

**Цель работы:** получение навыков разработки алгоритмов решения краевой задачи при реализации моделей, построенных на ОДУ второго порядка.

# 1 Теоретическая часть

Задана математическая модель.

Уравнение для функции  $T(x)$ :

$$\frac{d}{dx} \left( K(x) \frac{dT}{dx} \right) - \frac{2}{R} \alpha(x) T + \frac{2T_0}{R} \alpha(x) = 0 \quad (1)$$

Краевые условия:

$$\begin{cases} x = 0, & -k(0) \frac{dT}{dx} = F_0 \\ x = l, & -k(l) \frac{dT}{dx} = \alpha_N (T(l) - T_0) \end{cases} \quad (2)$$

Функции  $k(x), \alpha(x)$  заданы:

$$k(x) = \frac{a}{x-b} \quad (3)$$

$$\alpha(x) = \frac{c}{x-d} \quad (4)$$

Константы  $a, b$  следует найти из условий  $k(0) = k_0, k(l) = k_N$ , а константы  $c, d$  из условий  $\alpha(0) = \alpha_0, \alpha(l) = \alpha_N$ .

$$a = -k_0 b \quad (5)$$

$$b = \frac{k_N l}{k_N - k_0} \quad (6)$$

$$c = -\alpha_0 d \quad (7)$$

$$d = \frac{\alpha_0 l}{\alpha_N - \alpha_0} \quad (8)$$

Разностная схема

$$\begin{aligned} A_n y_{n-1} - B_n y_n + C_n y_{n+1} &= -D_n, 1 \leq n \leq N-1 \\ K_0 y_0 + M_0 y_1 &= P_0 \\ K_N y_N + M_N y_{N-1} &= P_N \end{aligned} \quad (9)$$

, где

$$A_n = \frac{x_{n+\frac{1}{2}}}{h}, \quad (10)$$

$$B_n = A_n + C_n + p_n h, \quad (11)$$

$$C_n = \frac{x_{n-\frac{1}{2}}}{h}, \quad (12)$$

$$D_n = f_n h \quad (13)$$

Для вычисления используем метод средних:

$$x_{n \pm \frac{1}{2}} = \frac{k_n + k_{n \pm 1}}{2} \quad (14)$$

Система решается в два прохода: прямой и обратный.

В прямом проходе вычисляем прогоночные коэффициенты  $\varepsilon$  и  $\eta$ . Начальные значения:

$$\begin{aligned} \varepsilon_1 &= -\frac{M_0}{K_0} \\ \eta_1 &= \frac{P_0}{K_0} \end{aligned} \quad (15)$$

Вычисляем массивы прогоночных коэффициентов  $\varepsilon, \eta$ :

$$\begin{aligned}\varepsilon_{n+1} &= \frac{C_n}{B_n - A_n \varepsilon_n} \\ \eta_{n+1} &= \frac{D_n + A_n \eta_n}{B_n - A_n \varepsilon_n}\end{aligned}\quad (16)$$

Обратный проход.

Определим  $y_N$  - значение функции в последней точке:

$$y_N = \frac{P_N - M_N \eta_N}{K_N + M_N \varepsilon_N} \quad (17)$$

По основной прогоночной формуле находятся все значения неизвестных  $y_n$ :

$$y_n = \varepsilon_{n+1} y_{n+1} + \eta_{n+1} \quad (18)$$

Таким образом, массив, полученный после прогонки и есть искомый массив  $T(x)$ .

## Краевые условия

Обозначим:

$$\begin{aligned}F &= -k(x) \frac{dT}{dx} \\ p(x) &= \frac{2}{R} \alpha(x) \\ f(x) &= \frac{2T_0}{R} \alpha(x) \\ p_n &= p(x_n), \quad f_n = f(x_n)\end{aligned}\quad (19)$$

Разностные аналоги краевых условий при  $x = 0$  (из лекции № 7):

$$\left(x_{\frac{1}{2}} + \frac{h^2}{8} p_{\frac{1}{2}} + \frac{h^2}{4} p_0\right) \cdot y_0 - \left(x_{\frac{1}{2}} - \frac{h^2}{8} p_{\frac{1}{2}}\right) \cdot y_1 = \left(hF_0 + \frac{h^2}{4}(f_{\frac{1}{2}} + f_0)\right) \quad (20)$$

