МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО
Физико-механический институт
Высшая школа прикладной математики и вычислительной физики

Отчет по курсовому проекту
по курсу:
ЧИСЛЕННОЕ РЕШЕНИЕ КРАЕВЫХ ЗАДАЧ

на тему:

Исследование нестационарного поля температур в плоской неограниченной пластине с использованием неявного метода конечных разностей на неравномерной сетке

Работу выполнил:
студент группы 5030301/10002
Тугай В.В.

Преподаватель:
к.т.н., доц. Плетнев А.А.

Санкт-Петербург
2023

Содержание

Постановка задачи

Рассчитать стационарное поле температуры в плоской неограниченной пластине толщиной 1.5 см с равномерно распределённым источником внутреннего объемного тепловыделения Q_v=50000 Вт/м^3, и коэффициентом теплопроводности, линейно зависящим от температуры с коэффициентом линейной зависимости k_t=-0.004 1/К. На боковых поверхностях пластины ГУ 3 рода с коэффициентами теплоотдачи α_1=60 Вт/(м²·К) и α_2=20 Вт/(м²·К). Задачу решить численным методом конечных разностей на неравномерной сетке с 35 узлами и коэффициентом сгущения узлов сетки k_Δ=1.2 с использованием неявной апроксимации.

T– температура пластины, К

T_{e1}– температура на левой границе пластины, К

T_{e2}– температура на правой границе пластины, К

T_0– начальная температура пластины, К

τ – итерационный параметр

x– координата, м

L – толщина пластины, м

q– плотность теплового потока, Вт/м^2

α_1 – коэффициент конвективной теплоотдачи на левой границе пластины, Вт/(м^2×К)

α_2 – коэффициент конвективной теплоотдачи на правой границе пластины, Вт/(м^2×К)

λ – коэффициент теплопроводности, Вт/(м×К)

Im – количество узлов сетки

k_Δ – коэффициент сгущения узлов сетки

k_t – коэффициент линейной зависимости λ(T), 1/К

Q_v – источник внутреннего тепловыделения, Вт/м^3

Физические свойства материала и значения физический величин (Вариант-26):

λ_0=6.5 Вт/(м·К)

L=0.015 м

T_{e1}=253.15 К

T_{e2}=283.15 К

T_0=273.15 К

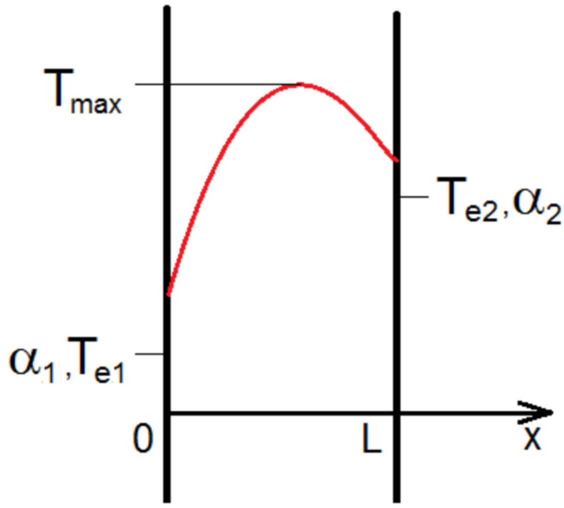

Рисунок 1
Расчетная область и стационарный профиль температуры

Одномерное стационарное уравнение теплопроводности с источником внутреннего тепловыделения:

$$0 = \frac{\partial}{\partial x}\left(\lambda \frac{\partial T}{\partial x}\right) + Q_V, \text{где } 0 \le x \le L, Q_V = Const, \lambda = \lambda_0\big(1 + k_t(T - T_0)\big).$$

k_t – коэффициент линейной зависимости $\lambda(T), 1/K$

Граничные условия:

$$\lambda \frac{\partial T}{\partial x}\bigg|_{x=0} = \alpha_1(T - T_{e1})$$

$$-\lambda \frac{\partial T}{\partial x}\bigg|_{x=L} = \alpha_2(T - T_{e2})$$

Нестационарный аналог уравнения для решения методом установления

$\frac{\partial T}{\partial \tau} = \frac{\partial}{\partial x}\left(\lambda \frac{\partial T}{\partial x}\right) + Q_V$, где τ – итерационный параметр.

