	Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)» (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: Гасанзаде М.А.

Группа ИУ7-66Б

Оценка (баллы) _______________

Преподаватель : Градов В.М.

Москва.
2020 г.

СОДЕРЖАНИЕ

I. АНАЛИТИЧЕСКАЯ ЧАСТЬ.

Цель работы

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

Исходные данные

1. Задана математическая модель.

   Уравнение для функции T(x,t)

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

2. Краевые условия: $\begin{cases} t=0, & T(x,0)=T_0, \\ x=0, & -k(T(0))\dfrac{\partial T}{\partial x}=F_0, \\ x=l, & -k(T(l))\dfrac{\partial T}{\partial x}=\alpha_N(T(l)-T_0) \end{cases}$

3. Разностная схема с разностным краевым условием при x=0

$$\widehat{A_n}\widehat{y_{n-1}}-\widehat{B_n}\widehat{y_n}+\widehat{D_n}\widehat{y_{n+1}}=-\widehat{F_n} \tag{2}$$

$$\begin{gathered}\left(\frac{h}{8}\widehat{c_{1/2}}+\frac{h}{4}\widehat{c_0}+\widehat{\chi_{1/2}}\frac{\tau}{h}+\frac{\tau h}{8}p_{1/2}+\frac{\tau h}{4}p_0\right)\widehat{y_0}+\left(\frac{h}{8}\widehat{c_{1/2}}-\widehat{\chi_{1/2}}\frac{\tau}{h}+\frac{\tau h}{8}p_{1/2}\right)\widehat{y_1}= \\ =\frac{h}{8}\widehat{c_{1/2}}(y_0+y_1)+\frac{h}{4}\widehat{c_0}y_0+\widehat{F}\tau+\frac{\tau h}{4}(\widehat{f_{1/2}}+\widehat{f_0})\end{gathered} \tag{3}$$

Разностный аналог краевого условия при $x=l$ интегро-интерполяционным методом, интегрируя на отрезке $[x_{N-1/2}, x_N]$ уравнение (1), учитывая, что поток $\widehat{F_N}=\alpha_N(\widehat{y_N}-T_0)$,а $\widehat{F_{N-1/2}}=\widehat{\chi_{N-1/2}}\dfrac{\widehat{y_{N-1}}-\widehat{y_N}}{h}$:

$$\int\limits_{x_{N-1/2}}^{x_N}dx\int\limits_{t_m}^{t_{m+1}}c(T)\frac{\partial T}{\partial t}dt=-\int\limits_{t_m}^{t_{m+1}}dt\int\limits_{x_{N-1/2}}^{x_N}\frac{\partial F}{\partial x}dx-\int\limits_{x_{N-1/2}}^{x_N}dx\int\limits_{t_m}^{t_{m+1}}p(x)Tdt+\int\limits_{x_{N-1/2}}^{x_N}dx\int\limits_{t_m}^{t_{m+1}}f(x)dt \quad \mathbf{(4)}$$

Методом правых прямоугольников для интегралов справа:

$$\int\limits_{x_{N-1/2}}^{x_N} \hat{c}\,(\hat{T}-T)\,dx = -\int\limits_{t_m}^{t_{m+1}} (F_N - F_{N-1/2})\,dt - \int\limits_{x_{N-1/2}}^{x_N} p\,\hat{T}\,\tau\,dx + \int\limits_{x_{N-1/2}}^{x_N} \hat{f}\,\tau\,dx \quad \textbf{(5)}$$

Первый интеграл справа вычислим применив метод правых прямоугольников, а последующие методом трапеций **(6)**:

$$\frac{h}{4}\left[\widehat{c_N}(\widehat{y_N}-y_N)+\widehat{c_{N-1/2}}(\widehat{y_{N-1/2}}-y_{N-1/2})\right] = -\tau(\widehat{F_N}-\widehat{F_{N-1/2}}) - \frac{h}{4}\tau(p_N\widehat{y_N}+p_{N-1/2}\widehat{y_{N-1/2}}) +$$
$$+\frac{h}{4}\tau(\widehat{f_N}+\widehat{f_{N-1/2}})$$