Разностные аналоги краевых условий при  $x = l$ :

Примем простую аппроксимацию:

$$\begin{aligned}p_{\frac{1}{2}} &= \frac{p_0 + p_1}{2} \\ f_{\frac{1}{2}} &= \frac{f_0 + f_1}{2}\end{aligned}\quad (21)$$

Проинтегрируем уравнение (1) на отрезке  $[x_{N-\frac{1}{2}}, x_N]$  с учетом замен (19).

$$-\int_{x_{N-\frac{1}{2}}}^{x_N} \frac{dF}{dx} dx - \int_{x_{N-\frac{1}{2}}}^{x_N} p(x) T dx + \int_{x_{N-\frac{1}{2}}}^{x_N} f(x) dx = 0$$

Второй и третий интегралы вычислим методом трапеций

$$F_{N-\frac{1}{2}} - F_N - \frac{p_{N-\frac{1}{2}} y_{N-\frac{1}{2}} + p_N y_N}{4} h + \frac{f_{N-\frac{1}{2}} + f_N}{4} h = 0$$

Учтем, что

$$\begin{aligned}F_{N-\frac{1}{2}} &= x_{N-\frac{1}{2}} \frac{y_{N-1} - y_N}{h} \\ F_N &= \alpha_N (y_N - T_0) \\ y_{N-\frac{1}{2}} &= \frac{y_N + y_{N-1}}{2}\end{aligned}$$

После подстановки и приведения подобных получаем:

$$\left(-\frac{x_{N-\frac{1}{2}}}{h} - \alpha_N - \frac{2p_N - 1}{16} h\right) \cdot y_N + \left(\frac{x_{N-\frac{1}{2}}}{h} - \frac{2p_N - 1}{16} h\right) \cdot y_{N-1} = -\alpha_N T_0 - \frac{h}{4} (f_{N-\frac{1}{2}} + f_N) \quad (22)$$

Заданы начальные параметры:

$k_0 = 0.4$  Вт/см К,  
 $k_N = 0.1$  Вт/см К,  
 $\alpha_0 = 0.05$  Вт/см<sup>2</sup> К,  
 $\alpha_N = 0.01$  Вт/см<sup>2</sup> К,  
 $l = 10$  см,  
 $T_0 = 300$ К,  
 $R = 0.5$  см,  
 $F_0 = 50$  Вт/см<sup>2</sup>.

## Физическое содержание задачи

Сформулированная математическая модель описывает температурное поле  $T(x)$  вдоль цилиндрического стержня радиуса  $R$  и длиной  $l$ , причем  $R \ll l$  и температуру можно принять постоянной по радиусу цилиндра. Ось  $x$  направлена вдоль оси цилиндра и начало координат совпадает с левым торцом стержня. Слева при  $x = 0$  цилиндр нагружается тепловым потоком. Стержень обдувается воздухом, температура которого равна  $T_0$ . В результате происходит съём тепла с цилиндрической поверхности и поверхности правого торца при  $x = l$ . Функции  $k(x), \alpha(x)$  являются, соответственно, коэффициентами теплопроводности материала стержня и теплоотдачи при обдуве.

## 2 Практическая часть

В листинге 1 представлен код программы:

Листинг 1: Листинг программы

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 # коэффициенты теплопроводности и теплоотдачи
5 def k(x):
6     return a / (x - b)
7
8 def alpha(x):
9     return c / (x - d)
10
11
12 def p(x):
13     return 2 * alpha(x) / R
14
15 def f(x):
16     return 2 * alpha(x) * T0 / R
17
18
19 # метод средних
20 def x_plus_half(x):
21     return (k(x) + k(x + h)) / 2
22
23 def x_minus_half(x):
24     return (k(x) + k(x - h)) / 2
25
26
27 # коэффициенты разностной схемы
28 def A(x):
29     return x_plus_half(x) / h
30
31 def C(x):
32     return x_minus_half(x) / h
33
```

