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

ФАКУЛЬТЕТ «Информатика и системы управления»
КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 4

Дисциплина	Моделирование.
Тема	Программно-алгоритмическая реализация моделей на основе дифференциальных уравнений в частных производных с краевыми условиями II и III рода.

Студент	Сиденко А.Г.
Группа	ИУ7-63Б
Оценка (баллы)	
Преподаватель	Градов В.М.

Москва, 2020 г.

Цель работы: Получение навыков разработки алгоритмов решения смешанной краевой задачи при реализации моделей, построенных на квазилинейном уравнении параболического типа.

1. Дано:

$$k(T) = a_1(b_1 + c_1 T^{m_1})$$

$$c(T) = a_2 + b_2 T^{m_2} - \frac{c_2}{T^2}$$

$$a_1 = 0.0134$$

$$b_1 = 1$$

$$c_1 = 4.35 \cdot 10^{-4}$$

$$m_1 = 1$$

$$a_2 = 2.049$$

$$b_2 = 0.563 \cdot 10^{-3}$$

$$c_2 = 0.528 \cdot 10^{5}$$

$$m_2 = 1$$

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

$$\alpha_0 = 0.05$$

$$\alpha_N = 0.01$$

$$l = 10$$

$$T_0 = 300$$

$$R = 0.5$$

$$F(t) = 50$$

2. Уравнение

$$c(T)\frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k(T)\frac{\partial T}{\partial x}\right) - \frac{2}{R}\alpha(x)T + \frac{2T_0}{R}\alpha(x) \tag{1}$$

3. Краевые условия

$$\begin{cases} t = 0, \ T(x, 0) = T_0 \\ x = 0, \ -k(T(0))\frac{\partial T}{\partial x} = F_0 \\ x = l, \ -k(l)\frac{\partial T}{\partial x} = \alpha_N(T(l) - T_0) \end{cases}$$

Функция $\alpha(x)$ задана:

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

, где

$$c = -\alpha_0 d = \frac{\alpha_0 \alpha_N l}{\alpha_0 - \alpha_N}, d = \frac{\alpha_N l}{\alpha_N - \alpha_0}$$

4. Разностная схема

$$\begin{aligned}
&\widehat{A}_n \widehat{y}_{n-1} - \widehat{B}_n \widehat{y}_n + \widehat{D}_n \widehat{y}_{n+1} = -\widehat{F}_n, 1 \le n \le N-1 \\
&\widehat{A}_n = \widehat{\chi}_{n-\frac{1}{2}} \frac{\tau}{h}, \\
&\widehat{B}_n = \widehat{A}_n + \widehat{D}_n + \widehat{c}_n h + p_n h\tau, \\
&\widehat{D}_n = \widehat{\chi}_{n+\frac{1}{2}} \frac{\tau}{h}, \\
&\widehat{F}_n = f_n h\tau + \widehat{c}_n y_n h
\end{aligned} \tag{2}$$

5. Краевые условия

   Обозначим:

$$\begin{aligned}
F &= -k(T)\frac{\partial T}{\partial x} \\
p(x) &= \frac{2}{R}\alpha(x) \\
f(x) &= \frac{2T_0}{R}\alpha(x)
\end{aligned} \tag{3}$$

Разностные аналоги краевых условий при x=0:

$$\left(\frac{h}{8}\widehat{c}_{\frac{1}{2}} + \frac{h}{4}\widehat{c}_0 + \widehat{\chi}_{\frac{1}{2}}\frac{\tau}{h} + \frac{\tau h}{8}p_{\frac{1}{2}} + \frac{\tau h}{4}p_0\right)\widehat{y}_0 + \left(\frac{h}{8}\widehat{c}_{\frac{1}{2}} - \widehat{\chi}_{\frac{1}{2}}\frac{\tau}{h} + \frac{\tau h}{8}p_{\frac{1}{2}}\right)\widehat{y}_1 =$$
$$= \frac{h}{8}\widehat{c}_{\frac{1}{2}}(y_0 + y_1) + \frac{h}{4}\widehat{c}_0 y_0 + \widehat{F}\tau + \frac{\tau h}{4}(\widehat{f}_{\frac{1}{2}} + \widehat{c}_0) \tag{4}$$

Взято из лекции.