Метод конечных разностей с использованием неравномерной стеки

Построение неравномерной конечно-разностной сетки

Введем обозначение $\Delta x = x_i - x_{i-1}$ - шаг сетки. Для неравномерной сетки шаг Δx_i не постоянен. Величина сгущения узлов определяется отношением двух соседних шагов сетки: $\frac{\Delta x_i}{\Delta x_{i-1}} = k_i$ коэффициент сгущения. Из теории разностных схем известно, что допустимый диапазон изменения коэффициента сгущения $0{,}5 < k_i < 2,$, т.е. уменьшение (при $k_i < 1$) или увеличение (при $k_i > 1$) не должно быть слишком большим.

Узлы сетки целесообразно сгущать там, где ожидается появление больших градиентов искомой функции, в нашем случае – около границ пластины.

Сгущение вблизи левой границы пластины предполагает, что с возрастанием индекса *i* шаг сетки увеличивается. В этом случае $1 < k_i < 2.$ Для практических целей вполне достаточно рассмотреть случай $k_i = k =$*Const.* Тогда последовательность шагов сетки образует геометрическую прогрессию, сумма всех членов которой равна (*k*, *Im* и *L* заданы)

$$\sum_{i=2}^{Im} \Delta x_i = \Delta x_2 \frac{k^{Im-1}-1}{k-1} = L \qquad (1)$$

Из (1) можно найти величину наименьшего шага сетки Δx_2. Все последующие шаги и координаты узлов могут быть найдены по рекуррентным соотношениям

$\Delta x_i = \Delta x_{i-1} k$, $i = 3 \ldots Im$; $\qquad x_i = x_{i-1} + \Delta x_i$, $i = 2 \ldots Im$.

Поскольку в нашей задаче сгущение узлов должно быть выполнено с двух сторон (вблизи левой и правой границ пластины), следует сначала построить сетку для одной половины расчетной области (при этом $Im^* = (Im+1)/2$, где *Im* обязательно нечетное), а на вторую половину перенести координаты узлов отражением от центра пластины $x = L/2$ (например, по формуле (2)).

$$\sum_{i=2}^{Im^*} \Delta x_i = \Delta x_2 \frac{k^{Im^*-1}-1}{k-1} = \frac{L}{2} \qquad (1a)$$

$\Delta x_i = \Delta x_{i-1} k$, $i = 3 \ldots Im^*$; $\qquad x_i = x_{i-1} + \Delta x_i$, $i = 2 \ldots Im^*$.

$$x_i = x_{i-1} + \Delta x_{Im+2-i},\ i = (Im^*+1) \ldots Im; \qquad (2)$$

Далее (см. рис.2) будет введено понятие границ сеточной ячейки, координаты которых обозначены xx_i и могут быть вычислены по формуле

$$xx_i = \frac{x_i + x_{i-1}}{2},\ i = 2 \ldots Im. \qquad (3)$$

Чтобы узлы x_i располагались точно по центру сеточных ячеек, необходимо после вычисления по (3) выполнить коррекцию

$$x_i = \frac{xx_{i+1}+xx_i}{2}, i = 2 \dots Im - 1. \qquad (4)$$

Крайние узлы корректировать не нужно, так как их координаты по определению совпадают с границами области: $x_1 = 0$; $x_{Im} = L$

Вывод сеточных уравнений

Для построения разностной схемы введем в рассмотрение понятие элементарной сеточной ячейки, границы которой обозначены xx_i (рис. 2). При этом узлы сетки (за исключением граничных) должны быть расположены в центре сеточных ячеек.

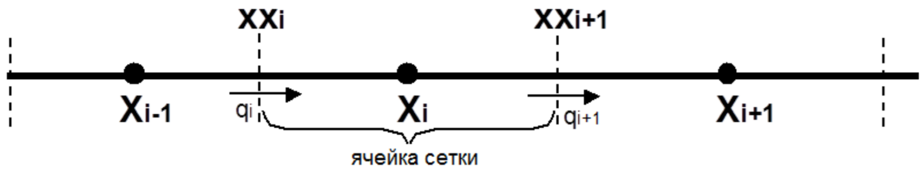

Рисунок 2
Шаблон аппроксимации уравнения теплопроводности

Считаем, что в пределах одной сеточной ячейки коэффициент λ постоянен: $\lambda_i = f(T_i)$. Тогда тепловой поток на границе сеточной ячейки (см. рис. 2) может быть вычислен по формуле

$$q_i = -\frac{T_i - T_{i-1}}{\frac{x_i - xx_i}{\lambda_i} + \frac{xx_i - x_{i-1}}{\lambda_{i-1}}} \qquad (5)$$