Подставим в выражения для потока **(7)**:

$$\frac{h}{4}\left[\widehat{c_N}(\widehat{y_N}-y_N)+\widehat{c_{N-1/2}}\left(\frac{\widehat{y_N}+\widehat{y_{N-1}}}{2}-\frac{y_N+y_{N-1}}{2}\right)\right] = -\tau\left(\alpha_N(\widehat{y_N}-T_0)-\widehat{\chi_{N-1/2}}\frac{\widehat{y_{N-1}}-\widehat{y_N}}{h}\right)-$$
$$-\frac{h}{4}\tau\left(p_N\widehat{y_N}+p_{N-1/2}\frac{\widehat{y_N}+\widehat{y_{N-1}}}{2}\right)+\frac{h}{4}\tau\left(\widehat{f_N}+f_{N-1/2}\right)$$

Приведя к общему виду, получаем **(8)**:

$$\left(\frac{h}{4}\widehat{c_N}+\frac{h}{8}\widehat{c_{N-1/2}}+\alpha_N\tau+\frac{\tau}{h}\chi_{N-1/2}+\frac{h}{4}\tau\,p_N+\frac{h}{8}\tau\,p_{N-1/2}\right)y_N+\left(\frac{h}{8}\widehat{c_{N-1/2}}-\frac{\tau}{h}\chi_{N-1/2}+\frac{h}{8}\tau\,p_{N-1/2}\right)\cdot$$
$$\cdot y_{N-1}=\frac{h}{4}\widehat{c_N}\,y_N+\frac{h}{8}\widehat{c_{N-1/2}}(y_N+y_{N-1})+\tau\,\alpha_N T_0+\frac{h}{4}\tau\left(\widehat{f_N}+\widehat{f_{N-1/2}}\right)$$

Принимая простую аппроксимацию:

$$p_{N-1/2}=\frac{p_N+p_{N-1}}{2},\qquad \widehat{f_{N-1/2}}=\frac{\widehat{f_N}+\widehat{f_{N-1}}}{2},\qquad \widehat{c_{N-1/2}}=\frac{\widehat{c_N}+\widehat{c_{N-1}}}{2}$$

Как мы видим, если принять c(u) = 0 и сократить τ, формула (8) перейдёт в формулу для разностного краевого условия при $x=l$ из предыдущей лабораторной работы.

Физический смысл задачи

1. Сформулированная в данной работе математическая модель описывает **нестационарное** температурное поле *T(x,t)*, зависящее от координаты *x* и меняющееся во времени.

2. Свойства материала стержня привязаны к температуре, т.е. теплоемкость и коэффициент теплопроводности $c(T)$, $k(T)$ зависят от T , тогда как в работе №3 $k(x)$ зависит от координаты, а $c = 0$.

3. При $x = 0$ цилиндр нагружается тепловым потоком $F(t)$, в общем случае зависящим от времени, а в работе №3 поток был постоянный.

Если в настоящей работе задать поток постоянным, т.е. $F(t) = const$, то будет происходить формирование температурного поля от начальной температуры T_0 до некоторого установившегося (стационарного) распределения $T(x,t)$. Это поле в дальнейшем с течением времени меняться не будет и должно совпасть с температурным распределением $T(x)$, получаемым в лаб. работе №3, если все параметры задач совпадают, в частности, вместо $k(T)$ надо использовать $k(x)$ из лаб. работы №3. Это полезный факт для тестирования программы.

Если после разогрева стержня положить поток $F(t) = 0$, то будет происходить остывание, пока температура не выровняется по всей длине и не станет равной T_0 .

При произвольной зависимости потока $F(t)$ от времени температурное поле будет как-то сложным образом отслеживать поток.