Разностные аналоги краевых условий при x=l, вывод:

Проинтегрируем уравнение 1 с учетом замен 3 на отрезке $[x_{N-\frac{1}{2}}, x_N]$

$$\int_{x_{N-\frac{1}{2}}}^{x_N} dx = -\int_{t_m}^{t_{m+1}} dt \int_{x_{N-\frac{1}{2}}}^{x_N} \frac{\partial F}{\partial x} dx - \int_{x_{N-\frac{1}{2}}}^{x_N} dx \int_{t_m}^{t_{m+1}} p(x) T dt + \\ + \int_{x_{N-\frac{1}{2}}}^{x_N} dx \int_{t_m}^{t_{m+1}} f(T) dt$$

Вычисляем интегралы методом правых прямоугольников.

$$\int_{x_{N-\frac{1}{2}}}^{x_N} \widehat{c}\ (\widehat{T} - T) dx = -\int_{t_m}^{t_{m+1}} (F_N - F_{N-\frac{1}{2}}) dt - \int_{x_{N-\frac{1}{2}}}^{x_N} p\ \widehat{T}\ \tau dx + \int_{x_{N-\frac{1}{2}}}^{x_N} \widehat{f}\ \tau dx$$

Вычисляем интегралы методом трапеций, а первый справа – методом правых прямоугольников.

$$\left(\widehat{c}_N\ (\widehat{y}_N - y_N) + \widehat{c}_{N-\frac{1}{2}}\ (\widehat{y}_{N-\frac{1}{2}} - y_{N-\frac{1}{2}}) \right) \frac{h}{4} = -\tau(\widehat{F}_N - \widehat{F}_{N-\frac{1}{2}}) - \\ - \left(p_N\ \widehat{y}_N + p_{N-\frac{1}{2}}\ \widehat{y}_{N-\frac{1}{2}}\right)\tau\frac{h}{4} + (\widehat{f}_N + \widehat{f}_{N-\frac{1}{2}})\tau\frac{h}{4}$$

Зная

$$\begin{aligned} &\widehat{F}_N = \alpha_N(\widehat{y}_N - T_0) \\ &\widehat{F}_{N-\frac{1}{2}} = \widehat{\chi}_{N-\frac{1}{2}}\ \frac{\widehat{y}_{N-1} - \widehat{y}_N}{h} \end{aligned} \tag{5}$$

$$\widehat{c}_N \widehat{y}_N\ \frac{h}{4} - \widehat{c}_N\ y_N \frac{h}{4} + \widehat{c}_{N-\frac{1}{2}} \widehat{y}_N\ \frac{h}{8} + \widehat{c}_{N-\frac{1}{2}} \widehat{y}_{N-1}\ \frac{h}{8} - \widehat{c}_{N-\frac{1}{2}}\ y_N \frac{h}{8} - \\ - \widehat{c}_{N-\frac{1}{2}}\ y_{N-1} \frac{h}{8} = -\tau\alpha_N\ \widehat{y}_N + \tau\alpha_N T_0 + \tau\ \widehat{\chi}_{N-\frac{1}{2}}\ \frac{\widehat{y}_{N-1}}{h} - \tau\ \widehat{\chi}_{N-\frac{1}{2}}\ \frac{\widehat{y}_N}{h} - \\ - p_N\ \widehat{y}_N\ \tau\frac{h}{4} - p_{N-\frac{1}{2}}\ \widehat{y}_N\ \tau\frac{h}{8} - p_{N-\frac{1}{2}}\ \widehat{y}_{N-1}\ \tau\frac{h}{8} + (\widehat{f}_N + \widehat{f}_{N-\frac{1}{2}})\tau\frac{h}{4}$$

$$\widehat{y}_{N-1} \left(\widehat{c}_{N-\frac{1}{2}}\ \frac{h}{8} - \frac{\tau\ \widehat{\chi}_{N-\frac{1}{2}}}{h} + p_{N-\frac{1}{2}}\tau\frac{h}{8} \right) + \\ + \widehat{y}_N \left(\widehat{c}_N\ \frac{h}{4} + \widehat{c}_{N-\frac{1}{2}}\ \frac{h}{8} + \tau\alpha_N + \frac{\tau\ \widehat{\chi}_{N-\frac{1}{2}}}{h} + p_N\tau\frac{h}{4} + p_{N-\frac{1}{2}}\tau\frac{h}{8} \right) = \\ = \widehat{c}_{N-\frac{1}{2}} \left(y_N\frac{h}{8} + y_{N-1}\frac{h}{8} \right) + \widehat{c}_N\ y_N\frac{h}{4} + \tau\alpha_N T_0 + (\widehat{f}_N + \widehat{f}_{N-\frac{1}{2}})\tau\frac{h}{4} \tag{6}$$

Соответственно из этих краевых условий (формулы 4, 6) можем найти коэффициенты $K_0,\ K_N,\ M_0,\ M_N,\ P_0,\ P_N$.