```

34 def B(x):
35     return A(x) + C(x) + p(x) * h
36
37 def D(x):
38     return f(x) * h
39
40
41 # левое граничное условие
42 def left_boundary_condition():
43     K0 = x_plus_half(0) + h * h * (p(0) + p(h)) / 16 + h * h * p(0) / 4
44     M0 = -x_plus_half(0) + h * h * (p(0) + p(h)) / 16
45     P0 = h * F0 + h * h / 4 * ((f(0) + f(h)) / 2 + f(0))
46     return K0, M0, P0
47
48 # правое граничное условие
49 def right_boundary_condition():
50     KN = -x_minus_half(1) / h - aN - p(1) * h / 4 - ((p(1) + p(1 - h)) *
51         h) / 16
52     MN = x_minus_half(1) / h - ((p(1) + p(1 - h)) * h) / 16
53     PN = -aN * T0 - h * (f(1) + f(1 - h) + f(1)) / 8
54     return KN, MN, PN
55
56 if __name__ == "__main__":
57     K0 = 0.4
58     KN = 0.1
59     a0 = 0.05
60     aN = 0.01
61     l = 10
62     T0 = 300
63     R = 0.5
64     F0 = 0
65     h = 1e-3
66
67     # параметры коэффициентов теплопроводности и теплоотдачи
68     b = (KN * l) / (KN - K0)
69     a = -K0 * b
70     d = (aN * l) / (aN - a0)
71     c = -a0 * d
72
73     K0, M0, P0 = left_boundary_condition()
74     KN, MN, PN = right_boundary_condition()
75
76     # прямой ход
77     # массивы прогоночных коэффициентов
78     eps = [0]
79     eta = [0]
80
81     eps1 = -M0 / K0
82     eta1 = P0 / K0
83
84     eps.append(eps1)
85     eta.append(eta1)
86
87     x = h
88     n = 1
89     while x + h < l:
90         eps.append((C(x) / (B(x) - A(x) * eps[n])))
91         eta.append((A(x) * eta[n] + D(x)) / (B(x) - A(x) * eps[n]))
92         n += 1
93         x += h

```

