо пусто, либо имеет меру нуль)

$$\int_0^1 p\varphi'_k\varphi'_i dx = 0, \quad i \geq k + 2, \quad i \leq k - 2. \quad (1.27)$$

Далее, заменяем интегрирование по всей области интегрированием

по пересечению носителей (см. рис. 2)

$$\int_0^1 p\varphi'_k\varphi'_k dx = \int_{x_{k-1}}^{x_{k+1}} p\varphi'_k\varphi'_k dx, \quad \int_0^1 p\varphi'_k\varphi'_{k+1} dx = \int_{x_k}^{x_{k+1}} p\varphi'_k\varphi'_{k+1} dx,$$

$$\int_0^1 p\varphi'_k\varphi'_{k-1} dx = \int_{x_{k-1}}^{x_k} p\varphi'_k\varphi'_{k-1} dx. \quad (1.28)$$

Аналогично для других членов, входящих в формулу (1.8)

$$\int_0^1 q\varphi_k\varphi_i dx = 0, \quad k-2 \leq i \leq k+2, \quad (1.29)$$

$$\int_0^1 q\varphi_k\varphi_k dx = \int_{x_{k-1}}^{x_{k+1}} q\varphi_k\varphi_k dx, \quad \int_0^1 q\varphi_k\varphi_{k+1} dx = \int_{x_k}^{x_{k+1}} q\varphi_k\varphi_{k+1} dx,$$

$$\int_0^1 q\varphi_k\varphi_{k-1} dx = \int_{x_{k-1}}^{x_k} q\varphi_k\varphi_{k-1} dx, \quad \int_0^1 f\varphi_i dx = \int_{x_{i-1}}^{x_{i+1}} f\varphi_i dx. \quad (1.30)$$

Таким образом, матрица A_{ik} является трехдиагональной матрицей.

Пример. В простых случаях интегралы (1.28), (1.30) легко вычисляются. Рассмотрим краевую задачу (см. (1.12))

$$-u''(x) = 1, \quad u(0) = 0, \quad u(1) = 0.$$

Сравнивая с (1.1), запишем

$$p = 1, \quad q = 0, \quad f(x) = 1.$$

Производя вычисление интегралов, получим следующие результаты (в окончательном ответе, для простоты, полагаем $x_k = kh$)

$$\int_{x_{k-1}}^{x_{k+1}} \varphi'_k\varphi'_k dx = \int_{x_{k-1}}^{x_k} \varphi'_k\varphi'_k dx + \int_{x_k}^{x_{k+1}} \varphi'_k\varphi'_k dx =$$

$$= \frac{1}{x_k - x_{k-1}} + \frac{1}{x_{k+1} - x_k} = \frac{2}{h}.$$

$$\int_{x_k}^{x_{k+1}} \varphi'_k \varphi'_{k+1} dx = \frac{1}{x_k - x_{k+1}} = -\frac{1}{h}.$$

$$\int_{x_{k-1}}^{x_k} \varphi'_k \varphi'_{k-1} dx = \frac{1}{x_{k-1} - x_k} = -\frac{1}{h}.$$

$$\int_0^1 \varphi_i dx = \int_{x_{i-1}}^{x_{i+1}} \varphi_i dx = \int_{x_{i-1}}^{x_i} \varphi_i dx + \int_{x_i}^{x_{i+1}} \varphi_i dx.$$

$$\int_{x_{i-1}}^{x_i} \varphi_i dx = \int_{x_{i-1}}^{x_i} \frac{x - x_{i-1}}{x_i - x_{i-1}} dx = \frac{x_i - x_{i-1}}{2} = \frac{h}{2}.$$

$$\int_{x_i}^{x_{i+1}} \varphi_i dx = \int_{x_i}^{x_{i+1}} \frac{x - x_{i+1}}{x_i - x_{i+1}} dx = \frac{x_{i+1} - x_i}{2} = \frac{h}{2}.$$

Таким образом,

$$A_{ii-1} = -1/h, \quad A_{ii} = 2/h, \quad A_{ii+1} = -1/h, \quad b_i = h.$$

Используя полученные соотношения, запишем систему (1.7)

$$A_{ii-1}c_{i-1} + A_{ii}c_i + A_{ii+1}c_{i+1} = b_i$$

или

$$-\frac{c_{i-1} - 2c_i + c_{i+1}}{h^2} = 1.$$

С учетом (1.26) запишем

$$-\frac{u^h(x_{i-1}) - 2u^h(x_i) + u^h(x_{i+1}))}{h^2} = 1.$$

Сравнивая полученное выражение с (1.13), видим, что вновь получена обычная конечно-разностная схема (при $f(x) = 1$).

Конечно, такие простые соотношения получаются лишь для простых примеров. В более общем случае для вычисления интегралов (1.28), (1.30) следует использовать те или иные квадратурные формулы. От правильного выбора квадратурной формулы будет зависеть точность вычисления интеграла и, в конечном итоге, точность представления системы линейных уравнений.

1.9 Естественные и главные краевые условия

При использовании метода конечных элементов важное значение имеет правильный учет краевых условий рассматриваемой краевой задачи. В случае задачи (1.1), (1.2) при построении приближенного решения в виде (1.5) выполнение краевых условий (1.2) обеспечивалось выбором базисных функций. Действительно, для базисных функций требовалось выполнение краевых условий (1.6), которые и были соблюдены при задании φ_k соотношениями (1.21).

В общем случае для задач с краевыми условиями, отличными от (1.2), не всегда удастся удовлетворить краевым условиям, выбирая подходящие базисные функции. Более точно, такой выбор может быть сопряжен с большими техническими трудностями — функции могут оказаться слишком сложными, матрица системы линейных уравнений (1.7) не будет сильно разреженной и пр.

Рассмотрим следующую краевую задачу, которая отличается от задачи (1.1), (1.2) (при $p = 1, q = 0$) лишь заданием краевого условия третьего рода при $x = 0$

$$-u''(x) = f(x), \quad 0 < x < 1, \quad (1.31)$$

$$u'(0) - \alpha u(0) = 0, \quad u(1) = 0, \quad (1.32)$$

где $f(x)$ — известная функция, α — заданное число.

Следуя процедуре, описанной в п. 1.8.1, умножим (1.31) на $v(x)$ и проинтегрируем от 0 до 1 с использованием интегрирования по частям

$$\int_0^1 \frac{du}{dx} \frac{dv}{dx} dx - \frac{du}{dx} v \Big|_0^1 = \int_0^1 f v dx. \quad (1.33)$$

Потребуем, чтобы $v(x)$ при $x = 1$ удовлетворяла тому же краевому условию, что и $u(x)$, т. е.

$$v(1) = 0. \quad (1.34)$$

Заметим, что нельзя требовать выполнения условия $v(0) = 0$, т. к. в этом случае (1.33) примет вид (1.4) (при $p = 1, q = 0$), что, очевидно, будет соответствовать решению задачи (1.31), (1.32) с краевым условием $u(0) = 0$ вместо краевого условия $u'(0) - \alpha u(0) = 0$.

Соотношение (1.33) с учетом (1.34) примет вид

$$\int_0^1 \left(\frac{du}{dx} \frac{dv}{dx} - fv \right) dx + u'(0)v(0) = 0, \quad v(1) = 0.$$

Используя краевое условие (1.32) при $x = 0$, получим (ср. с (1.4))

$$\int_0^1 \left(\frac{du}{dx} \frac{dv}{dx} - fv \right) dx + \alpha u(0)v(0) = 0, \quad v(1) = 0. \quad (1.35)$$

Функцию $u(x)$, являющуюся решением задачи (1.35), будем называть **слабым решением** задачи (1.31), (1.32).

Приближенное решение задачи (1.35) строим в виде, аналогичном (1.5), **добавив к базисным функциям еще одну дополнительную функцию** $\varphi_0(x)$, линейно не зависящую от прежних базисных функций

$$u^h(x) = \sum_{k=0}^{n-1} c_k \varphi_k(x). \quad (1.36)$$

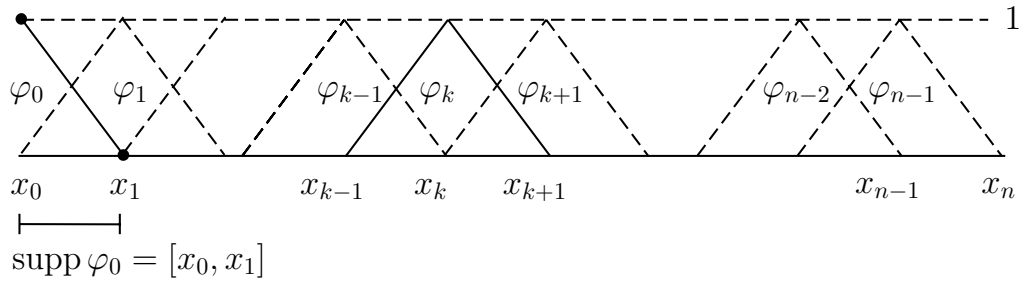
Необходимость введения дополнительной базисной функции достаточно очевидна. Сохранив прежний набор базисных функций, невозможно удовлетворить краевым условиям при $x = 0$, т. к. в силу (1.6) $\varphi_k(0) = 0$, $\varphi_k(1) = 0$, $k = 1, \dots, n-1$. Понятно также, что, выбирая $\varphi_0(x)$, следует потребовать выполнения условия $\varphi_0(1) = 0$. В противном случае нарушится выполнение краевого условия при $x = 1$ для приближенного решения, т. е. будет $u^h(1) \neq 0$. (Конечно, кроме тривиального случая, когда $c_0 = 0$, что соответствует просто прежнему набору базисных функций.)

Зададим дополнительную базисную функцию $\varphi_0(x)$ в виде **финитной кусочно-линейной функции** с носителем

$$\text{supp } \varphi_0 = [x_0, x_1]$$

(ср. с (1.21) и см. рис. 3 и 2)

$$\varphi_0(x) = \begin{cases} 1, & x = x_0, \\ \frac{x - x_1}{x_0 - x_1}, & x_0 < x < x_1, \\ 0, & x > x_1. \end{cases} \quad (1.37)$$

Рис. 3. Набор базисных функций $\varphi_k(x)$

Выбор $\varphi_0(x)$ в форме (1.37) продиктован следующими соображениями. Новая базисная функция не должна «сильно отличаться» от прежних, по крайней мере, принадлежать тому же классу функций (в данном случае, быть кусочно-линейной). Желательно сохранение условия (1.26), т. е. $u^h(x_0) = c_0$, что приводит к условию $\varphi_0(0) = 1$. Желательно также сохранение условий, аналогичных (1.23)–(1.25):

$$\text{supp } \varphi_0 \cap \text{supp } \varphi_1 \neq \emptyset, \quad \text{supp } \varphi_0 \cap \text{supp } \varphi_i = \emptyset, \quad i > 2, \quad (1.38)$$

$$\text{supp } \varphi_0 \cap \text{supp } \varphi_2 \neq \emptyset,$$

$$\text{mes}(\text{supp } \varphi_0 \cap \text{supp } \varphi_2) = 0.$$

Особенно подчеркнем, что функция $\varphi_0(x)$ не удовлетворяет краевому условию (1.32) при $x = 0$. Конечно можно было бы выбрать $\varphi_0(x)$ в виде $\varphi_0(x) = 1 + \alpha x$, например, на отрезке $[x_0, x_1]$, и $\varphi_0(x) = 0$ — вне этого отрезка. Это привело бы к $\varphi_0'(0) - \alpha \varphi_0(0) \equiv 0$. Однако, при этом теряется некоторая «универсальность» базисных функций — φ_0 будет зависеть от параметра задачи α . Более того, детальный анализ показывает, что такой выбор существенно ухудшает процесс построения решения.