$$\begin{cases} \overset{\frown}{K}_0 \overset{\frown}{y}_0 + \overset{\frown}{M}_0 \overset{\frown}{y}_1 = \overset{\frown}{P}_0 \\ \overset{\frown}{A}_n \overset{\frown}{y}_{n-1} - \overset{\frown}{B}_n \overset{\frown}{y}_n + \overset{\frown}{D}_n \overset{\frown}{y}_{n+1} = -\overset{\frown}{F}_n, 1 \le n \le N-1 \\ \overset{\frown}{K}_N \overset{\frown}{y}_N + \overset{\frown}{M}_{N-1} \overset{\frown}{y}_{N-1} = \overset{\frown}{P}_N \end{cases} \tag{7}$$

Система 7 решается методом простых итераций.

6. Метод простых итераций

   Обозначим текущую итерацию s, а предыдущую (s-1) , тогда итерационный процесс организуется по схеме

$$A_n^{s-1} y_{n+1}^s - B_n^{s-1} y_n^s + D_n^{s-1} y_{n-1}^s = -F_n^{s-1} \tag{8}$$

   Решение которой осуществляется методом прогонки. Итерации прекращаются при условии

$$\max \left| \frac{y_n^s - y_n^{s-1}}{y_n^s} \right| \le \varepsilon, \text{для всех } n = \overline{0...N}$$

Приведем листинг.

Листинг 1: Заданные параметры

```
a1 = 0.0134
b1 = 1
c1 = 4.35e-4
m1 = 1
a2 = 2.049
b2 = 0.563e-3
c2 = 0.528e5
m2 = 1
alpha0 = 0.05
alphaN = 0.01
l = 10
T0 = 300
R = 0.5
F0 = 50
h = 0.01
t = 1
```

Листинг 2: Коэффициенты теплопроводности материала стержня и теплоотдачи при обдуве; теплоемкость

```
def k(T):
    return a1 * (b1 + c1 * T ** m1)

def c(T):
    return a2 + b2 * T ** m2 - c2 / T ** 2

def alpha(x):
    d = (alphaN * l) / (alphaN - alpha0)
    c = - alpha0 * d
    return c / (x - d)
```

Листинг 3: Выполненные замены

```
def p(x):
    return 2 * alpha(x) / R

def f(x):
    return 2 * alpha(x) * T0 / R
```

Листинг 4: Метод средних для вычисления значения функций

```
def func_plus_1_2(x, step, func):
    return (func(x) + func(x + step)) / 2

def func_minus_1_2(x, step, func):
    return (func(x) + func(x - step)) / 2
```

Листинг 5: Функции для нахождения параметров разностной схемы

```
def A(T):
    return func_minus_1_2(T, t, k) * t / h

def D(T):
    return func_plus_1_2(T, t, k) * t / h

def B(x, T):
    return A(T) + D(T) + c(T) * h + p(x) * h * t

def F(x, T):
    return f(x) * h * t + c(T) * T * h
```

Листинг 6: Метод прогонки

```
def run_through(prevT):
```

```
    K0 = h / 8 * func_plus_1_2(prevT[0], t, c) + \
                  h / 4 * c(prevT[0]) + \
                  func_plus_1_2(prevT[0], t, k) * t / h + \
                  t * h / 8 * p(h / 2) + t * h / 4 * p(0)

    M0 = h / 8 * func_plus_1_2(prevT[0], t, c) - \
                  func_plus_1_2(prevT[0], t, k) * t / h + \
                  t * h * p(h / 2) / 8

    P0 = h / 8 * func_plus_1_2(prevT[0], t, c) * (prevT[0]+ \
                  prevT[1]) + \
                  h / 4 * c(prevT[0]) * prevT[0] + \
                  F0 * t + t * h / 8 * (3 * f(0) + f(h))

    KN = h / 8 * func_minus_1_2(prevT[-1], t, c) + \
                  h / 4 * c(prevT[-1]) + \
                  func_minus_1_2(prevT[-1], t, k) * t / h + \
                  t * alphaN + \
                  t * h / 8 * p(l - h / 2) + t * h / 4 * p(l)

    MN = h / 8 * func_minus_1_2(prevT[-1], t, c) - \
                  func_minus_1_2(prevT[-1], t, k) * t / h + \
                  t * h * p(l - h / 2) / 8

    PN = h / 8 * func_minus_1_2(prevT[-1], t, c) * \
                  (prevT[-1] + prevT[-2]) + \
                  h / 4 * c(prevT[-1]) * prevT[-1] + \
                  t * alphaN * T0 + \
                  t * h / 4 * (f(l) + f(l - h / 2))

    # Прямой ход
    eps = [0, -M0 / K0]
    eta = [0, P0 / K0]

    x = h
    n = 1
    while (x + h < l):
        eps.append(D(prevT[n]) / (B(x, prevT[n]) - \
                  A(prevT[n]) * eps[n]))
        eta.append((F(x, prevT[n]) + \
                  A(prevT[n]) * eta[n]) / (B(x, prevT[n]) \
                  - A(prevT[n]) * eps[n]))
        n += 1
```

```python
        x += h

    # Обратный ход
    y = [0] * (n + 1)
    y[n] = (PN - MN * eta[n]) / (KN + MN * eps[n])

    for i in range(n - 1, -1, -1):
        y[i] = eps[i + 1] * y[i + 1] + eta[i + 1]

    return y
```