С учетом (5) неявную разностную аппроксимацию уравнения для внутренних узлов (i = 2 ... Im-1) запишем в виде

$$\frac{T_i^{n+1} - T_i^n}{\Delta\tau} = \frac{1}{xx_{i+1} - xx_i}\left(\frac{T_{i+1}^{n+1} - T_i^{n+1}}{\frac{x_{i+1} - xx_{i+1}}{\lambda_{i+1}} + \frac{xx_{i+1} - x_i}{\lambda_i}} - \frac{T_i^{n+1} - T_{i-1}^{n+1}}{\frac{x_i - xx_i}{\lambda_i} + \frac{xx_i - x_{i-1}}{\lambda_{i-1}}}\right) + Q_V \qquad (6)$$

Для крайних (граничных) ячеек получим

$$\frac{T_1^{n+1} - T_1^n}{\Delta\tau} = \frac{1}{xx_2 - xx_1}\left(\frac{T_2^{n+1} - T_1^{n+1}}{\frac{x_2 - xx_2}{\lambda_2} + \frac{xx_2 - x_1}{\lambda_1}} - \alpha_1(T_1^{n+1} - T_{e1})\right) + Q_V \qquad (7)$$

$$\frac{T_{Im}^{n+1} - T_{Im}^n}{\Delta\tau} = \frac{1}{x_{Im} - xx_{Im}}\left(-\frac{T_{Im}^{n+1} - T_{Im-1}^{n+1}}{\frac{x_{Im} - xx_{Im}}{\lambda_{Im}} + \frac{xx_{Im} - x_{Im-1}}{\lambda_{Im-1}}} - \alpha_2(T_{Im}^{n+1} - T_{e2})\right) + Q_V \qquad (12)$$

В выражениях (10) - (12) предполагается, что значения λ_i вычисляются на текущем временном слое, т.е. с «запаздыванием»: $\lambda_i = f(T_i^n)$

Решение системы линейных алгебраических уравнений (СЛАУ)

Разностное уравнение (10) представим в виде

$$A_i T_{i+1}^{n+1} + B_i T_i^{n+1} + C_i T_{i-1}^{n+1} + D_i = 0\,, \tag{13}$$

где T_i^{n+1} искомая сеточная функция, A_i, B_i, C_i, D_i – прогоночные коэффициенты.

Как видно из (13), коэффициенты A_i, B_i и C_i представляют собой выражения, на которые умножаются соответствующие неизвестные значения T в разностном уравнении (10). Коэффициент D_i объединяет слагаемые разностного уравнения, которые не вошли ни в один из коэффициентов A_i, B_i, C_i.

Перенесем все слагаемые (10) в правую часть уравнения. Тогда получим:

$$D_i = \frac{T_n^n}{\Delta\tau} + Q_V.$$

Для крайних узловых точек некоторые из прогоночных коэффициентов не определены. Поэтому постулируем $C_i = A_{Im} = 0$.

Когда все четыре массива $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$ сформированы, для нахождения массива $\boldsymbol{T^{n+1}}$ следует выполнить обращение к процедуре векторной прогонки.

Вычисление теплового баланса

При достижении стационарного состояния количество теплоты, выделяемое внутри пластины, должно быть равно количеству теплоты, отводимой в окружающую среду от ее поверхности. Поскольку задача решается методом установления, то указанное равенство достигается только в пределе при $\tau \to \infty$. Поэтому условием достижения теплового баланса (и одновременно выхода из итеративного цикла) является:

$$abs\left(1 - \frac{\alpha_2\left(T_{Im}^{n+1} - T_{e2}\right) + \alpha_1\left(T_1^{n+1} - T_{e1}\right)}{Q_V L}\right) \le \varepsilon\,, \tag{14}$$

где ε – заданная точность.