После того как выбор дополнительной базисной функции осуществлен, схема построения системы линейных уравнений для определения c_k будет такая же, как в п. 1.6.1.

Подставляя (1.36) в (1.35), получим ($k = 0, 1, \dots, n-1$)

$$\int_0^1 \left(\sum_{k=0}^{n-1} c_k \varphi_k' v' - f v \right) dx + \alpha \sum_{k=0}^{n-1} c_k \varphi_k(0) v(0) = 0.$$

Выбирая в качестве $v(x)$ набор функций $\varphi_i(x)$, $i = 0, \dots, n-1$, имеем

$$\int_0^1 \left(\sum_{k=0}^{n-1} c_k \varphi'_k \varphi'_i - f \varphi_k \right) dx + \alpha \sum_{k=0}^{n-1} c_k \varphi_k(0) \varphi_i(0) = 0, \quad i = 0, \dots, n-1.$$

Для определения c_k имеем систему линейных уравнений

$$\sum_{k=0}^{n-1} B_{ik} c_k = b_i, \quad i = 0, \dots, n-1, \quad (1.39)$$

где введены следующие обозначения

$$\begin{aligned} \int_0^1 \varphi'_i \varphi'_k dx + \alpha \varphi_i(0) \varphi_k(0) &= B_{ik}, \\ \int_0^1 f \varphi_i dx &= b_i, \quad i, k = 0, \dots, n-1. \end{aligned} \quad (1.40)$$

Нетрудно показать, что элементы матрицы B_{ik} для $i, k = 1, \dots, n-1$ совпадают с элементами матрицы A_{ik} (при $p = 1$, $q = 0$), определяемыми формулами (1.8). Действительно, базисные функции $\varphi_1, \dots, \varphi_{n-1}$ остались прежними и $\varphi_k(0) = 0$, $k = 1, \dots, n-1$.

В силу (1.38) при вычислении интегралов в (1.40) получим (пересечение носителей соответствующих функций имеет нулевую меру)

$$B_{0k} = 0, \quad B_{k0} = 0, \quad k = 2, \dots, n-1.$$

Используя вид функций φ_0 и φ_1 (см. (1.21) и (1.37)) в случае, когда длины всех отрезков $[x_k, x_{k+1}]$ одинаковы и равны h , получим

$$B_{00} = \frac{1 + \alpha h}{h}, \quad B_{10} = B_{01} = -\frac{1}{h}.$$

Таким образом, матрица B_{ik} размерности $n \times n$ отличается от матрицы A_{ik} размерности $(n-1) \times (n-1)$ лишь левым столбцом и верхней строкой. Система уравнений (1.39), для удобства умноженная на h , в матричной форме имеет вид (выделена часть, соответствующая

матрице A_{ik})

$$hBc = \begin{pmatrix} 1 + \alpha h & -1 & 0 & 0 & \dots & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & \dots & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & \dots & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & \dots & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ \dots \\ c_{n-2} \\ c_{n-1} \end{pmatrix} = \begin{pmatrix} hb_0 \\ hb_1 \\ hb_2 \\ hb_3 \\ \dots \\ hb_{n-2} \\ hb_{n-1} \end{pmatrix}.$$

Интересно посмотреть, чему соответствует первое уравнение системы (1.39). При $i = 0$, с учетом

$$c_0 = u(x_0) = u(0), \quad c_1 = u(x_1) = u(x_0 + h) = u(h),$$

имеем

$$(1 + \alpha h)u(0) - u(h) = h \int_0^h f(x) \left(\frac{h-x}{h} \right) dx = \int_0^h f(x)(h-x) dx.$$

Используя разложение в ряд в окрестности точки $h = 0$, выводим

$$(\alpha u(0) - u'(0))h - \frac{1}{2}u''(0)h^2 + \mathcal{O}(h^3) = \frac{1}{2}f(0)h^2 + \mathcal{O}(h^3).$$

Если предполагать выполнение уравнения (1.31) при $x = 0$, то имеем

$$u''(0) = -f(0).$$

Это означает, что краевое условие (1.32) при $x = 0$ выполняется с точностью до членов порядка $\mathcal{O}(h^2)$

$$u'(0) - \alpha u(0) = \mathcal{O}(h^2), \quad h \rightarrow 0.$$

Этот факт служит косвенным подтверждением правильности выбора базисной функции $\varphi_0(x)$.

Рассмотренные примеры показывают, что имеются различные возможности удовлетворения краевым условиям. В первом случае все краевые условия для задачи (1.1), (1.2) были выполнены за счет выбора базисных функций (см. формулу (1.21)). Во втором случае, для

задачи (1.31), (1.32) краевое условие (1.32) при $x = 1$ по-прежнему выполнено за счет выбора базисных функций. Краевое условие (1.32) при $x = 0$ учитывается «естественным образом». Имеется в виду следующее: в результате преобразования уравнений в (1.35) возникает дополнительный член $\alpha u(0)v(0)$. Затем к базисным функциям добавляется новая функция φ_0 (не удовлетворяющая краевому условию!) и задача приводится к решению системы линейных уравнений (1.39) с матрицей B_{ik} , включающей дополнительные, по сравнению с матрицей A_{ik} , члены $\alpha \varphi_i(0)\varphi_k(0)$.

Определение 1.3. Краевые условия, которым можно удовлетворить за счет выбора базисных функций, называются **главными краевыми условиями**.

Определение 1.4. Краевые условия, удовлетворение которых возможно за счет преобразования задачи, а не за счет выбора базисных функций, называются **естественными краевыми условиями**.

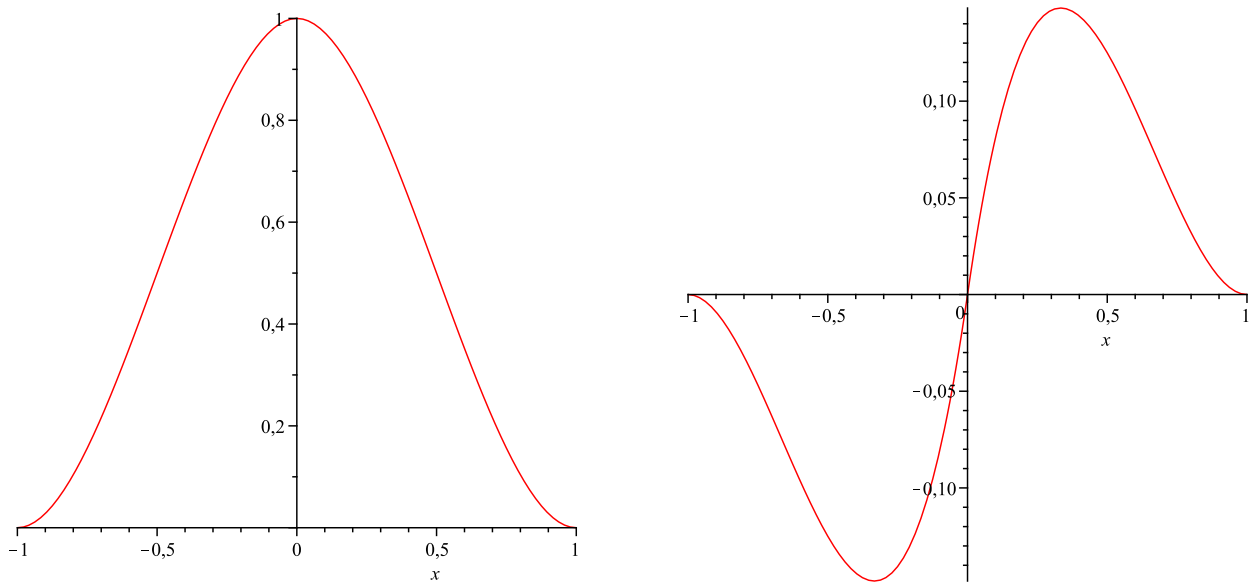
Таким образом, в задаче (1.1), (1.2) оба краевых условия (1.2) являются **главными**. В задаче (1.31), (1.32) краевое условие (1.32) при $x = 1$ будет **главным**, а краевое условие (1.32) при $x = 0$ является **естественным**.

Понятие о главных и естественных краевых условиях является весьма важным для метода конечных элементов. В случае естественных краевых условий, в некотором смысле, можно не заботиться о выборе базисных функций. В случае же главных краевых условий приходится ставить дополнительные ограничения на выбор базисных функций. Заметим, что большое количество примеров задач с естественными и главными краевыми условиями имеется в [5], где подробно разъясняются всевозможные математические проблемы, возникающие в связи с численной реализацией различных граничных условий.

1.10 Кубические конечные элементы

Помимо функции Куранта можно выбирать другие базисные функции $\varphi_1(x)$, $\varphi_2(x)$ с конечным носителем, например, показанные на

рис. 4.