Замечание. Варьируя параметры задачи, следует обращать внимание на то, что решения, в которых температура превышает примерно 2000К, физического смысла не имеют и практического интереса не представляют.

II. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

ЯП был выбран `Python3` из-за простоты работы с графиками и библиотеки `matplotlib`. Ниже на листингах будет представлена реализация программы:

Листинг

Листинг 1. Метод прогонки

```
void Sweep::setLeftBoundary(double K0, double M0, double P0)
{
    _K0 = K0;
    _M0 = M0;
    _P0 = P0;
}

void Sweep::setRightBoundary(double KN, double MN, double PN)
{
    _KN = KN;
    _MN = MN;
    _PN = PN;
}

void Sweep::setCoeffs(const QVector<double> &A, const QVector<double> &B, const
QVector<double> &C, const QVector<double> &F)
{
    _A = A;
    _B = B;
    _C = C;
    _F = F;
}

QVector<double> Sweep::solve()
{
    QVector<double> xi;
    QVector<double> eta;
    int N = _A.size();
    QVector<double> y(N);
    xi.push_back(0);
    eta.push_back(0);
    xi.push_back(-_M0 / _K0);
    eta.push_back(_P0 / _K0);
    for (int i = 1; i < N - 1; i++)
    {
        double div = _B[i] - _A[i] * xi[i];
        xi.push_back(_C[i] / div);
        eta.push_back((_F[i] + _A[i] * eta[i]) / div);
    }
    y[N-1] = (_PN - _MN * eta[N-1]) / (_KN + _MN * xi[N-1]);
    for (int i = N - 2; i >= 0; i--)
    {
        y[i] = xi[i+1] * y[i+1] + eta[i+1];
    }
    return y;
}
```

Листинг 2. Вычисление коэффициентов разностой схемы.

```
void Rod::calculateCoeffs(QVector<double> &A, QVector<double> &B,
QVector<double> &D, QVector<double> &F)
{
    int N = static_cast<int>(floor(_l / _h) + 1);
    A = QVector<double>(N);
    B = QVector<double>(N);
    D = QVector<double>(N);
    F = QVector<double>(N);
    double x = _h;
    for (int i = 1; i < N - 1; i++)
    {
        A[i] = _tau * calculateChi(_prevIteration[i], _prevIteration[i + 1]) /
_h;
        D[i] = _tau * calculateChi(_prevIteration[i], _prevIteration[i - 1]) /
_h;
        B[i] = A[i] + D[i] + calculateC(_prevIteration[i]) * _h + calculateP(x)
* _h * _tau;
        F[i] = calculateF(x) * _h * _tau + calculateC(_prevIteration[i]) *
_prevT[i] * _h;
        x += _h;
    }
}
```

Листинг 3. Вычисление краевых условий (правого)

```
    double chiHalf = calculateChi(_prevIteration.back(),
_prevIteration[_prevIteration.size() - 2]);
    double pN = calculateP(_l);
    double fN = calculateF(_l);
    double cN = calculateC(_prevIteration.back());
    double pN1 = calculateP(_l - _h);
    double fN1 = calculateF(_l - _h);
    double cN1 = calculateC(_prevIteration[_prevIteration.size() - 2]);
    double pHalf = (pN + pN1) / 2;
    double fHalf = (fN + fN1) / 2;
    double cHalf = (cN + cN1) / 2;
    double h8 = _h / 8;
    double h4 = h8 * 2;

    KN = h4 * cN + h8 * cHalf + _tau * _alphaN + _tau / _h * chiHalf + h4 * _tau
* pN + h8 * _tau * pHalf;
    MN = h8 * cHalf - _tau / _h * chiHalf + h8 * _tau * pHalf;
    PN = h4 * cN * _prevT.back() + h8 * cHalf * (_prevT.back() +
_prevT[_prevT.size() - 2]) + _tau * _alphaN * _T0 + h4 * _tau * (fN + fHalf);
}
```