Этапы программной реализации

1) Задание исходных данных

2) Вычисление координат узлов сетки (массив x_i) и координат границ сеточных

ячеек (массив xx_i). Коррекция массива x_i.

3) Вычисление итерационного параметра $\Delta\tau$.

4) Присвоение начальных значений массиву $T_i^1 = T_0$.

5) Начало итеративного цикла.

6) Вычисление значений $\lambda_i = f(T_i^n)$ в узлах сетки.

7) Вычисление прогоночных коэффициентов (заполнение массивов A, B, C, D для всех i = 1 ... Im).

8) Обращение к процедуре прогонки (вычисление элементов массива T_i^{n+1}).

9) Проверка достижения теплового баланса (условие выхода из цикла).

10) Переприсвоение массивов ($T_i^n = T_i^{n+1}$).

11) Возврат к п.6

Результаты решения

Программа main.f95, написанная на Fortran 95, состоит из основного тела программы и модуля methods, содержащий в себе 3 подпрограммы : подпрограмма sweep отвечает за Решение СЛАУ для нахождения массива T с помощью прогоночных коэффициентов; подпрограмма grid для построения неравномерной сетки, сгущающейся у стенок пластины; подпрограмма coeffs, использующие предыдущие две подпрограммы при работе для определения значений профиля температуры и отвечающая за расчет данных, представленных в этом отчете (см. Приложение (1)).

Рисунок с расположением узлов сетки (x_i) и границ сеточных ячеек (xx_i) вблизи стенки (примерно 10 узлов)

На рисунке 3 изображено расположение узлов сетки и границ сеточных ячеек вблизи правой стенки. Как видно из рисунка, сетка действительно является неравномерной и сгущается в направлении к стенкам.

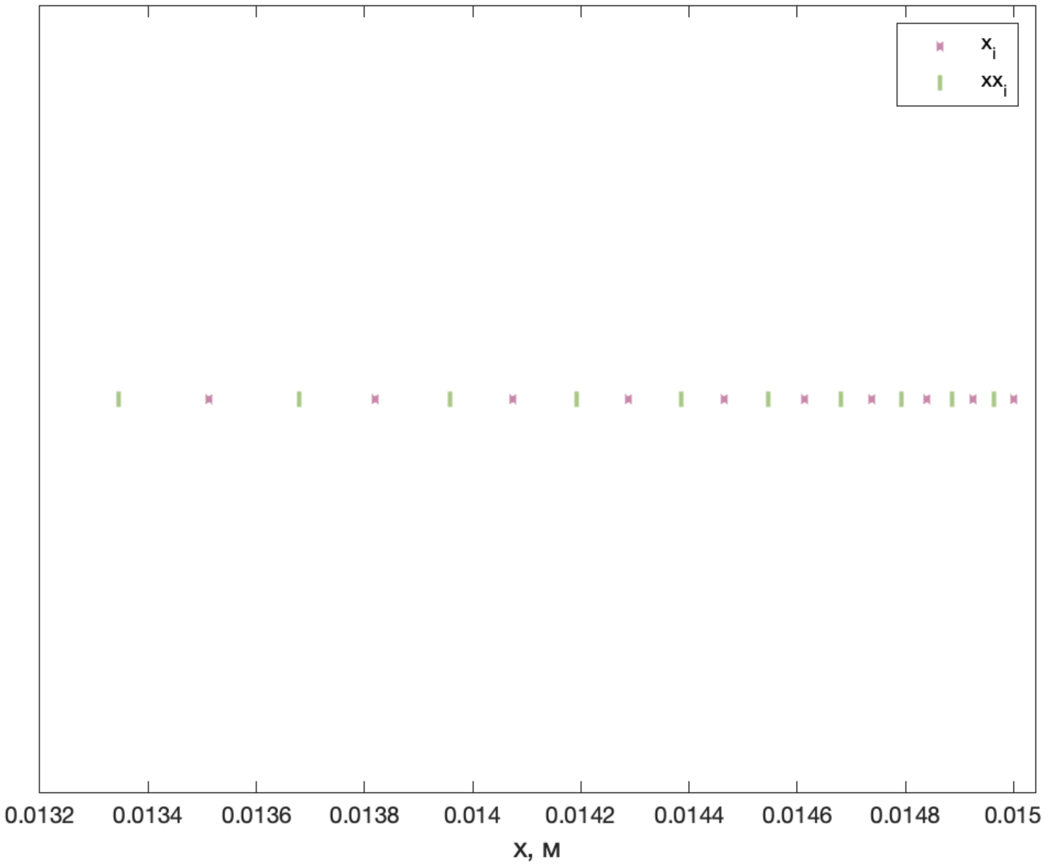

Рисунок 3

Расположение узлов сетки (x_i) и границ сеточных ячеек (xx_i) вблизи правой стенки (10 узлов)