Рис. 4. Функции $\varphi_1(x)$, $\varphi_2(x)$

Опишем способ построения таких кусочно-полиномиальных функций. Выберем функции $\Phi_1(x)$, $\Phi_2(x)$ в виде кубических полиномов, удовлетворяющих на отрезке $[0, 1]$ условиям (см., например, [4, 5, 15])

$$\Phi_1(0) = 1, \quad \Phi_1(1) = 0, \quad \Phi_1'(0) = 0, \quad \Phi_1'(1) = 0, \quad (1.41)$$

$$\Phi_2(0) = 0, \quad \Phi_2(1) = 0, \quad \Phi_2'(0) = 1, \quad \Phi_2'(1) = 0. \quad (1.42)$$

На отрезке $[-1, 0]$ эти функции будут соответственно $\Phi_1(-x)$ и $-\Phi_2(-x)$. Вне отрезка, т. е. при $|x| > 1$ предполагается, что функция обращается в нуль, т. е. функции $\varphi_1(x)$, $\varphi_2(x)$ имеют конечный носитель $[-1, 1]$. Легко записать такие функции (см., например, [4, 5, 15])

$$\varphi_1(x) = \begin{cases} \Phi_1(-x) = (x+1)^2(1-2x), & -1 \leq x \leq 0, \\ \Phi_1(x) = (x-1)^2(1+2x), & 0 \leq x \leq 1, \\ 0, & |x| > 1, \end{cases} \quad (1.43)$$

$$\varphi_2(x) = \begin{cases} -\Phi_2(-x) = x(x+1)^2, & -1 \leq x \leq 0, \\ \Phi_2(x) = x(x-1)^2, & 0 \leq x \leq 1, \\ 0, & |x| > 1. \end{cases} \quad (1.44)$$

Решение задачи, ограничиваясь случаем равномерного разбиения отрезка точками x_i , разыскиваем в виде

$$u^h(x) = \sum_i \left\{ a_i \varphi_1 \left(\frac{x - x_i}{h} \right) + h b_i \varphi_2 \left(\frac{x - x_i}{h} \right) \right\}. \quad (1.45)$$

Заметим, что

$$u^h(x_i) = a_i, \quad \frac{du^h(x_i)}{dx} = b_i. \quad (1.46)$$

Таким образом, как и в случае функции Куранта коэффициенты a_i, b_i совпадают со значениями решения и его производной в точках x_i .

1.10.1 Пример

Вновь рассмотрим задачу (1.12)

$$-u''(x) = f(x), \quad 0 < x < 1, \quad u(0) = 0, \quad u(1) = 0. \quad (1.47)$$

Введем обозначения

$$\varphi_{i1}(x) = \varphi_1 \left(\frac{x - x_i}{h} \right), \quad \varphi_{i2}(x) = \varphi_2 \left(\frac{x - x_i}{h} \right). \quad (1.48)$$

Разыскиваем слабое решение задачи (1.4)

$$\int_0^1 \left(p \frac{du}{dx} \frac{dv}{dx} + quv - fv \right) dx = 0, \quad v(0) = 0, \quad v(1) = 0 \quad (1.49)$$

в виде (1.45), который переписан с учетом соотношений (1.48)

$$u^h(x) = \sum_{i=1}^{n-1} \{ a_i \varphi_{i1}(x) + h b_i \varphi_{i2}(x) \}. \quad (1.50)$$

В качестве функции $v(x)$ выбираем функции

$$v(x) = \varphi_{i1}(x), \quad i = 1, \dots, n-1 \quad (1.51)$$

и

$$v(x) = \varphi_{i2}(x), \quad i = 1, \dots, n-1. \quad (1.52)$$

Очевидно, что подстановка (1.50) в (1.49) с учетом (1.51) и (1.52) после вычисления интегралов приведет к системе линейных уравнений относительно коэффициентов a_i, b_i .

Упражнение.

1) Показать непосредственным вычислением, что

$$\int_{x_{i-1}}^{x_{i+1}} \varphi_{i1}(x)\varphi_{i2}(x) dx = 0, \quad \int_{x_{i-1}}^{x_{i+1}} \varphi'_{i1}(x)\varphi'_{i2}(x) dx = 0. \quad (1.53)$$

(Этот же результат можно получить не вычисляя интеграл. Действительно, функция $\varphi_{i1}(x)$ является четной, а функция $\varphi_{i2}(x)$ является нечетной относительно точки x_i . Произведение $\varphi_{i1}(x)\varphi_{i2}(x)$ — нечетная функция. Интеграл от нечетной функции по симметричному отрезку очевидно равен нулю. Аналогично для случая производных.)

2) Показать, что матрица системы линейных уравнений является пятидиагональной. Точнее блочной двухдиагональной матрицей с размерами блоков 2×2 . Более того, блоки не на главной диагонали имеют вид треугольных матриц.

2 Метод конечных элементов в двумерном случае

С точки зрения подхода к построению решения метод конечных элементов для пространственно многомерных задач ничем не отличается от одномерного случая. Также, как и в одномерном случае, используется проекционный метод и получается аналог, например, задачи (1.4). Конструируется набор финитных базисных функций и приближенное решение разыскивается в виде их линейной комбинации аналогично (1.5). Затем получается система линейных уравнений вида (1.7) с достаточно разреженной матрицей и решение этой системы определяет значения приближенного решения в некоторых точках области.

Серьезные математические и вычислительные трудности возникают реально на этапе реализации метода конечных элементов, например, при триангуляции плоской области, при выборе подходящего для конкретной задачи набора финитных базисных функций, при решении систем линейных уравнений большой размерности (несмотря на сильную разреженность матриц, это достаточно серьезная проблема) и при доказательстве сходимости приближенного решения к точному решению задачи.

2.1 Задача Дирихле для уравнения Лапласа. Слабое решение

Рассмотрим задачу Дирихле для уравнения Лапласа (первую краевую задачу) для определения функции $u(x, y)$ в случае квадратной области:

$$-\Delta u = f(x, y), \quad (x, y) \in D = (0, 1) \times (0, 1), \quad \Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}, \quad (2.1)$$

$$u|_{\Gamma} = 0. \quad (2.2)$$

Здесь $f(x, y)$ — заданная в области D функция, $\Gamma = \partial D$ — граница области D (стороны квадрата), Δ — оператор Лапласа.

Краевые условия выбраны однородными для простоты изложения.

Прежде чем записать задачу (2.1), (2.2) в форме, аналогичной (1.4), напомним некоторые основные формулы и обозначения.

Формула Грина (аналог интегрирования по частям для одномерного случая) в двумерном случае имеет вид

$$\iint_D v \Delta u \, dx \, dy = - \iint_D \nabla u \cdot \nabla v \, dx \, dy + \int_{\Gamma} v \frac{\partial u}{\partial n} \, ds, \quad (2.3)$$

где

$$\nabla v = \left(\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y} \right), \quad \nabla u = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right), \quad \nabla u \cdot \nabla v = \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial v}{\partial y}, \quad (2.4)$$

$$\mathbf{n} = (n_x, n_y), \quad \frac{\partial v}{\partial n} = \mathbf{n} \cdot \nabla v = n_x \frac{\partial v}{\partial x} + n_y \frac{\partial v}{\partial y}. \quad (2.5)$$

Здесь ∇u — градиент функции u (другое обозначение $\text{grad } u$), \mathbf{n} — вектор нормали (внешней к области D) с компонентами n_x, n_y , $\partial v / \partial n$ — производная по нормали.

Для вывода соотношения аналогичного (1.4) умножим (2.1) на функцию $v(x, y)$ и проинтегрируем по области D

$$- \iint_D v \Delta u \, dx \, dy = \iint_D v f \, dx \, dy. \quad (2.6)$$

Используя формулу Грина (2.3), получим

$$\iint_D \nabla u \cdot \nabla v \, dx \, dy - \int_{\Gamma} v \frac{\partial u}{\partial n} \, ds - \iint_D f v \, dx \, dy = 0. \quad (2.7)$$

Потребуем выполнения для функции $v(x, y)$ краевого условия (2.2)

$$v|_{\Gamma} = 0. \quad (2.8)$$

Окончательно, из (2.7) с учетом (2.8) получим аналог (1.4):

$$\forall v, \quad \iint_D (\nabla u \cdot \nabla v - f v) \, dx \, dy = 0, \quad v|_{\Gamma} = 0. \quad (2.9)$$

Соотношение (2.9) должно выполняться при **любом** v . Решение задачи (2.9) называется слабым (или обобщенным) решением исходной задачи (2.1), (2.2).

Напомним, что для строгой постановки задачи необходимо дополнительно указать каким функциональным пространствам принадлежат функции u, v , например, являются ли они непрерывными, дифференцируемыми и т. п. Далее считаем, что все дополнительные условия поставлены (подробнее см., например, [5, 7]).

2.2 Построение приближенного решения

Разобьем область $\bar{D} = [0, 1] \times [0, 1]$ на квадратные ячейки с длиной стороны h и вершинами в узлах

$$x_i = ih, \quad y_k = kh, \quad i, k = 0, \dots, n, \quad nh = 1. \quad (2.10)$$

Каждую квадратную ячейку $(x, y) \in [ih, ih + h] \times [kh, kh + h]$ дополнительно разобьем диагональю, проходящей через вершины (ih, kh) и $(ih + h, kh + h)$ (см. рис. 5). Таким образом, вся область \bar{D} будет разбита на треугольники, которые обозначим T_m . Такое разбиение области называется **триангуляцией** области \bar{D} . В рассматриваемом случае объединение треугольников полностью покрывает область \bar{D}

$$\bigcup_m T_m = \bar{D}. \quad (2.11)$$

Отметим, что пересечение треугольников T между собой возможно только по их границам ∂T (сторонам и вершинам) и мера пересечений треугольников равна нулю

$$\text{mes}(T_j \cap T_m) = 0, \quad j \neq m. \quad (2.12)$$

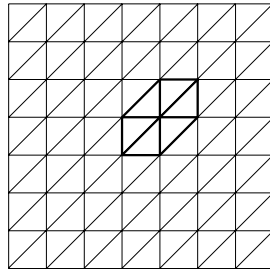


Рис. 5. Пример триангуляции области $\bar{D} = [0, 1] \times [0, 1]$

Определим вспомогательную кусочно-линейную функцию (функция Куранта, [5])

$$\varphi(\xi, \eta) = \begin{cases} 1 - \xi + \eta, & 0 \leq \xi \leq 1, \quad \xi - 1 \leq \eta \leq 0, & (T_1), \\ 1 - \xi, & 0 \leq \xi \leq 1, \quad 0 \leq \eta \leq \xi, & (T_2), \\ 1 - \eta, & 0 \leq \xi \leq 1, \quad \xi \leq \eta \leq 1, & (T_3), \\ 1 + \xi - \eta, & -1 \leq \xi \leq 0, \quad 0 \leq \eta \leq 1 + \xi, & (T_4), \\ 1 + \xi, & -1 \leq \xi \leq 0, \quad \xi \leq \eta \leq 0, & (T_5), \\ 1 + \eta, & -1 \leq \xi \leq 0, \quad -1 \leq \eta \leq \xi, & (T_6). \end{cases} \quad (2.13)$$

Здесь T_s соответствуют треугольникам на рис. 6.

Для каждого внутреннего узла $i, k = 1, \dots, n - 1$ определим финитную кусочно-линейную базисную функцию с носителем в виде шестиугольника S^{ik} , состоящего из треугольников с общей вершиной в точке (x_i, y_k) , которые обозначим $T_1, T_2, T_3, T_4, T_5, T_6$ (см. рис. 6)

$$\varphi_{ik}(x, y) = \varphi\left(\frac{x - x_i}{h}, \frac{y - y_k}{h}\right), \quad \text{supp } \varphi_{ik} = \bigcup_{s=1}^6 T_s = S^{ik}. \quad (2.14)$$

Иногда, для того чтобы подчеркнуть наличие у треугольников общей вершины (x_i, y_k) , будет использовано обозначение T_s^{ik} .