Листинг 4. Вычисление краевых условий (левого)

```
    double chiHalf = calculateChi(_prevIteration[0], _prevIteration[1]);
    double p0 = calculateP(0);
    double f0 = calculateF(0);
    double c0 = calculateC(_prevIteration[0]);
    double p1 = calculateP(_h);
    double f1 = calculateF(_h);
    double c1 = calculateC(_prevIteration[1]);
    double pHalf = (p0 + p1) / 2;
    double fHalf = (f0 + f1) / 2;
```

```
    double cHalf = (c0 + c1) / 2;
    double h8 = _h / 8;
    double h4 = h8 * 2;
    K0 = h8 * cHalf + h4 * c0 + chiHalf * _tau / _h + _tau * h8 * pHalf + _tau *
h4 * p0;
    M0 = h8 * cHalf - chiHalf * _tau / _h + _tau * h8 * pHalf;
    P0 = h8 * cHalf * (_prevT[0] + _prevT[1]) + h4 * c0 * _prevT[0] + _F0 * _tau
+ _tau * h4 * (fHalf + f0);
}
```

Листинг 5. Вычисление температуры

```
{
    QVector<double> A, B, C, F;
    QVector<QVector<double>> result;
    double K0, M0, P0;
    double KN, MN, PN;
    Sweep sweep;
    _d = _alphaN * _l / (_alphaN - _alpha0);
    _c = -_d * _alpha0;

    _currT = QVector<double>(floor(_l / _h) + 1, _T0);
    result.push_back(_currT);
    double t = 0;
    do
    {
        _prevT = _currT;
        _currIteration = _prevT;
        do
        {
            _prevIteration = _currIteration;
            calculateCoeffs(A, B, C, F);
            calculateLeftBoundary(K0, M0, P0);
            calculateRightBoundary(KN, MN, PN);
            sweep.setCoeffs(A, B, C, F);
            sweep.setLeftBoundary(K0, M0, P0);
            sweep.setRightBoundary(KN, MN, PN);
            _currIteration = sweep.solve();
        } while (calculateDifference(_currIteration, _prevIteration) > _eps);
        _currT = _currIteration;
        result.push_back(_currT);
        t += _tau;
    } while (calculateDifference(_currT, _prevT) > _eps);
    return result;
}
```

Далее, в экспериментальной части, тестирование будет производиться по этим данным:

$$k(T)=a_1(b_1+c_1T^{m_1}),\quad Вт/см\,K,$$

$$c(T)=a_2+b_2T^{m_2}-\frac{c_2}{T^2},\quad Дж/см^3K.$$

$$\alpha_1=0.0134,\quad b_1=1,\quad c_1=4.35\cdot10^{-4},\quad m_1=1,$$
$$\alpha_2=2.049,\quad b_2=0.563\cdot10^{-3},\quad c_2 0.528\cdot10^{5},\quad m_2=1.$$

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

$\alpha_0=0.05\ Вт/см^2\ K$,
$\alpha_N=0.01\ Вт/см^2\ K$,
$l=10\ см$,
$T_0=300\ K$,
$R=0.5\ см$
$F(t)=50\ Вт/см^2$ (для отладки принять постоянным).

III. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

В данном разделе будет рассмотрен вывод программы и представлены графики зависимостей. Замеры проводились при точности $\varepsilon = 10^{-5}$.