Стационарный профиль температуры

На рисунке 4 изображен стационарный профиль температуры в Кельвинах. Как видно из рисунка, профиль температуры имеет горизонтальную асимптоту – значения температуры от левой стенки до центра пластины линейно возрастает, и далее от центра пластины до правой стенки производная графика уменьшается и значения температуры увеличиваются, стремясь к максимальному значению. Значения на правой и левой стенках T_{left}=269.626 K; T_{right}=271.224 K (см. Приложение (2)), соответственно, и максимальное значение температуры достигается на правой стенке T_{max}=271.224 K (см. Приложение (2)).

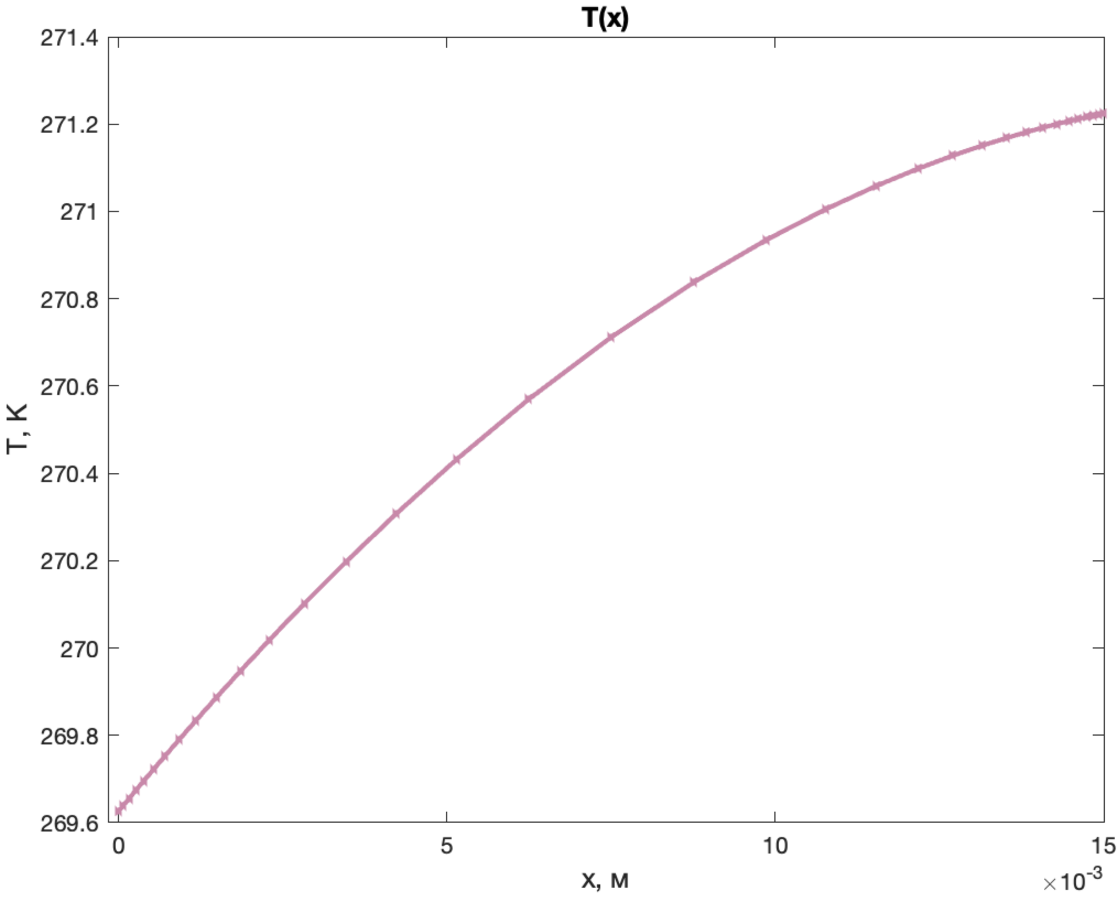

Рисунок 4
Стационарный профиль температуры

Распределение коэффициента λ в сечении пластины

На рисунке 5 изображено распределение коэффициента теплопроводности в сечении пластины. Сравнивая рисунки 4, 5 можно подметить, что рисунок 5 является отражением по вертикали рисунка 4.