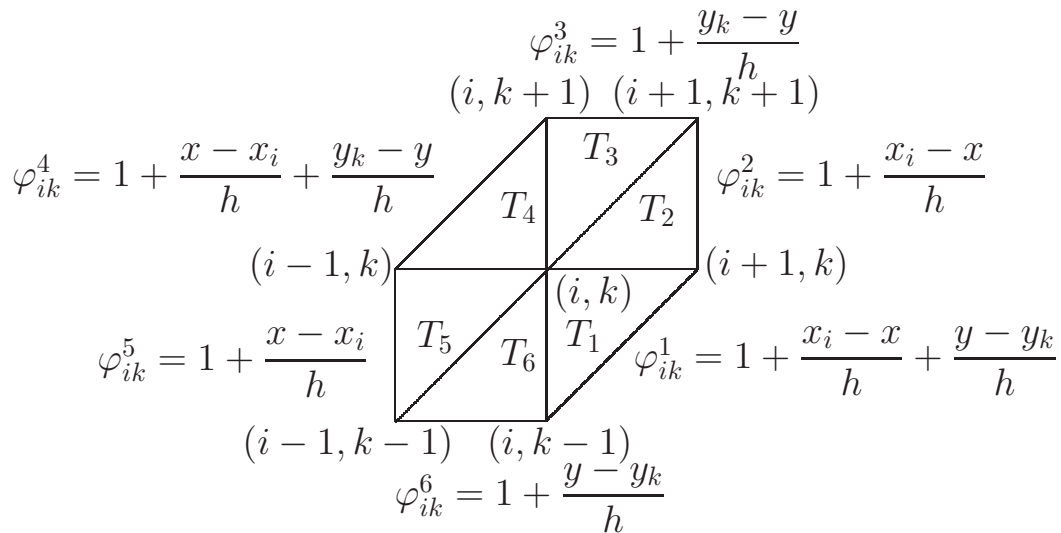


Рис. 6. Носитель функции $\varphi_{ik}(x, y)$ (конечный элемент) и ее значения в T_k

Значения функции $\varphi_{ik}(x, y)$ в каждом треугольнике $T_s, s = 1, \dots, 6$ для удобства приведены на рис. 6. Заметим, что функция $\varphi_{ik}(x, y)$ обращается в нуль во всех узлах за исключением узла (x_i, y_k)

$$\varphi_{ik}(x_i, y_k) = 1. \quad (2.15)$$

Кроме этого, на всех сторонах шестиугольника функция $\varphi_{ik}(x, y)$, очевидно, равна нулю.

Приближенное решение задачи (2.9) ищем в виде линейной комбинации базисных функций по всем внутренним узлам области D

$$u^h(x, y) = \sum_{i,k=1}^{n-1} u_{ik} \varphi_{ik}(x, y), \quad u_{ik} = u^h(x_i, y_k). \quad (2.16)$$

Как уже говорилось, на границах носителя базисные функции обращаются в нуль. Задавая в (2.16) суммирование по всем внутренним узлам области D , удастся удовлетворить краевым условиям (2.2) для $u^h(x, y)$ почти на всей границе области D . Иными словами, краевые условия (2.2), так же как и для задачи (1.1), (1.2), являются главными, т. е. их удастся выполнить за счет выбора базисных функций (см. определение 1.3, с. 42).

Заметим, что объединение носителей всех базисных функций не покрывает полностью область D — значения $u^h(x, y)$ не определены в левом верхнем и правом нижнем треугольниках триангулированной области (см. рис. 2.1). Это означает, что на части границы краевые условия все-таки не выполнены. К счастью, можно показать, что для рассматриваемой задачи это не играет существенной роли. Однако данный факт означает, что во избежание ошибок к выбору базисных функций следует относиться внимательно.

Далее, процедура определения неизвестных величин u_{ik} в формуле (2.16) полностью следует [7] и п. 1.6.1. Подставляя (2.16) в (2.9) и выбирая в качестве функций $v(x, y)$ базисные функции $\varphi_{pq}(x, y)$, $p, q = 1, \dots, n-1$, получим (проекция (2.9) на линейное пространство, задаваемое базисными функциями)

$$\sum_{i,k=1}^{n-1} u_{ik} \iint_D \left(\frac{\partial \varphi_{ik}}{\partial x} \frac{\partial \varphi_{pq}}{\partial x} + \frac{\partial \varphi_{ik}}{\partial y} \frac{\partial \varphi_{pq}}{\partial y} \right) dx dy = \int_D f \varphi_{pq} dx dy. \quad (2.17)$$

Для определения u_{ik} имеем линейные алгебраические уравнения

$$\sum_{i,k=1}^{n-1} A_{pq,ik} u_{ik} = b_{pq}, \quad p, q = 1, \dots, n-1, \quad (2.18)$$

где

$$A_{pq,ik} = \iint_D \left(\frac{\partial \varphi_{ik}}{\partial x} \frac{\partial \varphi_{pq}}{\partial x} + \frac{\partial \varphi_{ik}}{\partial y} \frac{\partial \varphi_{pq}}{\partial y} \right) dx dy, \quad b_{pq} = \int_D f \varphi_{pq} dx dy. \quad (2.19)$$

В рассматриваемом случае элементы матрицы $A_{pq,ik}$ легко определяются. Действительно, т. к. мера пересечения шестиугольника S_{pq} с ше-

стигмольниками S_{ik} равна нулю при $i \geq p + 2, k \geq q + 2$, то

$$A_{pq,ik} = \iint_D (\dots) dx dy = \iint_{S_{ik} \cap S_{pq}} (\dots) dx dy = 0, \quad i \geq p + 2, \quad k \geq q + 2. \quad (2.20)$$

Далее заметим, что для любой базисной функции независимо от индексов i, k производные определяются соотношениями

$$\begin{aligned} \frac{\partial \varphi_{ik}^m}{\partial x} &= 0, \quad m = 3, 6; & \frac{\partial \varphi_{ik}^m}{\partial x} &= -\frac{1}{h}, \quad m = 1, 2; & \frac{\partial \varphi_{ik}^m}{\partial x} &= \frac{1}{h}, \quad m = 4, 5; \\ \frac{\partial \varphi_{ik}^m}{\partial y} &= 0, \quad m = 2, 5; & \frac{\partial \varphi_{ik}^m}{\partial y} &= -\frac{1}{h}, \quad m = 3, 4; & \frac{\partial \varphi_{ik}^m}{\partial y} &= \frac{1}{h}, \quad m = 1, 6. \end{aligned}$$

Тогда, например, для $A_{pq,pq+1}$ с учетом того, что пересечение $\text{supp } \varphi_{pq}$ и $\text{supp } \varphi_{pq+1}$ является лишь пересечениями треугольников $T_4^{pq} \cap T_6^{pq+1}$ и $T_3^{pq} \cap T_1^{pq+1}$, имеем

$$\begin{aligned} A_{pq,pq+1} &= \iint_D \left(\frac{\partial \varphi_{pq+1}}{\partial x} \frac{\partial \varphi_{pq}}{\partial x} + \frac{\partial \varphi_{pq+1}}{\partial y} \frac{\partial \varphi_{pq}}{\partial y} \right) dx dy = \\ &= \iint_{T_4^{pq} \cap T_6^{pq+1}} \left(0 \cdot \frac{1}{h} - \frac{1}{h} \cdot \frac{1}{h} \right) dx dy + \iint_{T_3^{pq} \cap T_1^{pq+1}} \left(-\frac{1}{h} \cdot 0 - \frac{1}{h} \cdot \frac{1}{h} \right) dx dy = -1. \end{aligned}$$

Аналогично для остальных элементов получим

$$\begin{aligned} A_{pq,p-1q} &= \iint_{T_5^{pq} \cap T_1^{p-1q}} (\dots) dx dy + \iint_{T_4^{pq} \cap T_2^{p-1q}} (\dots) dx dy = -1. \\ A_{pq,p-1q-1} &= \iint_{T_5^{pq} \cap T_3^{p-1q-1}} (\dots) dx dy + \iint_{T_6^{pq} \cap T_2^{p-1q-1}} (\dots) dx dy = 0. \\ A_{pq,pq-1} &= \iint_{T_6^{pq} \cap T_4^{pq-1}} (\dots) dx dy + \iint_{T_1^{pq} \cap T_3^{pq-1}} (\dots) dx dy = -1. \\ A_{pq,p+1q} &= \iint_{T_2^{pq} \cap T_4^{p+1q}} (\dots) dx dy + \iint_{T_1^{pq} \cap T_5^{p+1q}} (\dots) dx dy = -1. \\ A_{pq,p+1q+1} &= \iint_{T_3^{pq} \cap T_5^{p+1q+1}} (\dots) dx dy + \iint_{T_2^{pq} \cap T_6^{p+1q+1}} (\dots) dx dy = 0. \\ A_{pq,pq} &= \iint_D (\dots) dx dy = \iint_{T_1^{pq} \cup T_2^{pq} \cup T_3^{pq} \cup T_4^{pq} \cup T_5^{pq} \cup T_6^{pq}} (\dots) dx dy = 4. \end{aligned}$$

Таким образом, система (2.18) записывается в виде

$$(u_{pq,p-1q} - 2u_{pq,pq} + u_{pq,p+1q}) + (u_{pq,pq-1} - 2u_{pq,pq} + u_{pq,pq+1}) = \iint_D f \varphi_{pq} dx dy.$$

Вновь, как и в п. 1.8.3, имеем обычную конечно-разностную схему для задачи (2.1), (2.2), если для интеграла в правой части использовать приближенное значение (аналог формулы центральных прямоугольников)

$$\iint_D f \varphi_{pq} dx dy \approx f(x_p, y_q) \varphi_{pq}(x_p, y_q) h^2 = f(x_p, y_q) h^2.$$

2.3 Конечно-разностные методы в двумерном случае

В качестве альтернативного способа решения задачи Дирихле для уравнения Лапласа можно рассмотреть конечно-разностные методы.

Для более общего изложения метода решения будем рассматривать уравнение Пуассона

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad u = u(x, y), \quad (x, y) \in D = [0, a] \times [0, b], \quad (2.21)$$

где функция $f(x, y)$ заранее известна.