Листинг 7: Метод простых итераций

```python
def check_eps(T, newT):
    for i, j in zip(T, newT):
        if fabs((i - j) / j) > 1e-3:
            return True
    return False

def check_iter(T, newT):
    max = fabs((T[0] - newT[0]) / newT[0])
    for i, j in zip(T, newT):
        d = fabs(i - j) / j
        if d > max:
            max = d
    return max < 1e-2

def iter_method():
    result = []
    n = int(l / h)
    T = [T0] * (n + 1)
    newT = [0] * n

    result.append(T)

    while (True):
        buf = T
        while True:
            newT = run_through(buf)
            if check_iter(buf, newT):
                break
            buf = newT

        result.append(newT)
```

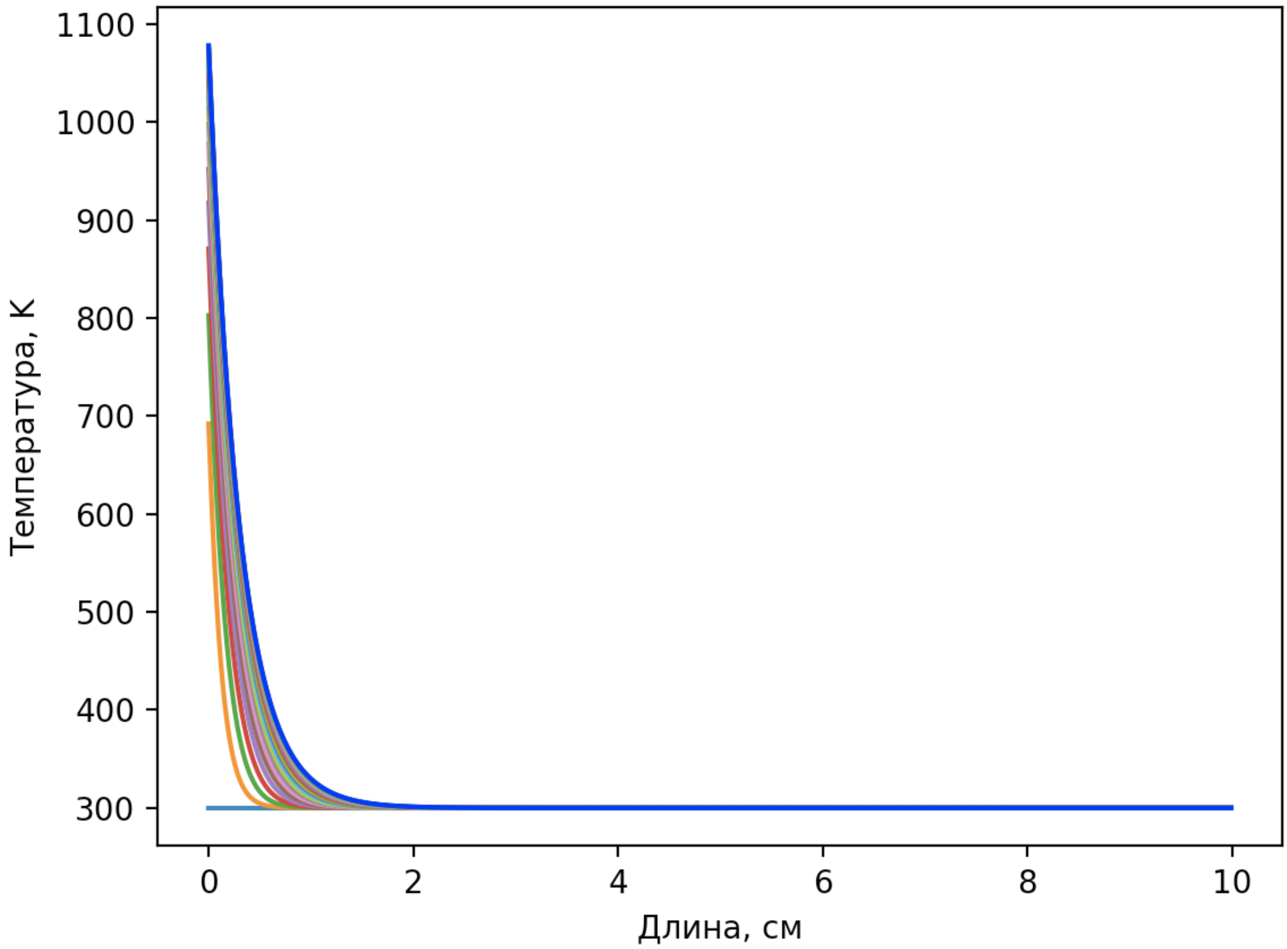
```
            if (check_eps(T, newT) == False):
                break

            T = newT

    return result
```