```

94
95     # обратный ход
96     t = [0] * (n + 1)
97     # значение функции в последней точке
98     t[n] = (PN - MN * eta[n]) / (KN + MN * eps[n])
99
100     for i in range(n - 1, -1, -1):
101         t[i] = eps[i + 1] * t[i + 1] + eta[i + 1]
102
103     x = [i for i in np.arange(0, 1, h)]
104
105     plt.plot(x, t[:-1])
106     plt.xlabel("X, cm")
107     plt.ylabel("T, K")
108     plt.grid()
109     plt.show()

```

## Результат работы программы

1. График зависимости температуры  $T(x)$  от координаты  $x$  при заданных выше параметрах.

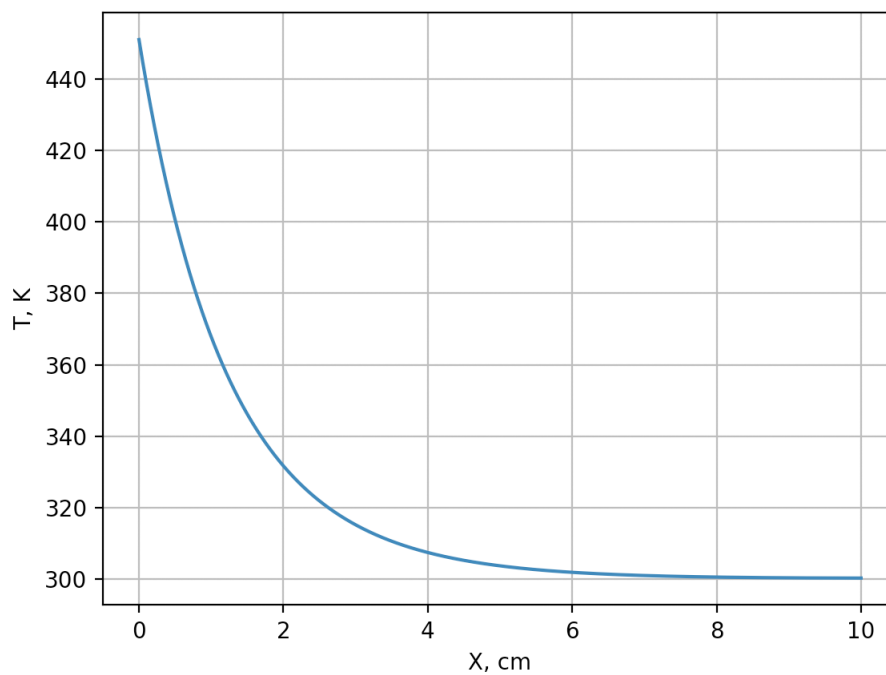


Рис. 1: График зависимости температуры  $T(x)$  от координаты  $x$  при заданных выше параметрах

2. График зависимости  $T(x)$  при  $F_0 = -10$  Вт/см<sup>2</sup>.

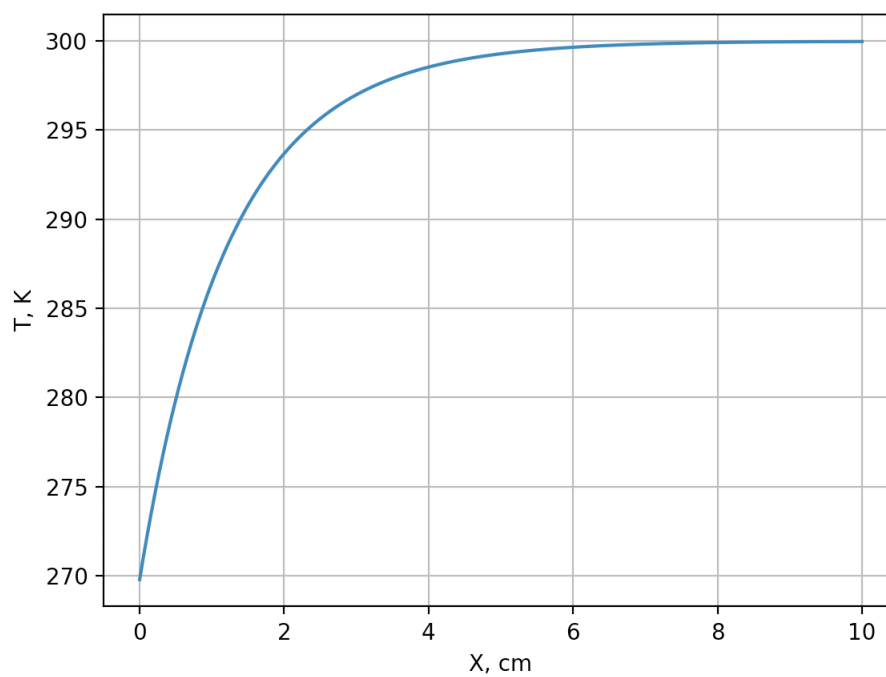


Рис. 2: График зависимости  $T(x)$  при  $F_0 = -10$  Вт/см<sup>2</sup>

3. График зависимости  $T(x)$  при увеличенных значениях  $\alpha(x)$  (например, в 3 раза).

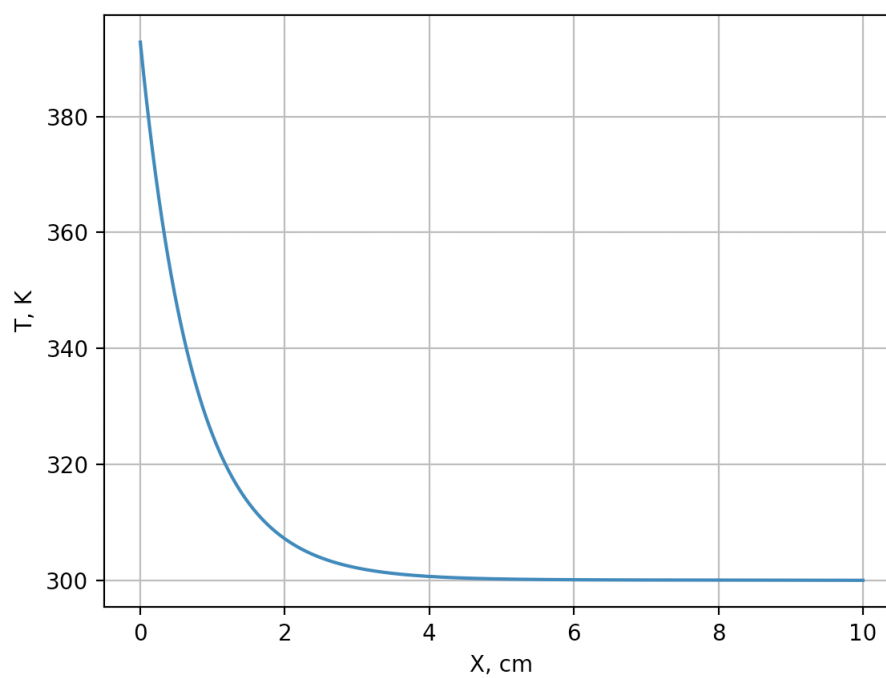


Рис. 3: График зависимости  $T(x)$  при увеличенных значениях  $\alpha(x)$  (например, в 3 раза)

4. График зависимости  $T(x)$  при  $F_0 = 0$ .

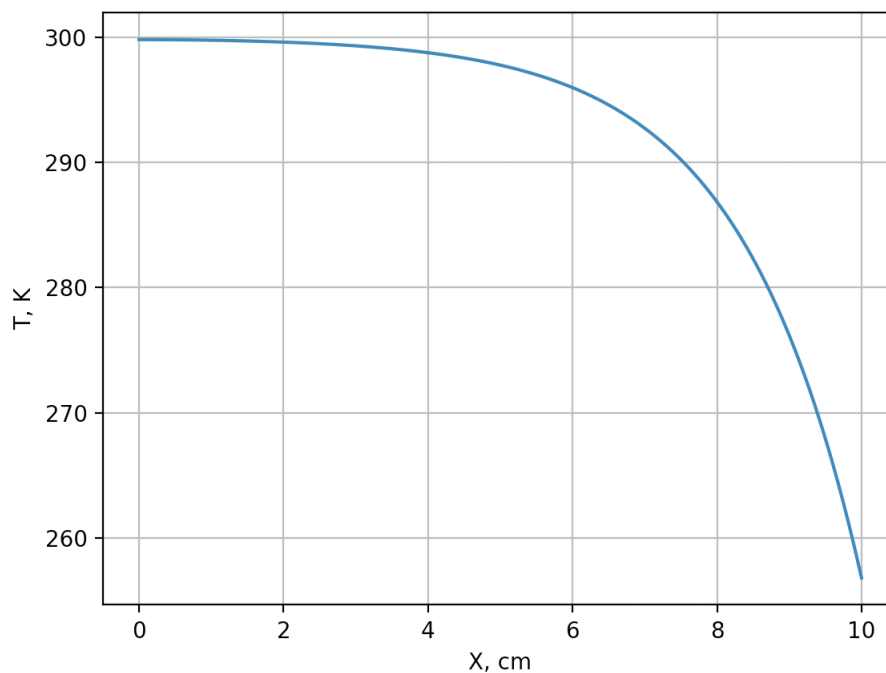


Рис. 4: График зависимости  $T(x)$  при  $F_0 = 0$

### 3 Ответы на вопросы

1. Какие способы тестирования программы можно предложить?