Пусть на границе Γ области D заданы условия

$$\begin{aligned} \alpha_1 \frac{\partial u}{\partial x}(0, y) + \beta_1 u(0, y) &= \varphi_1(y), \\ \alpha_2 \frac{\partial u}{\partial x}(a, y) + \beta_2 u(a, y) &= \varphi_2(y), \\ \alpha_3 \frac{\partial u}{\partial x}(x, 0) + \beta_3 u(x, 0) &= \varphi_3(x), \\ \alpha_4 \frac{\partial u}{\partial x}(x, b) + \beta_4 u(x, b) &= \varphi_4(x). \end{aligned} \quad (2.22)$$

На область D наложим сетку вида

$$\omega_{h_x, h_y} = \{x_i = ih_x, i = 0, 1, \dots, N, y_j = jh_y, j = 0, 1, \dots, M\}. \quad (2.23)$$

Здесь $h_x = \frac{a}{N}$, $h_y = \frac{b}{M}$.

Во внутренних узлах сетки ω_{h_x, h_y} , т. е. при $i = 1, 2, \dots, N - 1$ и $j = 1, 2, \dots, M - 1$, аппроксимируем дифференциальное уравнение (2.21) по схеме

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h_y^2} + O(h_x^2 + h_y^2) = f_{ij}, \quad (2.24)$$

где $u_{ij} = u(x_i, y_j)$, $f_{ij} = f(x_i, y_j)$.

Граничные условия аппроксимируются разделёнными разностями первого порядка

$$\begin{aligned} \alpha_1 \frac{u_{1j} - u_{0j}}{h_x} + \beta_1 u_{0j} &= \varphi_1(y_j), \\ \alpha_2 \frac{u_{Nj} - u_{N-1,j}}{h_x} + \beta_2 u_{Nj} &= \varphi_2(y_j), \\ \alpha_3 \frac{u_{i1} - u_{i0}}{h_y} + \beta_3 u_{i0} &= \varphi_3(x_i), \\ \alpha_4 \frac{u_{iM} - u_{i,M-1}}{h_y} + \beta_4 u_{iM} &= \varphi_4(x_i), \end{aligned} \quad (2.25)$$

$i = 1, 2, \dots, N - 1$, $j = 1, 2, \dots, M - 1$.

Схема (2.24) имеет второй порядок точности по переменным x и y , т. к. шаблон для конечно-разностных отношений симметричен (схема «крест»).

Система линейных алгебраических уравнений (2.24) имеет пятидиагональный вид. Здесь каждое уравнение имеет пять неизвестных и при соответствующей нумерации переменных матрица системы имеет ленточную структуру. Решать данную систему можно различными способами. Например методом матричной прогонки, итерационными методами и т. д.

Рассмотрим **разностно-итерационный метод Либмана** численного решения задачи Дирихле (2.21), (2.22). Опустим остаточный член аппроксимации $O(h_x^2 + h_y^2)$ в (2.24) и, выделяя переменную u_{ij} , запишем формулу итерационного процесса

$$u_{ij}^{k+1} = \frac{1}{4(h_x^2 + h_y^2)} (h_y^2(u_{i-1,j}^k + u_{i+1,j}^k) + h_x^2(u_{i,j-1}^k + u_{i,j+1}^k) - h_x^2 h_y^2 f_{i,j}), \quad (2.26)$$

где $i = 1, 2, \dots, N - 1$, $j = 1, 2, \dots, M - 1$, k — номер итерации.

Начальное приближение u_{ij}^0 определяется из граничных условий (2.25) при фиксированных значениях $y_j \equiv \text{const}$, $j = 1, 2, \dots, M - 1$.

Итерационный процесс Либмана прекращается, когда

$$\|u^{k+1} - u^k\| \leq \varepsilon, \quad (2.27)$$

где ε — наперед заданная погрешность.

Метод Либмана отличается довольно медленной сходимостью, что существенно при использовании мелких сеток.

2.4 Индивидуальное задание № 3

Рассматривается краевая задача (2.21)–(2.22). Требуется найти приближённое решение $u_{i,j}$, $i = 0, 1, \dots, N$, $j = 0, 1, \dots, M$ разностно-итерационным методом Либмана (2.26), используя центрально-разностную схему (2.23).

Указание 1. Для определения нормы вектора в условии (2.27) использовать выражение

$$\|u^{k+1} - u^k\| = \max_{i,j} |u_{ij}^{k+1} - u_{ij}^k|.$$

Указание 2. В индивидуальных заданиях использовано обозначение дифференциального оператора

$$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}.$$

Указание 3. Вычислить погрешность численного решения u^{k+1} с приведёнными в индивидуальных вариантах аналитическими решениями u^t . Исследовать зависимость погрешности от сеточных параметров h_x, h_y .

Вариант 1. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u(0, y) = y, \quad u(1, y) = 1 + y, \\ u(x, 0) = x, \quad u(x, 1) = x + 1. \end{cases}$$

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

Вариант 2. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u_x(0, y) = 0, \quad u(1, y) = 1 - y^2, \\ u_y(x, 0) = 0, \quad u(x, 1) = x^2 - 1. \end{cases}$$

Аналитическое решение: $U(x, y) = x^2 - y^2$.

Вариант 3. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u(0, y) = \cos y, \quad u(1, y) = e \cos y, \\ u_y(x, 0) = 0, \quad u_y(x, \frac{\pi}{2}) = -e^x. \end{cases}$$

Аналитическое решение: $U(x, y) = e^x \cos y$.

Вариант 4. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u_x(0, y) = e^y, \quad u_x(\pi, y) = -e^y, \\ u(x, 0) = \sin x, \quad u(x, 1) = e \sin x. \end{cases}$$

Аналитическое решение: $U(x, y) = \sin x e^y$.

Вариант 5. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 2(y - x) \\ u(0, y) = 0, \quad u_y(1, y) = 1 - 2y, \\ u(x, 0) = 0, \quad u_x(x, 1) = 2x - 1. \end{cases}$$

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

Вариант 6. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 2(y - x)^3 \\ u(0, y) = 0, \quad u(1, y) = y^2(y - 1), \\ u(x, 0) = 0, \quad u(x, 1) = x^2(1 - x). \end{cases}$$

Аналитическое решение: $U(x, y) = x^2 y^2 (y - x)$.

Вариант 7. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u(0, y) = \cos y, \quad u(1, y) = e^3 \cos y, \\ u_y(x, 0) = 0, \quad u_y(x, \frac{\pi}{2}) = -e^{3x}. \end{cases}$$

Аналитическое решение: $U(x, y) = e^{3x} \cos y$.

Вариант 8. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u_x(0, y) = 0, \quad u(1, y) = 2 + y^2, \\ u_y(x, 0) = 0, \quad u(x, 1) = 2x^2 + 1. \end{cases}$$

Аналитическое решение: $U(x, y) = 2x^2 + y^2$.

Вариант 9. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 2x(x - 1) + 2y(y - 1) \\ u(0, y) = 0, \quad u(1, y) = 0, \\ u(x, 0) = 0, \quad u(x, 1) = 0. \end{cases}$$

Аналитическое решение: $U(x, y) = xy(x - 1)(y - 1)$.

Вариант 10. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u(0, y) = y^2, \quad u(1, y) = 1 + y^2, \\ u_x(x, 0) = 0, \quad u_y(x, 1) = 2. \end{cases}$$

Аналитическое решение: $U(x, y) = x^2 + y^2$.

Вариант 11. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u = 0 \\ u_x(0, y) = e^{y+2}, \quad u_x(\pi, y) = -e^{y+2}, \\ u(x, 0) = e^2 \sin x, \quad u(x, 1) = e^3 \sin x. \end{cases}$$

Аналитическое решение: $U(x, y) = \sin x e^{y+2}$.

Вариант 12. Методом Либмана решить краевую задачу

$$\begin{cases} \Delta u + 2 \sin x \cos y = 0 \\ u(0, y) = 0, \quad u_x(\pi, y) = -\cos y, \\ u_y(x, 0) = 0, \quad u(x, \pi) = -\sin x. \end{cases}$$

Аналитическое решение: $U(x, y) = \sin x \cos y$.

3 Использование пакета FreeFem++ для решения пространственно двумерных задач

Для решения пространственно двумерных задач можно успешно использовать различные пакеты прикладных программ такие, как FlexPDE, Comsol, ANSYS, FreeFem и др. Перечисленные пакеты требуют минимального объема программирования и зачастую достаточно создавать рабочие коды программ на языке, близком к естественной математической записи задачи.

Приведем несколько примеров решения краевых задач при помощи пакета FreeFem++, давая лишь самые необходимые сведения о пакете, и приводя простейшие коды программ, позволяющие решать задачу. Достаточно подробно язык пакета программ FreeFem++ описан в [1, 2, 21] (см. также, <http://www.freefem.org/>).

3.1 Задача о стационарном распределении температуры

3.1.1 Постановка задачи

Рассмотрим задачу о стационарном распределении температуры в некоторой двумерной области D , на фрагментах границы которой задана температура g_1 , поток тепла g_2 и условие теплопередачи Ньютона (краевое условие третьего рода), а также имеется внутренний источник тепла $f(x, y)$. Уравнение, описывающее распределение температуры имеет вид

$$-\Delta u = f(x, y), \quad (x, y) \in D, \quad \Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}. \quad (3.1)$$

Поставим следующие краевые условия:

$$u|_{\Gamma_1} = g_1, \quad (3.2)$$

$$\frac{\partial u}{\partial n} \Big|_{\Gamma_2} = g_2, \quad (3.3)$$

$$\left(\frac{\partial u}{\partial n} + \beta u \right) \Big|_{\Gamma_3} = g_3. \quad (3.4)$$

Здесь $u(x, y)$ — температура, $f(x, y)$ — функция, заданная в области D ; $g_1(x, y)$, $g_2(x, y)$, $g_3(x, y)$ — функции, заданные на границе области D , β — заданный параметр. Для простоты считается, что коэффициент температуропроводности равен единице

Форма области D может быть любой, однако предполагается, что граница области $\partial D = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$ (возможно несвязная) является достаточно гладкой. Для определенности, считаем, что область

$$D = (0, a) \times (0, b)$$

и фрагмент границы Γ_1 является несвязным: $\Gamma_1 = \Gamma_{11} \cup \Gamma_{12}$ (см. рис. 7).

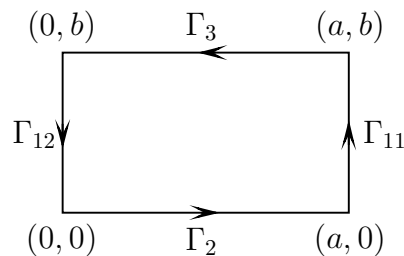


Рис. 7. Область \bar{D} . Прямоугольник $[0, a] \times [0, b]$

3.1.2 Слабая формулировка задачи

Запишем задачу (3.1)–(3.4) в слабой (вариационной) формулировке. Для этого умножим (3.1) на тестовую (пробную) функцию $v(x, y)$ и проинтегрируем по области D

$$-\iint_D v \Delta u \, dx \, dy = \iint_D v f \, dx \, dy.$$

Используя формулу Грина, получим слабую формулировку исходной задачи (3.1)–(3.4)

$$\iint_D \nabla v \cdot \nabla u \, dx \, dy - \int_{\Gamma} v \frac{\partial u}{\partial n} \, ds - \iint_D f v \, dx \, dy = 0, \quad \forall v(x, y). \quad (3.5)$$

Здесь $\partial u / \partial n$ — производная по внешней нормали к границе области D .