Результаты работы программы.

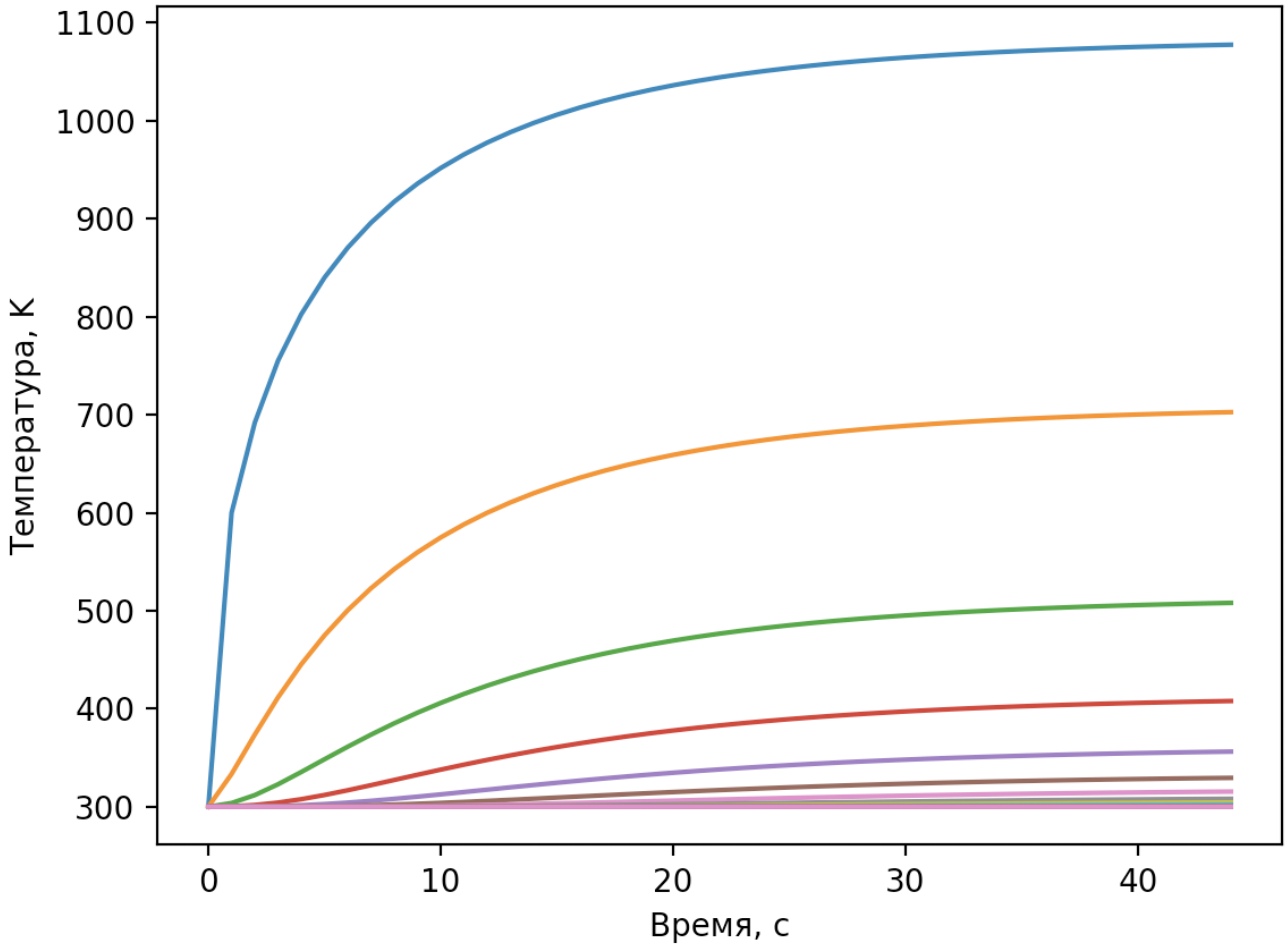
1. График зависимости температуры от координаты при нескольких фиксированных значениях времени.