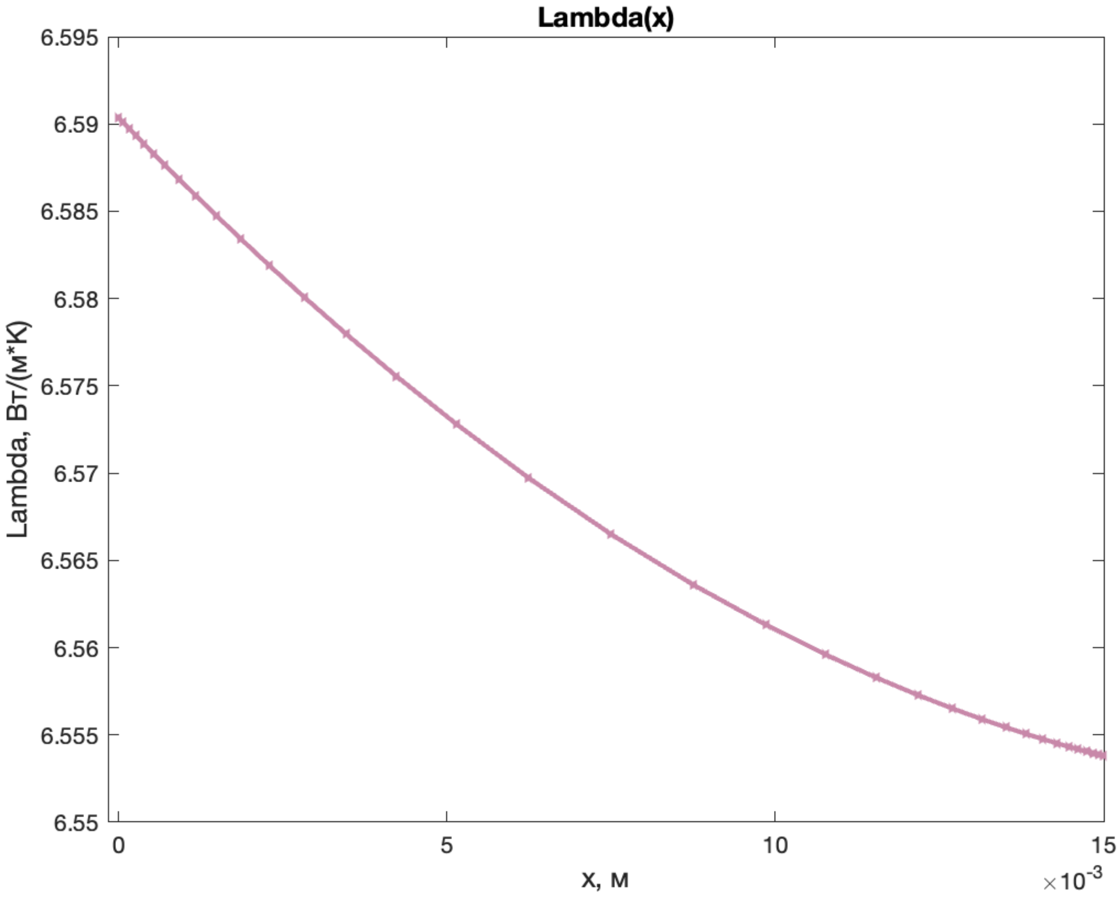

Рисунок 5
Распределение коэффициента λ в сечении пластины

Тепловые потоки на границах и тепловой баланс

Тепловой поток на левой границе q_{left}=991.45 Вт/м^2, на правой q_{right}=- 241.38 Вт/м^2, тепловой баланс Ba=$\frac{q_{\text{left}}+q_{\text{right}}}{Q_V L}$ =1.0001 .