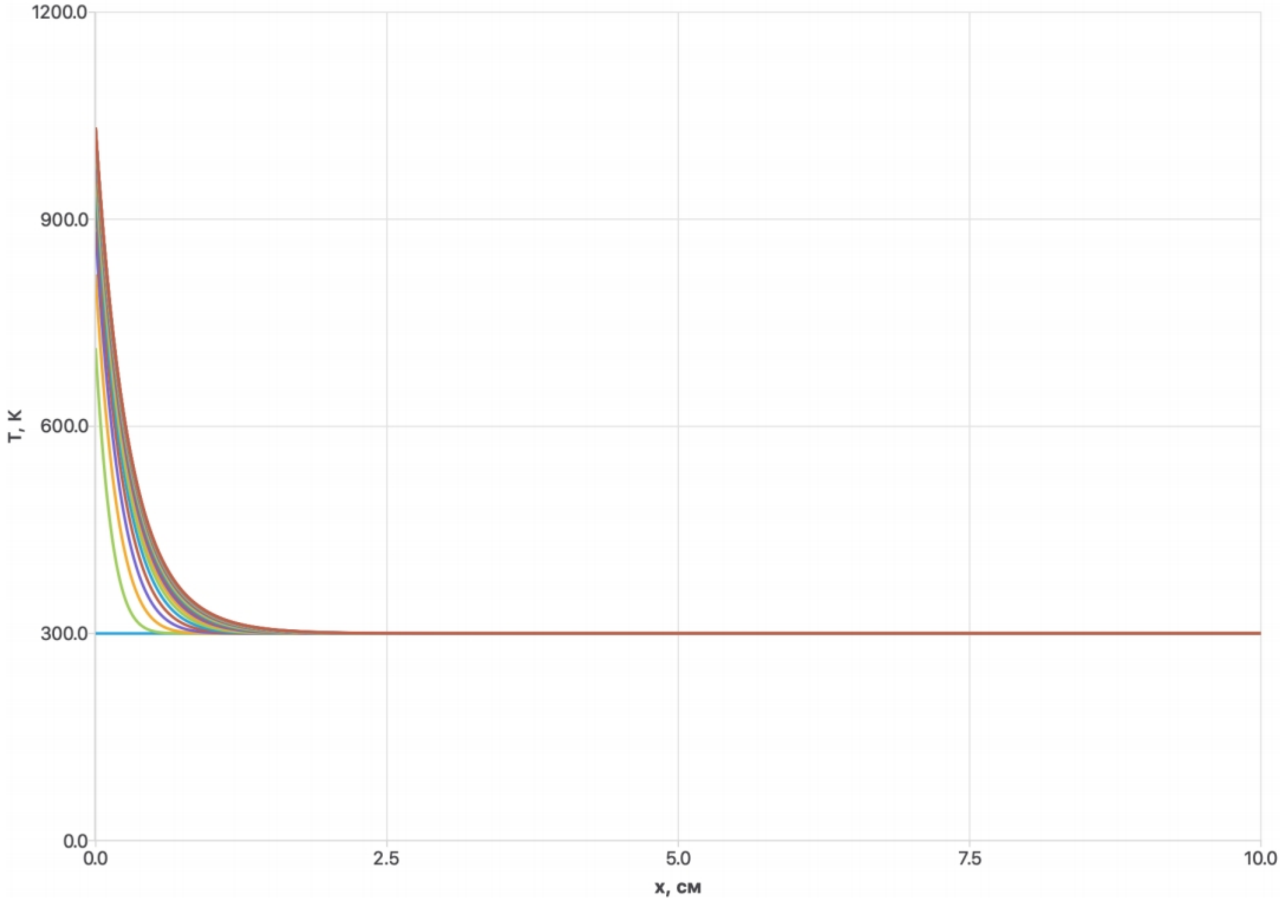

Рисунок 1. График температуры от координаты.

График зависимости температуры $T(x,t_m)$ от координаты x при нескольких фиксированных значениях времени t_m заданных выше параметрах. Здесь *нижний график* — температура в нулевой момент времени, *верхний график* — температура соответствующая установленному режиму. Из-за использования отличных от 3й лабораторной работы коэффициентов, график стационарного режима будет наклонным.