Увеличить коэффициент теплоотдачи ( $\alpha_0, \alpha_N$ ). Стержень будет отдавать больше тепла, увеличится скорость понижения температуры.

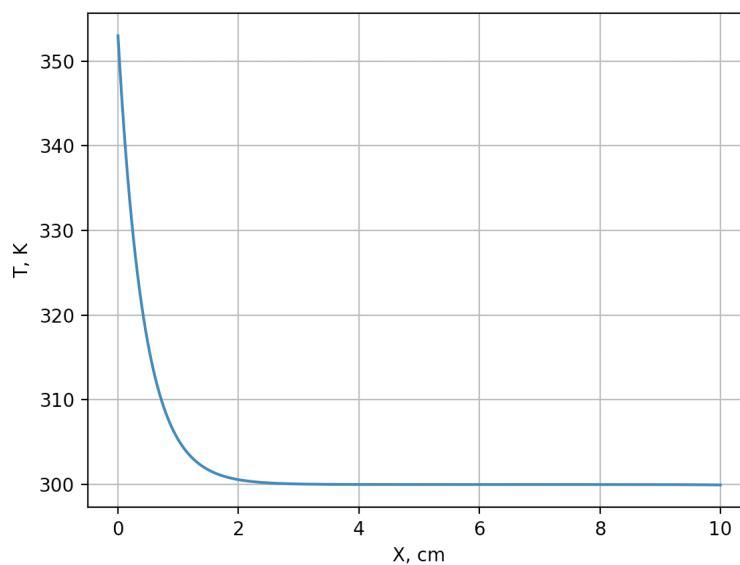


Рис. 5: График зависимости  $T(x)$  при  $\alpha_0 = 0.5, \alpha_N = 0.1$

2. Получите простейший разностный аналог нелинейного краевого условия при  $x = l$

$$-k(l) \frac{dT}{dx} = \alpha_N(T(l) - T_0) + \varphi(T),$$

где  $\varphi(T)$  заданная функция.

Построим разностную схему методом разностной аппроксимации на равномерной сетке с шагом  $h$ .

Аппроксимируем производную:

$$\frac{dT}{dx} = \frac{T(x) - T(x - h)}{h}$$

При  $x = l$ :

$$\frac{dT}{dx} = \frac{T_l - T_{l-1}}{h}$$

Подставим в исходное уравнение:

$$-k_l \frac{T_l - T_{l-1}}{h} = \alpha_N(T_l - T_0) + \varphi(T_l)$$

$$-k_l T_l + k_l T_{l-1} = \alpha_N h T_l - \alpha_N h T_0 + \varphi(T_l) h$$

$$-(k_l + \alpha_N h) T_l + k_l T_{l-1} = \varphi(T_l) h - \alpha_N h T_0 \quad (23)$$

3. Опишите алгоритм применения метода прогонки, если при  $x = 0$  краевое условие линейное (как в настоящей работе), а при  $x = l$ , как в п.2.

Для прямого хода при  $x = 0$  начальные прогоночные коэффициенты  $\varepsilon, \eta$  равны:

$$\varepsilon_1 = -\frac{M_0}{K_0}$$

$$\eta_1 = \frac{P_0}{K_0}$$

Вычислим все прогоночные коэффициенты:

$$\varepsilon_{n+1} = \frac{C_n}{B_n - A_n \varepsilon_n}$$

$$\eta_{n+1} = \frac{D_n + A_n \eta_n}{B_n - A_n \varepsilon_n}$$