   На рисунке представлены графики зависимости температуры от координаты при фиксированных t = 0, 2, 4, ...

   Последняя – синяя кривая соответствует установившемуся режиму, когда поле перестает меняться с точностью 1e-3.

2. График зависимости температуры от времени при нескольких фиксированных значениях координаты.

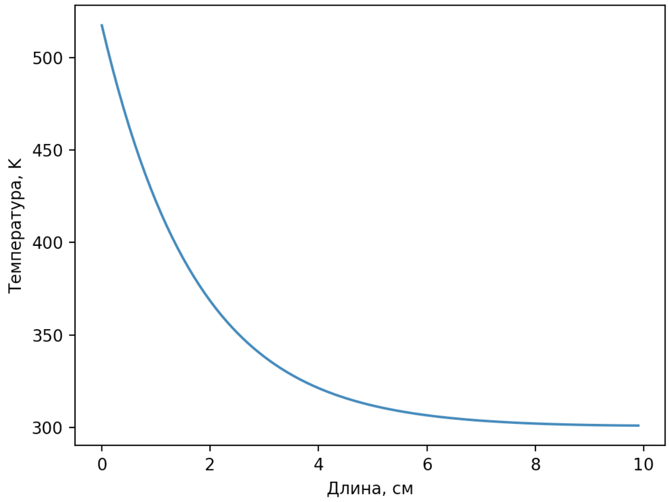


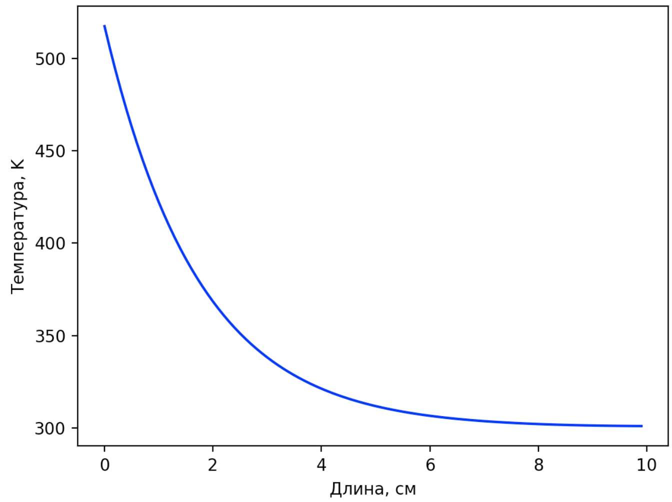
На рисунке представлены графики зависимости температуры от времени при фиксированных x = 0, 0.2, 0.4, .. 3.2

Ответы на вопросы.