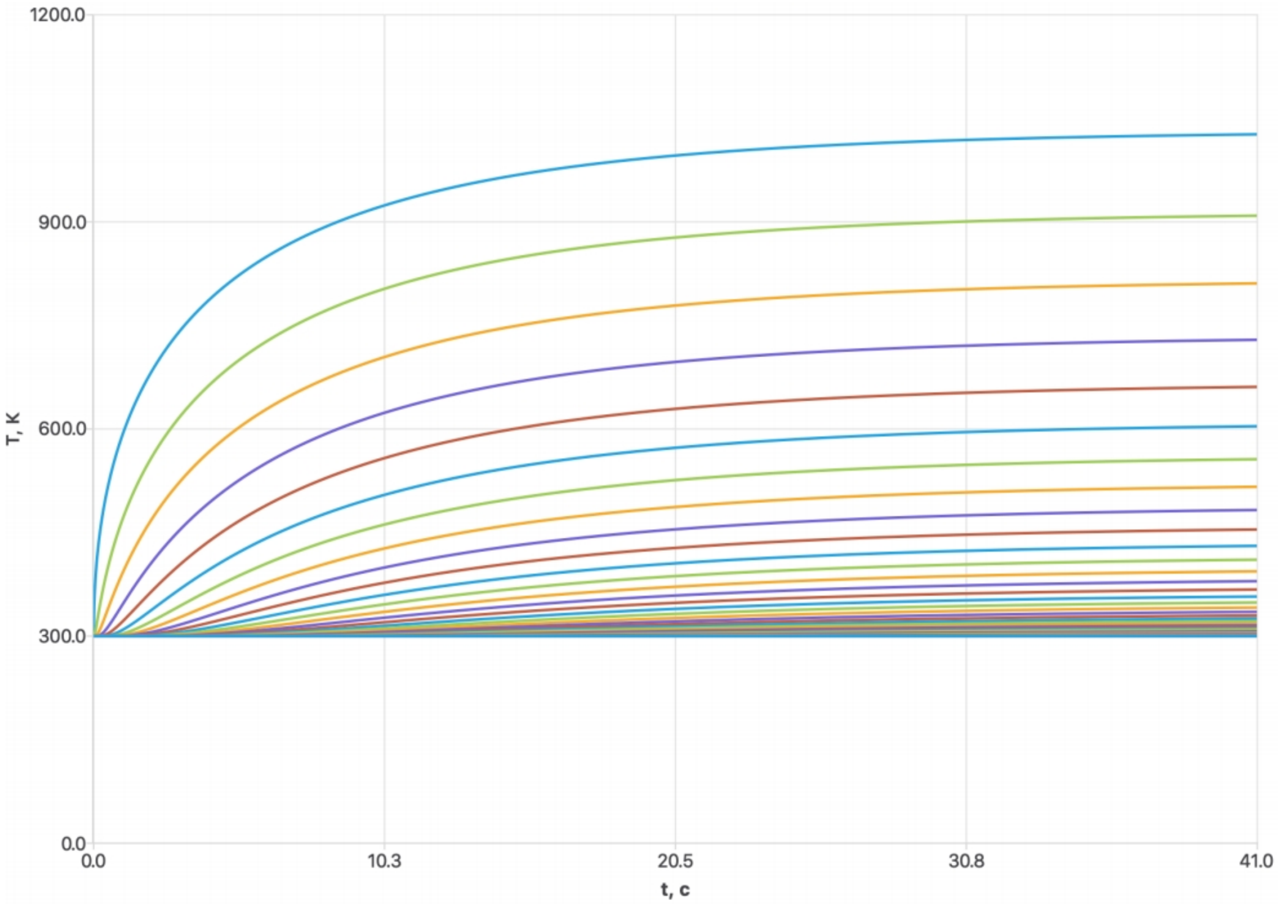

Рисунок 2. Зависимость температуры от времени.

График зависимостей $T(x_n,t)$ при нескольких фиксированных x_n. Здесь, *нижний график* — правый конец стрежня (в данном случае $x=x_N=l$), *верхний график* — левый конец стержня, который нагружается тепловым потоком (в данном случае $x=x_0=0$.)

IV. ОТВЕТЫ НА ВОПРОСЫ

1. Приведите результаты тестирования программы (графики, общие соображения, качественный анализ).