Находим  $y_N$  путем решения уравнения (23), полученного в п. 2, методом дихотомии. Обратным ходом найдем все коэффициенты до  $y_0$ .

Все значения неизвестных  $y_n$  находятся по основной прогоночной формуле:

$$y_n = \varepsilon_{n+1} y_{n+1} + \eta_{n+1}$$

4. Опишите алгоритм определения единственного значения сеточной функции  $y_p$  в одной заданной точке  $p$ . Использовать встречную прогонку, т.е. комбинацию правой и левой прогонок (лекция №8). Краевые условия линейные.

Обозначим  $i = p$ ,  $0 < p < N$ .

Правая прогонка.

Область:  $0 \leq i \leq p + 1$

Прогоночные коэффициенты  $\alpha_i, \beta_i$ :

$$\alpha_{i+1} = \frac{C_i}{B_i - \alpha_i A_i}$$



$$\beta_{i+1} = \frac{A_i \beta_i + D_i}{C_i - \alpha_i A_i}$$

Левая прогонка.

Область:  $p \leq i \leq N$

Прогоночные коэффициенты  $\varepsilon_i, \eta_i$ :

$$\varepsilon_i = \frac{C_i}{B_i - \varepsilon_{i+1} A_i}$$

$$\eta_i = \frac{A_i \eta_{i+1} + D_i}{B_i - \varepsilon_{i+1} A_i}$$

Тогда при  $i = p$ :

$$y_p = \alpha_{p+1} y_{p+1} + \beta_{p+1},$$

$$y_{p+1} = \varepsilon_{p+1} y_p + \eta_{p+1},$$

$$y_p = \frac{\beta_{p+1} + \alpha_{p+1} \eta_{p+1}}{1 - \alpha_{p+1} \varepsilon_{p+1}}$$

## Вывод

В ходе лабораторной работы были получены навыки разработки алгоритмов решения краевой задачи при реализации моделей, построенных на ОДУ второго порядка.