Ввиду того, что краевые условия (3.2)–(3.4) на фрагментах контура Γ заданы различными (см. также рис. 7), перепишем (3.5) в виде

$$\begin{aligned} & \iint_D \nabla v \cdot \nabla u \, dx \, dy - \iint_D f v \, dx \, dy - \\ & - \int_{\Gamma_1} v \frac{\partial u}{\partial n} \, ds - \int_{\Gamma_2} v \frac{\partial u}{\partial n} \, ds - \int_{\Gamma_3} v \frac{\partial u}{\partial n} \, ds = 0. \end{aligned} \quad (3.6)$$

Преобразуем интегралы по контурам Γ_2, Γ_3 , исключая производные с учетом краевых условий (3.3)–(3.4). Для контура Γ_3 с учетом краевого условия (3.4) выводим

$$- \int_{\Gamma_3} v \frac{\partial u}{\partial n} \, ds = - \int_{\Gamma_3} v (g_3 - \beta u) \, ds. \quad (3.7)$$

Аналогично, для контура Γ_2 с учетом краевого условия (3.3) имеем

$$- \int_{\Gamma_2} v \frac{\partial u}{\partial n} \, ds = - \int_{\Gamma_2} v g_2 \, ds. \quad (3.8)$$

По терминологии метода конечных элементов (см., в частности, п. 1.9) краевые условия (3.3), (3.4) являются **естественными краевыми условиями**. Грубо говоря, эти краевые условия могут быть выполнены за счет требования обращения в нуль интегралов по контуру (после исключения производных) для **всех произвольных** тестовых функций $v(x, y)$.

Интеграл по контуру Γ_1 не может быть преобразован подобным образом, т. к. условие (3.2) не содержит производных функции $u(x, y)$. Краевое условие (3.2) является **главным краевым условием**. Для того, чтобы краевое условие было выполнено, необходимо накладывать дополнительные требования на функции $v(x, y)$.

Дополнительное ограничение на функции $v(x, y)$ в данном случае аналогично **однородному** условию (3.2) и имеет вид

$$v|_{\Gamma_1} = 0. \quad (3.9)$$

Окончательно, исходная задача (3.1)–(3.4) в слабой формулировке

с учетом (3.6)–(3.8) принимает вид

$$\begin{aligned} \iint_D \nabla u \cdot \nabla v \, dx \, dy - \iint_D f v \, dx \, dy - \\ - \int_{\Gamma_1} v \frac{\partial u}{\partial n} \, ds - \int_{\Gamma_2} g_2 v \, ds - \int_{\Gamma_3} (g_3 - \beta u) v \, ds = 0, \quad \forall v \end{aligned}$$

или с учетом (3.9)

$$\begin{aligned} \iint_D \nabla u \cdot \nabla v \, dx \, dy - \int_{\Gamma_2} g_2 v \, ds - \int_{\Gamma_3} (g_3 - \beta u) v \, ds - \iint_D f v \, dx \, dy = 0, \quad (3.10) \\ \forall v|_{\Gamma_1} = 0. \end{aligned}$$

3.1.3 Слабая формулировка задачи на языке FreeFem++

Прежде чем приводить коды на языке FreeFem++, напомним поординатную форму записи следующего интеграла

$$\iint_D \nabla v \cdot \nabla u \, dx \, dy = \iint_D \left(\frac{\partial v}{\partial x} \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \frac{\partial u}{\partial y} \right) dx \, dy. \quad (3.11)$$

Удобно также использовать обозначения

$$\frac{\partial}{\partial x} = \partial_x, \quad \frac{\partial}{\partial y} = \partial_y. \quad (3.12)$$

С учетом (3.12) формула (3.11) запишется в виде

$$\iint_D \nabla v \cdot \nabla u \, dx \, dy = \iint_D (\partial_x v \partial_x u + \partial_y v \partial_y u) \, dx \, dy. \quad (3.13)$$

Приведем схему, при помощи которой задача в слабой формули-

ровке (3.10) записывается на языке FreeFem++

$$\begin{aligned}
 \iint_D \nabla v \cdot \nabla u \, dx \, dy &\rightarrow \text{int2d(Th)}(\text{dx}(u)*\text{dx}(v) + \text{dy}(u)*\text{dy}(v)) \\
 - \int_{\Gamma_2} g_2 v \, ds &\rightarrow -\text{int1d(Th, Gamma2)}(g2 * v) \\
 - \int_{\Gamma_3} (g_3 - \beta u) v \, ds &\rightarrow +\text{int1d(Th, Gamma3)}(\text{beta}*u*v) - \text{int1d(Th, Gamma3)}(g3*v) \\
 - \iint_D f v \, dx \, dy &\rightarrow -\text{int2d(Th)}(f * v) \\
 - \int_{\Gamma_1} v \frac{\partial u}{\partial n} \, ds &\rightarrow \text{on(Gamma1, u = g1)}
 \end{aligned}$$

Таким образом, интегралы, содержащиеся в выражении (3.10), практически дословно переписываются в кодах языка FreeFem++. Может быть следующая схема соответствий является даже более удобной

$$\begin{aligned}
 \iint_D (\dots) \, dx \, dy &\rightarrow \text{int2d(Th)}(\dots) \\
 \int_{\Gamma_i} (\dots) \, ds &\rightarrow \text{int1d(Th, GammaI)}(\dots) \\
 \partial_x(\dots) \partial_x(\dots) &\rightarrow \text{dx}(\dots) * \text{dx}(\dots) \\
 \partial_y(\dots) \partial_y(\dots) &\rightarrow \text{dy}(\dots) * \text{dy}(\dots)
 \end{aligned}$$

Интуитивно понятно, что Th в выражении int2d(Th) обозначает область интегрирования (более точно, Th — триангуляция области D , см. ниже), $(\text{Th}, \text{GammaI})$ в выражении int1d(Th, GammaI) указывает область и фрагмент ее границы, по которой проводится интегрирование.

Также должно быть ясно, что краевое условие (3.2) требует специальной формы записи. В подынтегральном выражении для интеграла по границе Γ_1 производная $\partial u / \partial n$ не может быть исключена при помощи каких-либо преобразований. Именно поэтому краевое условие (3.2) задается непосредственно в том виде, как оно записано, при помощи ключевого слова on в форме: $\text{on(Gamma1, u = g1)}$.

Фрагмент программы для решения задачи на языке FreeFem++ выглядит следующим образом (см. также полный текст программы, приведенный ниже на с. 64)

```
24 solve Poisson(u,v) = int2d(Th)( dx(u)*dx(v) + dy(u)*dy(v) )
25                       - int1d(Th,Gamma2)( g2 * v )
26                       + int1d(Th,Gamma3)( beta * u*v )
27                       - int1d(Th,Gamma3)( g3 * v )
28                       - int2d(Th)( f * v )
29                       + on( Gamma11, u = g1 )
30                       + on( Gamma12, u = g1 ) ;
```

Обратим внимание, что нумерация строк использована для удобства ссылок и при использовании кода должна быть удалена, т. к. FreeFem++ **не требует** нумерации строк.

✓. Интегралы `int1d` в случае фрагмента границы Γ_3 нельзя записывать в виде одного интеграла. Дело в том, что один из них, содержащий произведение v на известную функцию, является линейной формой относительно тестовой функции v , а другой, содержащий произведение uv , является билинейной формой относительно функций u, v . Язык FreeFem++ не позволяет смешивать эти понятия (выражения).

3.1.4 Задание области

Для того чтобы определить область D в FreeFem++ достаточно указать границу области D . Это возможно сделать при помощи параметрического задания отдельных частей границы с использованием ключевого слова `border`. Приведем пример задания области D в случае прямоугольника $[0, a] \times [0, b]$ (см. рис. 7)

```
6 border Gamma2 (t=0,1) { x=a*t; y=0; };
7 border Gamma11 (t=0,1) { x=a; y=b*t; };
8 border Gamma3 (t=0,1) { x=a*(1-t); y=b; };
9 border Gamma12 (t=0,1) { x=0; y=b*(1-t); };
```

Граница $\Gamma_1 = \Gamma_{11} \cup \Gamma_{12}$ предполагается несвязной. Именно поэтому для каждого ее связного фрагмента использованы различные идентификаторы `Gamma11` и `Gamma12`.

Интуитивно понятно, что, например, строка 6 соответствует параметрической форме записи для отрезка прямой линии $x \in [0, a]$, $y = 0$

$$x = at, \quad y = 0, \quad 0 \leq t \leq 1.$$

Заметим, что порядок определения фрагментов границы не имеет значения, т. е. строчки с операторами `border` в приведенном выше фрагменте программы могут быть переставлены. Однако важно, чтобы при возрастании параметра t обход области D совершался в направлении **против часовой стрелки**.

3.1.5 Полный код программы

Зададим для определенности следующие функции и параметры для исходной задачи (3.1)–(3.4)

$$\bar{D} = \{(x, y) : [0, a] \times [0, b]\}, \quad a = 1, \quad b = 1,$$

$$f(x, y) = \sin 2\pi x \sin 2\pi y, \quad g_1 = 0, \quad g_2 = 1, \quad g_3 = 1, \quad \beta = 1. \quad (3.14)$$

Приведем полный код программы для построения численного решения задачи

```

1  real a=1.0; // ширина области D
2  real b=1.0; // высота области D
3  int n=20;   // вспомогательный параметр для триангуляции обл. D
4  // задание границ области D (прямоугольник [0,a]x[0,b])
5  // сохраняем ориентацию контура -- против часовой стрелки
6  border Gamma2(t=0,1){ x=a*t; y=0; };           // bottom
7  border Gamma11(t=0,1){ x=a; y=b*t; };          // right
8  border Gamma3(t=0,1){ x=a*(1-t); y=b; };       // top
9  border Gamma12(t=0,1){ x=0; y=b*(1-t); };      // left
10 // построение сетки,
11 // каждый фрагмент границы разбит на 20 отрезков
12 mesh Th = buildmesh(Gamma2(n)+Gamma11(n)+Gamma3(n)+Gamma12(n));
13 // plot(Th,wait=1); // для визуализации построенной сетки

```



```
14 // задание пространства конечных элементов
15 fespace Vh(Th,P2);
16 // задание на пространстве Vh искомой ф-ции u и пробной ф-ции v
17 Vh u,v;
18 // определение функций и параметров исходной задачи
19 func f = sin(2*pi*x)*sin(2*pi*y);
20 func g1 = 0;
21 func g2 = 1;
22 func g3 = 1;
23 real beta=1;
24 // запись слабой (вариационной) формулировки задачи и ее решение
25 solve Poisson(u,v) = int2d(Th)( dx(u)*dx(v) + dy(u)*dy(v) )
26                       - int1d(Th,Gamma2)( g2 * v )
27                       + int1d(Th,Gamma3)( beta * u*v )
28                       - int1d(Th,Gamma3)( g3 * v )
29                       - int2d(Th)( f * v )
30                       + on( Gamma11, u = g1 )
31                       + on( Gamma12, u = g1 ) ;
32 plot(u); // графическое представление решения
```

Результат решения задачи (3.1)–(3.4) представлен на рис. 8, на котором изображены линии уровня (изолинии или изотермы, если u температура) функции $u(x, y)$. Заметим, что на экране дисплея изолинии будут цветными.