- Как тест, можно задать температуру окружающей среды стержню, т.е. считать тепловой поток нулевым ($F_0 = 0$):

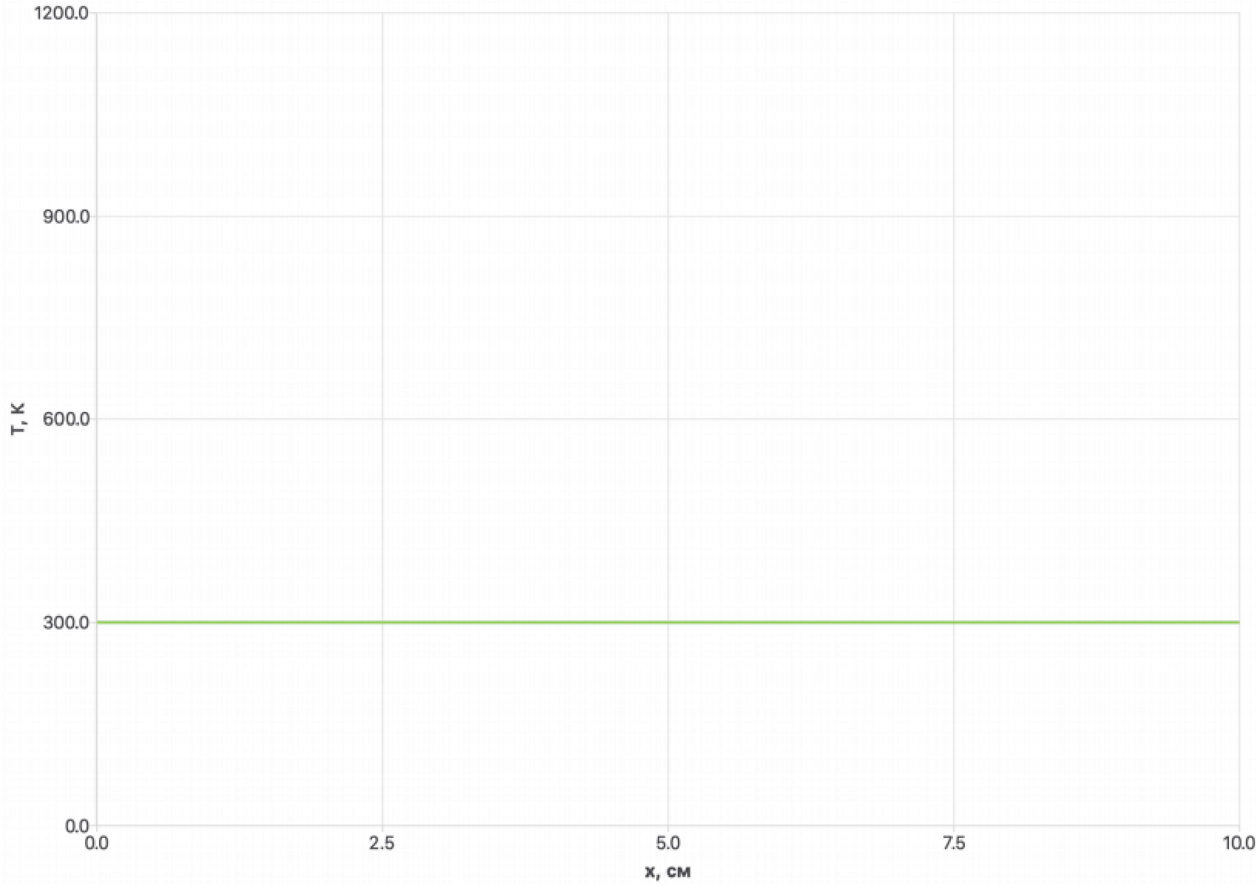

Рисунок 3. График температуры при отсутствии нагрева.