1. Тестирование

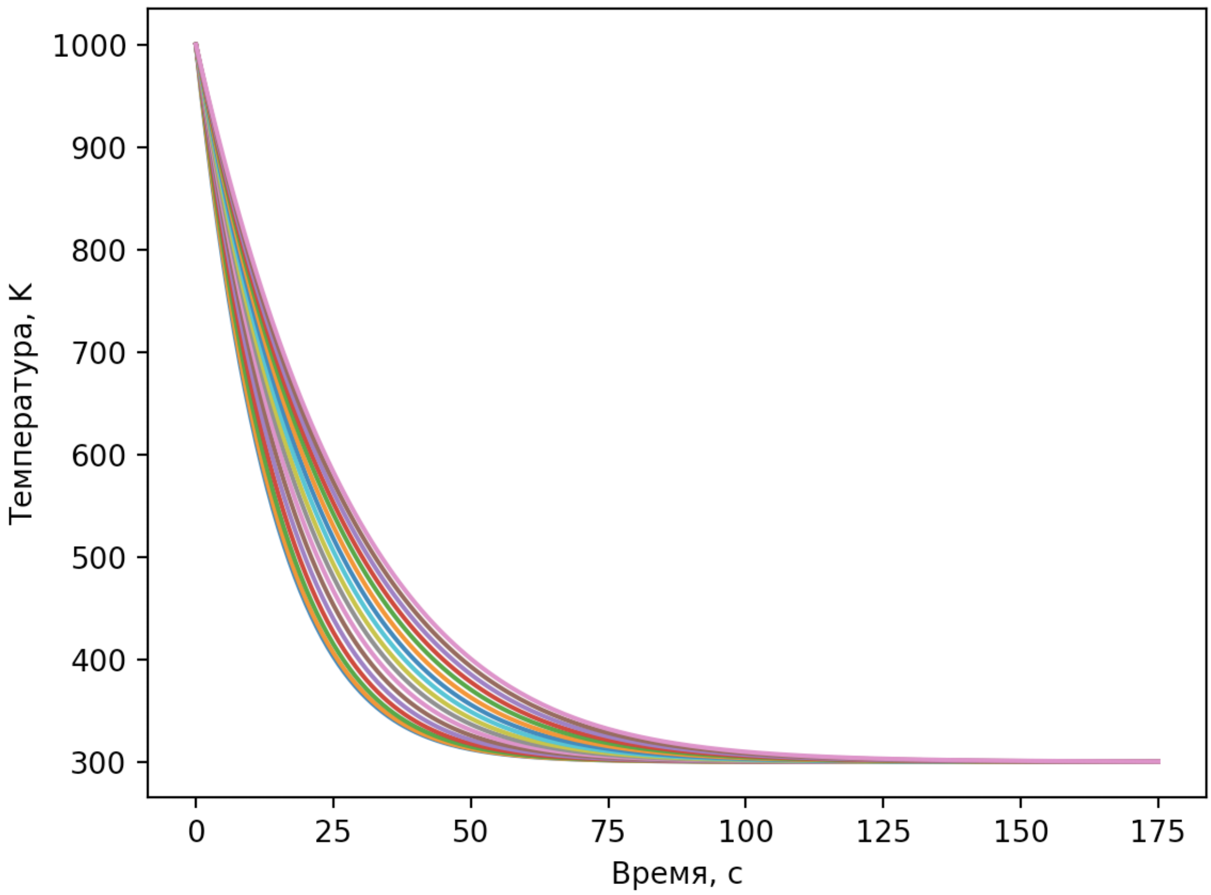
   (a) Свойства материала стержня привязаны к температуре, теплоемкость и коэффициент теплопроводности зависят от T, поменяем зависимости установим зависимости T от x, а c = 0. Тогда температурное поле T(x,t) должно совпасть с температурным распределением T(x) из лабораторной 3.





На левом рисунке представлен график из 3 работы, а на правом из текущей с измененными параметрами.

   (b) Если после разогрева стержня положить поток = 0, то будет происходить остывание, пока температура не выровняется по всей длине и не станет равной To.



Нагреем весь стержень до температуры 1000K, поток установим в 0, температура окружающей среды оставим 300K.

На данном рисунке при фиксированных x = 0, 0.2, 0.4, .. 3.2 представлены графики зависимости температуры от времени.

2. Выполните, полагая для простоты, что все коэффициенты зависят только от одной переменной y_n. Приведите линеаризованный вариант уравнения и опишите алгоритм его решения.

   Необходимо с помощью метода Ньютона линеаризовать систему 7.

   Зная, что в нашем случае коэффициенты системы зависят только от y_n.

$$\left(\overset{\frown}{A}_n \overset{\frown}{y}_{n-1} - \overset{\frown}{B}_n \overset{\frown}{y}_n + \overset{\frown}{D}_n \overset{\frown}{y}_{n+1} + \overset{\frown}{F}_n \right)\Bigg|_{(s-1)} + \overset{\frown}{A}_n^{(s-1)} \Delta \overset{\frown}{y}_{n-1}^{(s)} +$$

$$+ \left(\frac{\partial \overset{\frown}{A}_n}{\partial \overset{\frown}{y}_n} \overset{\frown}{y}_{n-1} - \frac{\partial \overset{\frown}{B}_n}{\partial \overset{\frown}{y}_n} \overset{\frown}{y}_n - \overset{\frown}{B}_n + \frac{\partial \overset{\frown}{D}_n}{\partial \overset{\frown}{y}_n} \overset{\frown}{y}_{n+1} + \frac{\partial \overset{\frown}{F}_n}{\partial \overset{\frown}{y}_n} \right)\Bigg|_{(s-1)} \cdot \Delta \overset{\frown}{y}_n^{(s)} +$$

$$+ \overset{\frown}{D}_n^{(s-1)} \Delta \overset{\frown}{y}_{n+1}^{(s)} = 0$$

   Тогда

$$A_n \Delta \overset{\frown}{y}_{n-1}^{(s)} - B_n \Delta \overset{\frown}{y}_{n}^{(s)} + D_n \Delta \overset{\frown}{y}_{n+1}^{(s)} = -F_n$$