Вывод

В ходе решения поставленной задачи был изучен метод конечных разностей на неравномерной сетке. Решение задачи данным методом было реализовано на языке программирования Fortran (см. Приложение 1), результаты программы представлены в виде графиков с использованием пакета Matlab.

По итогам работы программы были посчитаны суммарное тепловыделение в сечении пластины Q_{sum}=750 Вт/м^2, суммарный тепловой поток на границах q_{sum}=750.08 Вт/м^2, и точность решения $\Delta_{balance}$= Ba-1=0.0001 (см. Приложение (2)).

Приложение 1

Основная программа

```
module methods
contains

subroutine sweep(Im,A,B,C,D,F)
implicit none
integer, intent(in):: Im
real*16, dimension(1:Im), intent(in):: A, B, C, D
real*16, dimension(1:Im), intent(out):: F
real*16, dimension(1:Im):: alpha, beta
real*16 :: k0
integer::i
!Прямой ход
alpha(1) = -A(1) / B(1)
beta(1) = -D(1) / B(1)
do i = 2, (Im-1)
    k0 = (B(i) + C(i)*alpha(i-1))
    alpha(i) = -A(i) / k0
    beta(i) = -(D(i) + C(i)*beta(i-1)) / k0
end do
!Обратный ход
F(Im) = -(D(Im) + C(Im)*beta(Im-1)) / (B(Im) + C(Im)*alpha(Im-1))
do i = (Im-1), 1, -1
    F(i) = alpha(i)*F(i+1) + beta(i)
end do

end subroutine


subroutine grid(L, Im, koeff_d, lambda_0, x, xx, delt_tau)
implicit none
real*16 :: L, koeff_d, lambda_0, alpha_1, alpha_2, T0, Te1, Te2, po, cp, delt_tau
real*16, dimension (Im) :: delta_x, x, xx, lambda
integer :: Im, i, n, Im_star

Im_star=(Im+1.0)/2.0
x(1)=0.0
x(Im)=L
delta_x(2)=(L*(koeff_d-1.0))/(2.0*((koeff_d**(Im_star-1.0))-1.0))
do i=2,Im_star
    if (i/=2) delta_x(i)=delta_x(i-1)*koeff_d
    x(i)=x(i-1)+delta_x(i)
end do
do i=Im_star+1,Im
    x(i)=x(i-1)+delta_x(Im+2-i)
end do
```

```fortran
do i=2,Im
    xx(i)=(x(i)+x(i-1))/2.0
end do
do i=2,Im-1
    x(i)=(xx(i+1)+xx(i))/2.0
end do
delt_tau=(L**2)/(2*lambda_0*((Im-1)**2))

end subroutine


subroutine coeffs(L, Im, koeff_d, Qv, lambda_0, koeff_t, alpha_1, alpha_2, T0, Te1, Te2,
po, cp, eps)
implicit none
real*16 :: L, koeff_d, Qv, lambda_0, koeff_t, alpha_1, alpha_2, T0, Te1, Te2, po, cp
real*16 :: eps, delt_tau, balance
real*16, dimension (Im) :: x, xx, lambda, T, q,  AA, BB, CC, DD
integer :: Im, i, n, Im_star, check=0

call grid(L, Im, koeff_d, lambda_0, x, xx, delt_tau)

T=T0; AA=0; CC=0

do n=1,200000

    lambda=lambda_0*(1.0+koeff_t*(T-T0))
    do i=2,Im-1
        AA(i)=(lambda(i+1)*lambda(i))/((xx(i+1)-xx(i))*(lambda(i)*(x(i+1)-
xx(i+1))+lambda(i+1)*(xx(i+1)-x(i))))
        CC(i)=(lambda(i)*lambda(i-1))/((xx(i+1)-xx(i))*(lambda(i-1)*(x(i)-
xx(i))+lambda(i)*(xx(i)-x(i-1))))
        BB(i)=-AA(i)-CC(i)-(1.0/delt_tau)
        DD(i)=(T(i)/delt_tau)+Qv
    end do
    AA(1)=(lambda(2)*lambda(1))/((xx(2)-x(1))*(lambda(1)*(x(2)-xx(2))+lambda(2)*(xx(2)-
x(1))))
    BB(1)=-AA(1)-(1.0/delt_tau)-(alpha_1/(xx(2)-x(1)))
    DD(1)=(T(1)/delt_tau)+((alpha_1*Te1)/(xx(2)-x(1)))+Qv