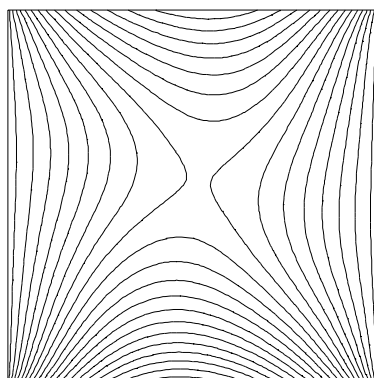


Рис. 8. Изолинии функции $u(x, y)$ (температуры) — решение задачи (3.1)–(3.4)

Сделаем некоторые пояснения:

1) Th — идентификатор генерируемой сетки.

2) $Vh(Th, P2)$ — идентификатор пространства конечных элементов. Имя Vh может быть любым, Th — должен совпадать с именем генерируемой сетки, $P2$ — наименование типа конечных элементов (зарезервированное слово в FreeFem++).

3) `soIve Poisson(u,v)` — имя `Poisson` задается пользователем и может быть любым. Ключевое слово `soIve` означает, что задача формулируется и одновременно решается.

На этом этапе использования языка FreeFem++ можно не задумываться над строками 10–14 программы. Интуитивно понятно, что при помощи этих строк задается триангуляция области D и выбирается способ аппроксимации решения некоторыми конечноэлементными функциями. Заметим только, что использованный тип конечных элементов ($P2$) соответствует аппроксимации кусочно-непрерывными квадратичными функциями.

3.2 Решение задачи о распределении температуры в областях сложной формы

Одной из важных особенностей метода конечных элементов и, в частности, языка FreeFem++ является возможность решать задачу в области сравнительно произвольной формы с достаточно гладкой границей. В языке FreeFem++ реализован эффективный алгоритм триангуляции области, требующий для генерации треугольной сетки задания лишь количества вершин треугольников на границах. Замечательно то, что основная часть кода программы для решения задачи в прямоугольнике остается прежней. Переход к другой области осуществляется заменой только строк 6–9, определяющих границу.

3.2.1 Решение задачи для круга

Пусть область \bar{D} представляет собой круг $x^2 + y^2 \leq 1$ единичного радиуса. Границу области \bar{D} можно задать параметрически при помощи соотношений $x = \cos t$, $y = \sin t$, $t \in [0, 2\pi]$. Предположим, что граница

области разбита на четыре фрагмента

$$\Gamma_{11}: x = \cos t, \quad y = \sin t, \quad t \in [0, 0,33\pi];$$

$$\Gamma_2: x = \cos t, \quad y = \sin t, \quad t \in [0,33\pi, \pi];$$

$$\Gamma_3: x = \cos t, \quad y = \sin t, \quad t \in [\pi, 1,41\pi];$$

$$\Gamma_{12}: x = \cos t, \quad y = \sin t, \quad t \in [1,41\pi, 2\pi].$$

Операторы, задающие границу, в этом случае будут следующими:

```
border Gamma11(t=0,0.33*pi) { x=cos(t); y=sin(t); };  
border Gamma2 (t=0.33*pi,pi) { x=cos(t); y=sin(t); };  
border Gamma3 (t=pi,1.41*pi) { x=cos(t); y=sin(t); };  
border Gamma12(t=1.41*pi,2.0*pi){ x=cos(t); y=sin(t); };
```

Решение задачи (изолинии температуры) (3.1)–(3.4) для параметров, определенных соотношениями (3.14), приведено на рис. 9.

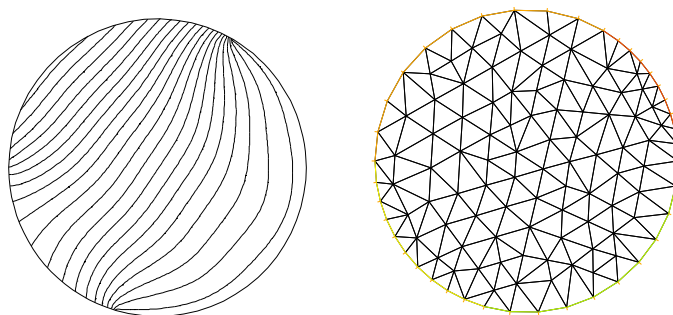


Рис. 9. Изолинии температуры для задачи в круге и триангуляция

3.2.2 Распределение температуры в четырехугольнике

Пусть D представляет собой четырехугольник с координатами вершин (x_i, y_i) ($i = 0, \dots, 3$). В этом случае код, задающий границу, может быть записан в виде

```
int m=4;
real[int] xx(m), yy(m);
xx[0]=0;    yy[0]=0.2;
xx[1]=0.5;  yy[1]=0;
xx[2]=0.8;  yy[2]=0.8;
xx[3]=0.5;  yy[3]=1;
border Gamma2(t=0,1) {x=xx[0]*(1-t)+xx[1]*t;
                      y=yy[0]*(1-t)+yy[1]*t;};
border Gamma11(t=0,1){x=xx[1]*(1-t)+xx[2]*t;
                      y=yy[1]*(1-t)+yy[2]*t;};
border Gamma3(t=0,1) {x=xx[2]*(1-t)+xx[3]*t;
                      y=yy[2]*(1-t)+yy[3]*t;};
border Gamma12(t=0,1){x=xx[3]*(1-t)+xx[0]*t;
                      y=yy[3]*(1-t)+yy[0]*t;};
```

Решение задачи (изолинии температуры) (3.1)–(3.4) для параметров, определенных соотношениями (3.14), приведено на рис. 10.

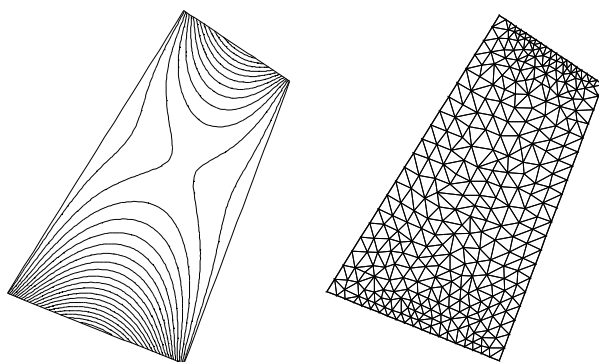


Рис. 10. Изолинии температуры для задачи в четырехугольнике и триангуляция

Для области, представляющей собой полигон, координаты вершин которого (x_i, y_i) ($i = 0, \dots, m$) известны, граница может быть задана аналогичным образом. Однако, при этом потребуются и другие изменения в программе. Необходимо будет переписать строку с операто-

ром генерации сетки `mesh Th = buildmesh(...)` и строки, задающие краевые условия на соответствующих участках границы полигона.

3.2.3 Решение задачи в криволинейной области

Две точки на плоскости (x_1, y_1) , (x_2, y_2) можно соединить кривой линией, используя параметрическое представление кривой, например,

1. $x = x_1 \cos t + x_2 \sin t, \quad y = y_1 \cos t + y_2 \sin t, \quad t \in [0, \pi/2];$
2. $x = x_1(1 - t^2) + x_2 t^2, \quad y = y_1(1 - t^2) + y_2 t^2, \quad t \in [0, 1].$

Приведем код для задания криволинейной области (см. рис. 11)

```
int m=4;
real[int] xx(m), yy(m);
xx[0]=0;    yy[0]=0.4;  xx[1]=0.5;    yy[1]=0;
xx[2]=1;    yy[2]=1;   xx[3]=0.15;   yy[3]=0.35;
border Gamma2(t=0,1) {x=xx[0]*(1-t)+xx[1]*t;
                      y=yy[0]*(1-t)+yy[1]*t;};
border Gamma11(t=0,1){x=xx[1]*(1-t)+xx[2]*t;
                      y=yy[1]*(1-t)+yy[2]*t;};
border Gamma3(t=0,1) {x=xx[2]*(1-t^2)+xx[3]*t^2;
                      y=yy[2]*(1-t^2)+yy[3]*t^2;};
border Gamma12(t=0,0.5*pi){x=xx[3]*cos(t)+xx[0]*sin(t);
                           y=yy[3]*cos(t)+yy[0]*sin(t);};
```

Решение задачи (изолинии температуры) (3.1)–(3.4) для параметров, определенных соотношениями (3.14), приведено на рис. 11.

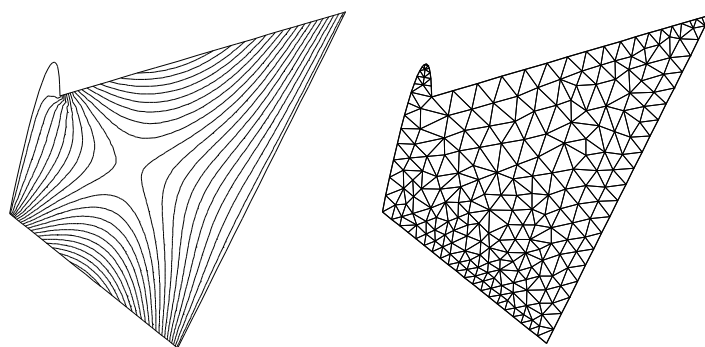


Рис. 11. Изолинии температуры и триангуляция

3.2.4 Решение задачи в области с отверстием

Приведем фрагмент кода, позволяющий рассмотреть задачу, например, в области D с круглым отверстием (см. рис. 12).

```
border G11(t=0,0.33*pi) { x=cos(t); y=sin(t); };
border G2 (t=0.33*pi,pi){ x=cos(t); y=sin(t); };
border G3 (t=pi,2*pi)   { x=cos(t); y=sin(t); };
border G12(t=0,2*pi)   { x=0.5*cos(t); y=0.5*sin(t); };
```

Решение задачи (изолинии температуры) (3.1)–(3.4) для параметров, определенных соотношениями (3.14), приведено на рис. 12.