$$\begin{aligned}
A_n &= \overset{\frown}{A}_n^{(s-1)} \\
B_n &= \left(-\frac{\partial \overset{\frown}{A}_n}{\partial \overset{\frown}{y}_n} \overset{\frown}{y}_{n-1} + \frac{\partial \overset{\frown}{B}_n}{\partial \overset{\frown}{y}_n} \overset{\frown}{y}_n + \overset{\frown}{B}_n - \frac{\partial \overset{\frown}{D}_n}{\partial \overset{\frown}{y}_n} \overset{\frown}{y}_{n+1} + \frac{\partial \overset{\frown}{F}_n}{\partial \overset{\frown}{y}_n} \right) \Bigg|_{(s-1)} \\
D_n &= \overset{\frown}{D}_n^{(s-1)} \\
F_n &= \left(\overset{\frown}{A}_n \overset{\frown}{y}_{n-1} - \overset{\frown}{B}_n \overset{\frown}{y}_n + \overset{\frown}{D}_n \overset{\frown}{y}_{n+1} + \overset{\frown}{F}_n \right) \Bigg|_{(s-1)}
\end{aligned}$$

Далее краевые условия

$$\left(\overset{\frown}{K}_0 \overset{\frown}{y}_0 + \overset{\frown}{M}_0 \overset{\frown}{y}_1 - \overset{\frown}{P}_0 \right) \Bigg|_{(s-1)} + \overset{\frown}{K}_0^{(s-1)} \Delta \overset{\frown}{y}_0^{(s)} + \overset{\frown}{M}_0^{(s-1)} \Delta \overset{\frown}{y}_1^{(s)} = 0$$

$$\left(\overset{\frown}{K}_N \overset{\frown}{y}_N + \overset{\frown}{M}_{N-1} \overset{\frown}{y}_{N-1} - \overset{\frown}{P}_N \right) \Bigg|_{(s-1)} + \overset{\frown}{K}_N^{(s-1)} \Delta \overset{\frown}{y}_N^{(s)} + \overset{\frown}{M}_{N-1}^{(s-1)} \Delta \overset{\frown}{y}_{N-1}^{(s)} = 0$$

Тогда

$$\begin{aligned}
K_0 \Delta \overset{\frown}{y}_0^{(s)} + M_0 \Delta \overset{\frown}{y}_1^{(s)} &= P_0 \\
K_N \Delta \overset{\frown}{y}_N^{(s)} + M_{N-1} \Delta \overset{\frown}{y}_{N-1}^{(s)} &= P_N
\end{aligned}$$

$$\begin{aligned}
K_0 &= \overset{\frown}{K}_0^{(s-1)} \\
M_0 &= \overset{\frown}{M}_0^{(s-1)} \\
P_0 &= -\left(\overset{\frown}{K}_0 \overset{\frown}{y}_0 + \overset{\frown}{M}_0 \overset{\frown}{y}_1 - \overset{\frown}{P}_0 \right) \Bigg|_{(s-1)} \\
K_N &= \overset{\frown}{K}_N^{(s-1)} \\
M_{N-1} &= \overset{\frown}{M}_{N-1}^{(s-1)} \\
P_N &= -\left(\overset{\frown}{K}_N \overset{\frown}{y}_N + \overset{\frown}{M}_{N-1} \overset{\frown}{y}_{N-1} - \overset{\frown}{P}_N \right) \Bigg|_{(s-1)}
\end{aligned}$$

Получаем систему

$$\begin{cases} A_n\Delta \overset{\frown}{y}{}^{(s)}_{n-1} - B_n\Delta \overset{\frown}{y}{}^{(s)}_{n} + D_n\Delta \overset{\frown}{y}{}^{(s)}_{n+1} = -F_n \\ K_0\Delta \overset{\frown}{y}{}^{(s)}_{0} + M_0\Delta \overset{\frown}{y}{}^{(s)}_{1} = P_0 \\ K_N\Delta \overset{\frown}{y}{}^{(s)}_{N} + M_{N-1}\Delta \overset{\frown}{y}{}^{(s)}_{N-1} = P_N \end{cases} \tag{9}$$

Решается методом прогонки, в результате находятся все $\Delta \overset{\frown}{y}{}^{s}_{n}$, после чего определяются значения искомой функции в узлах $\overset{\frown}{y}{}^{s}_{n} = \overset{\frown}{y}{}^{(s-1)}_{n} + \Delta \overset{\frown}{y}{}^{s}_{n}$. Итерационный процесс заканчивается при выполнении условия

$\max \left| \frac{\Delta \overset{\frown}{y}{}^{s}_{n}}{\overset{\frown}{y}{}^{s}_{n}} \right| \leq \varepsilon$.