    CC(Im)=(lambda(Im)*lambda(Im-1))/((x(Im)-xx(Im))*(lambda(Im-1)*(x(Im)-
xx(Im))+lambda(Im)*(xx(Im)-x(Im-1))))
    BB(Im)=-CC(Im)-(1.0/delt_tau)-(alpha_2/(x(Im)-xx(Im)))
    DD(Im)=(T(Im)/delt_tau)+((alpha_2*Te2)/(x(Im)-xx(Im)))+Qv

    call sweep(Im,AA,BB,CC,DD,T)

    balance=abs(1-(((alpha_2*(T(Im)-Te2))+alpha_1*(T(1)-Te1))/(Qv*L)))
    if (balance<=eps) exit

end do

write(16,*) T+273.15; write(17,*) lambda; write(14,*) x; write(15,*) xx
write(*,'(a12,$)') '  6) max T: ' ; write(*,'(f7.3,$)') maxval(T)+273.15;
write(*,'(a19,$)') ' K ,  left wall T: ';
```

```
write(*,'(f7.3,$)') T(1)+273.15; write(*,'(a20,$)') ' K ,  right wall T: ' ;
write(*,'(f7.3,$)') T(Im)+273.15; write(*,*) 'K'
write(*,'(a23,$)') '  8) heat dissipation: ' ; write(*,'(f8.3,$)') Qv*L ; write(*,*)
'W/m^2'
write(*,'(a16,$)') '  9) heat flow: ' ; write(*,'(f8.3,$)') (alpha_2*(T(Im)-
Te2))+(alpha_1*(T(1)-Te1)); write(*,*) 'W/m^2'
write(*,'(a20,$)') ' 10) balance error: '; write(*,'(f14.12)') balance
write(*,'(a17,$)') ' left heat flow: '; write(*,'(f8.3,$)') alpha_1*(T(1)-Te1)
write(*,'(a20,$)') ' ,  right heat flow: '; write(*,'(f8.3,$)') alpha_2*(T(Im)-Te2)
write(*,'(a19,$)') ' ,  term balance: '; write(*,'(f14.12)') ((alpha_2*(T(Im)-
Te2)+alpha_1*(T(1)-Te1))/(Qv*L))


end subroutine

end module


program main
use methods
implicit none
real*16 :: L, koeff_d, Qv, lambda_0, koeff_t, alpha_1, alpha_2, T0, Te1, Te2, po, cp,
eps
integer :: Im

open(201, file='bin/data.txt')
read(201,*) L; read(201,*) Im; read(201,*) koeff_d; read(201,*) Qv; read(201,*)
lambda_0; read(201,*) koeff_t; read(201,*) alpha_1;
read(201,*) alpha_2; read(201,*) T0; read(201,*) Te1; read(201,*) Te2; read(201,*) po;
read(201,*) cp; read(201,*) eps
close(201)

open(14, file='bin/results/x_i.txt'); open(15, file='bin/results/xx_i.txt')
open(16, file='bin/results/T.txt'); open(17, file='bin/results/lambda_i.txt')

write(*,*)'start program'; write(*,*); write(*,'(a4,$)') ' Im='; write(*,'(i3)') Im;
write(*,*)

call coeffs(L, Im, koeff_d, Qv, lambda_0, koeff_t, alpha_1, alpha_2, T0, Te1, Te2, po,
cp, eps)

close(14); close(15); close(16); close(17)

write(*,*); write(*,*) 'end program'

end program
```

Приложение 2

Вывод основной программы

```
start program
```

```
Im= 35

 6) max T: 271.081 K ,  left wall T: 269.674 K ,  right wall T: 271.081 K
 8) heat dissipation:  750.000 W/m^2
 9) heat flow:  750.075 W/m^2
10) balance error: 0.000099995654
left heat flow:  991.453 ,  right heat flow:-241.378  ,  term balance: 1.000099995654

end program
```

Приложение 3

Текстовый файл с исходными данными

```
0.015   – L
35   – Im
1.2   – kΔ
50000   – Qv
6.5   – lamda_0
-0.004   – kt
60  – alpha_1
20   – alpha_2
0   – T_0
-20   – T_e1
10   – T_e2
8600   – po
380   – c
0.01      – eps
```