Обратим внимание, что для сохранения ориентации контура в операторе `mesh` следует записывать внутренний контур в следующем формате `G12(-n)`. Соответствующая строка кода, генерирующая сетку, имеет вид

```
mesh Th = buildmesh(G2(n)+G11(n)+G3(n)+G12(-n));
```

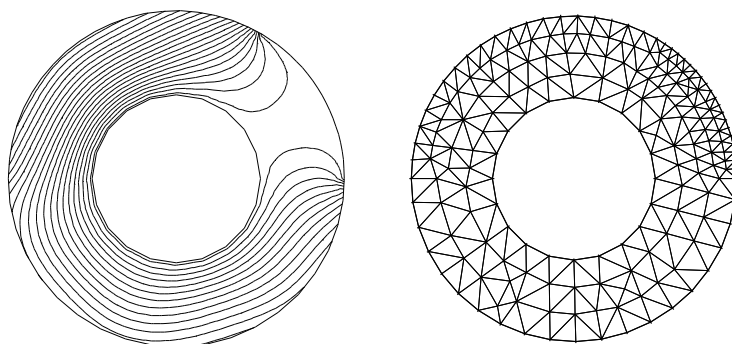


Рис. 12. Изолинии температуры для задачи в кольце и триангуляция

3.2.5 Решение задачи с переменной теплопроводностью

Стационарное распределение температуры в некоторой области D описывается уравнением (уравнение баланса энергии)

$$\operatorname{div} \mathbf{q} = f, \quad (3.15)$$

где \mathbf{q} — плотность потока тепла, f — плотность внутренних источников.

Плотность потока тепла \mathbf{q} связана с температурой u законом теплопроводности Фурье

$$\mathbf{q} = -\kappa \nabla u, \quad (3.16)$$

где κ — коэффициент теплопроводности ($\kappa \geq 0$).

Подстановка (3.16) в (3.14) дает стандартный вид уравнения теплопроводности с переменным коэффициентом теплопроводности

$$-\operatorname{div}(\kappa \nabla u) = f. \quad (3.17)$$

Умножая (3.15) на v и интегрируя по области D , получим

$$-\iint_D \mathbf{q} \cdot \nabla v \, dx \, dy + \int_{\partial D} v \mathbf{q} \cdot \mathbf{n} \, ds - \iint_D v f \, dx \, dy = 0 \quad (3.18)$$

или с учетом (3.16)

$$\iint_D \kappa \nabla u \cdot \nabla v \, dx \, dy + \int_{\partial D} v \mathbf{q} \cdot \mathbf{n} \, ds - \iint_D v f \, dx \, dy = 0. \quad (3.19)$$

Далее считаем, что на границе области D вновь заданы краевые условия вида (3.2)–(3.4), то есть

$$u|_{\Gamma_1} = g_1, \quad (3.20)$$

$$\frac{\partial u}{\partial \mathbf{n}} \Big|_{\Gamma_2} = g_2, \quad (3.21)$$

$$\left(\frac{\partial u}{\partial \mathbf{n}} + \beta u \right) \Big|_{\Gamma_3} = g_3. \quad (3.22)$$

Одним из достоинств метода конечных элементов и языка FreeFem++ является возможность несложного модифицирования кода программы для решения задачи с коэффициентами, зависящими от координат.

Пусть, например, $\kappa = \kappa(x, y)$. Оператор, задающий функцию κ , имеет вид (для определенности, $\kappa(x, y) = xy$)

```
vh kappa = x * y;
```

Эту строку можно вставить между строками 23, 24 (см. с. 64).

Для решения с помощью языка FreeFem++ достаточно вместо строки 25 (см. с. 64)

```
solve Poisson(u,v) = int2d(Th)(dx(u)*dx(v) + dy(u)*dy(v))
```

записать строку

```
solve Poisson(u,v) =  
    int2d(Th)(kappa*(dx(u)*dx(v) + dy(u)*dy(v)))
```

3.3 Индивидуальное задание № 4

При помощи пакета FreeFem++ решить задачи индивидуального задания № 3.

Список литературы

- [1] Жуков М. Ю., Ширяева Е. В. Использование пакета конечных элементов FreeFem++ для задач гидродинамики, электрофореза и биологии. Ростов-на-Дону: Изд. ЮФУ, 2008. 256 с.
- [2] Жуков М. Ю., Ширяева Е. В. Решение задач математической физики при помощи пакета конечных элементов FreeFem++. Ростов-на-Дону: Изд. ЮФУ, 2014. 256 с.
- [3] Hecht F., Pironneau O., Le Hyaric A., Ohtsuka K. FreeFem++. Version 2.17-1. <http://www.freefem.org/ff++>.
- [4] Марчук Г. И. Методы вычислительной математики. М.: Наука, 1980.
- [5] Марчук Г. И., Агошков В. И. Введение в проекционно-сеточные методы. М.: Наука. Главная редакция физико-математической литературы, 1981.
- [6] Красносельский М. А., Вайникко А. Г., Забрейко П. П., Рutiцкий Я. Б., Стеценко В. Я. Приближенное решение операторных уравнений.. М.: Наука. Главная редакция физико-математической литературы, 1969.
- [7] Бахвалов Н. С., Жидков Н. П., Кобельков Г. М. Численные методы. М.: Наука, 1987.
- [8] Деклу Ж. Метод конечных элементов. М.: Мир, 1976.
- [9] Деммель Дж. Вычислительная линейная алгебра. Теория и приложения. М.: Мир, 2001.
- [10] Зенкевич О. Метод конечных элементов в технике. М.: Мир, 1975.
- [11] Зенкевич О., Морган К. Конечные элементы и аппроксимация. М.: Мир, 1986.

- [12] [Сабоннадьер Ж.-К., Кулон Ж.-Л.](#) Метод конечных элементов и САПР. М.: Мир, 1989.
- [13] [Оден Дж.](#) Конечные элементы в нелинейной механике сплошных сред. М.: Мир, 1976.
- [14] [Норри Д., де Фриз Ж.](#) Введение в метод конечных элементов. М.: Мир, 1981.
- [15] [Стренг Г., Фикс Дж.](#) Теория метода конечных элементов. М.: Мир, 1977.
- [16] [Митчелл Э., Уэйт Р.](#) Метод конечных элементов для уравнений с частными производными. М.: Мир, 1981.
- [17] [Михлин С. Г.](#) Вариационные методы в математической физике. М.: Наука, 1970.
- [18] [Михлин С. Г.](#) Численная реализация вариационных методов. М.: Наука, 1966.
- [19] [Лионс Ж.-Л., Мадженес Э.](#) Неоднородные граничные задачи и их приложения. М.: Мир, 1971.
- [20] [Hecht F.](#) The mesh adapting software: bamg. INRIA report 1998.
- [21] [Hecht F.](#) New development in FreeFem++. J. Numer. Math. 20 (2012), no. 3–4, 251–265. 65Y15

При создании электронного пособия использовались системы $\text{\LaTeX} 2_{\epsilon}$, \XeTeX и материалы книги:

[Жуков М. Ю., Ширяева Е. В.](#) $\text{\LaTeX} 2_{\epsilon}$: искусство набора и вёрстки текстов с формулами. Ростов н/Д: Изд-во ЮФУ, 2009. 192 с.

Список иллюстраций

1	Разбиение отрезка	32
2	Набор базисных функций $\varphi_k(x)$ для задачи Дирихле . . .	32
3	Набор базисных функций $\varphi_k(x)$ в случае краевых условий третьего рода	39
4	Функции $\varphi_1(x), \varphi_2(x)$	43
5	Пример триангуляции области $\bar{D} = [0, 1] \times [0, 1]$	48
6	Носитель функции $\varphi_{ik}(x, y)$ (конечный элемент)	49
7	Область \bar{D} . Прямоугольник $[0, a] \times [0, b]$	59
8	Изолинии функции $u(x, y)$ (температуры) — решение за- дачи	65
9	Изолинии температуры для задачи в круге и триангуляция	67
10	Изолинии температуры для задачи в четырехугольнике и триангуляция	68
11	Изолинии температуры и триангуляция	69
12	Изолинии температуры для задачи в кольце и триангуля- ция	70

Предметный указатель

вариационная форма, 18

закон

 Фурье, 71

конечный элемент, 32

краевое условие

 главное, 42, 60

 естественное, 42, 60

носитель функции, 31

сильное решение, 18

слабая форма, 18

слабое решение, 19, 38, 47

триангуляция, 48

финитная функция, 31

функция

 Куранта, 48

 базисная, 19

 пробная, 19

 тестовая, 19

Об авторах

Долгих Татьяна Федоровна — ассистент кафедры «**Вычислительная математика и математическая физика**» института математики, механики и компьютерных наук им. И. И. Воровича ФГАОУВО «Южный федеральный университет».


Область научных интересов — течения вязкой жидкости, массоперенос в многокомпонентных смесях, метод конечных элементов, вычислительная математика, программирование.







Ширяева Елена Владимировна — кандидат физико-математических наук, доцент кафедры «**Вычислительная математика и математическая физика**» института математики, механики и компьютерных наук им. И. И. Воровича ФГАОУВО «Южный федеральный университет».

Область научных интересов — массоперенос в многокомпонентных смесях, конечно-разностные методы, метод конечных элементов, течения вязкой жидкости, вычислительная математика, программирование.

Интерфейс пользователя

Навигация по электронному документу

Навигация по электронному документу может осуществляться с помощью клавиш и колесика мыши, клавиатуры — используются стандартные горячие клавиши программы Adobe Acrobat; элементов управления программы Adobe Acrobat (панель навигации и список закладок в окне Bookmarks), а также панели навигации учебного пособия  (см. правый верхний угол страницы):

Пиктограмма	Действие
	предыдущая страница
	следующая страница
	предыдущий просмотр
	следующий просмотр
	переход на страницу N
	поиск слов в документе

Список горячих клавиш для работы с электронным документом в программе Adobe Acrobat

Действие	Комбинация клавиш
Полноэкранный режим	Ctrl + L
Выход из полноэкранного режима	Esc
Растянуть по ширине экрана	Ctrl + 2
Переход к началу документа	Home или Shift + Ctrl + Page Up или Shift + Ctrl + Up Arrow
Переход к концу документа	End или Shift+Ctrl+Page Down или Shift+Ctrl+Down Arrow
Переход на страницу N	Shift + Ctrl + N
Предыдущий экран	Page Up или Return
Следующий экран	Page Down или Shift + Return
Предыдущая страница	Left Arrow или Ctrl + Page Up
Следующая страница	Right Arrow или Ctrl + Page Down
Предыдущий просмотр	Alt + Left Arrow
Следующий просмотр	Alt + Right Arrow