- Также, учитывая, что лабораторная частично схожа с 3й, для тестирование программы можно сравнить графики, получившиеся при выходе на стационарном режиме, с графиком, который мы получили в предыдущей лабораторной работе. Для этого избавляемся от зависимости коэффициента теплопроводности от T, заменяя его коэффициентом который будет зависеть только от координаты, так же как и в предыдущей лабораторной работе (*теплоёмкость обнуляем*):

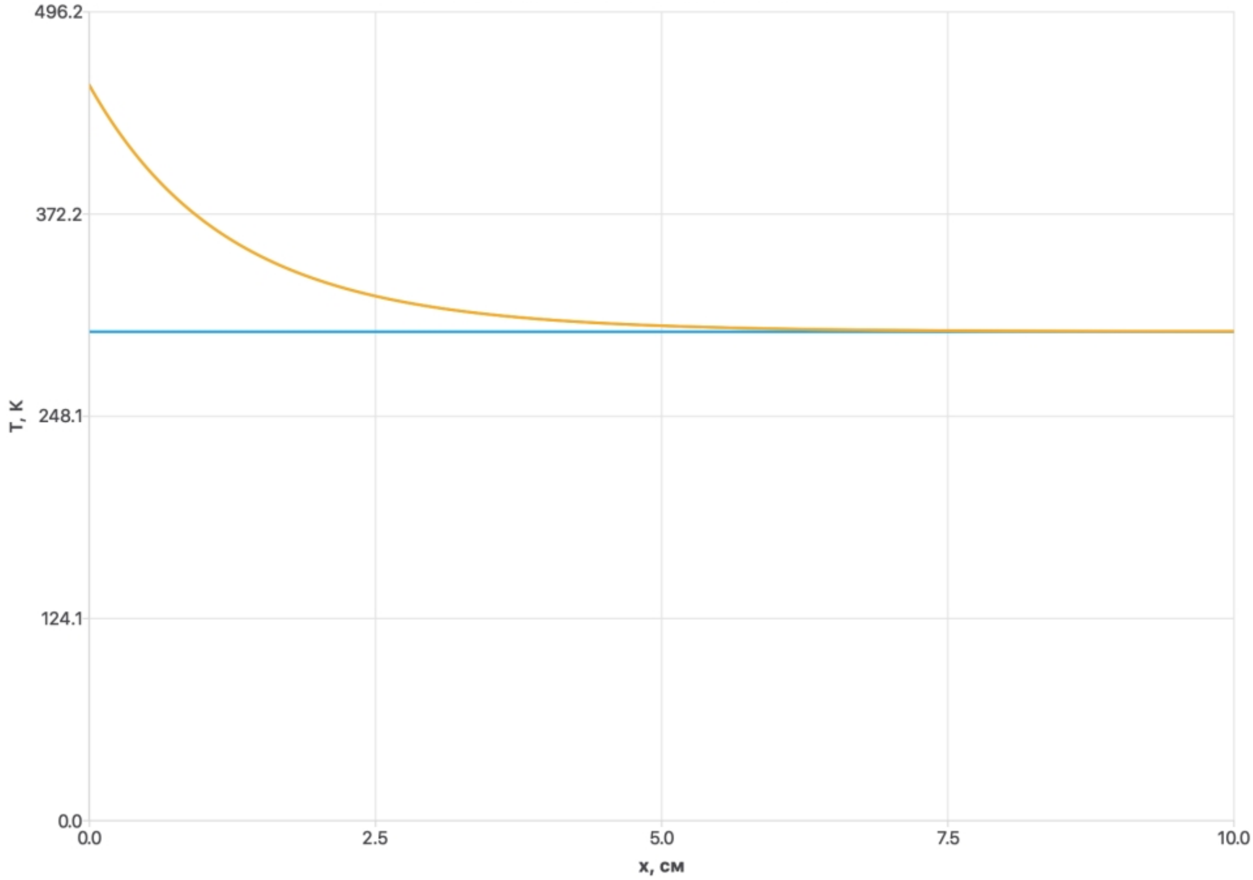

Рисунок 4. График температуры при параметрах из 3й ЛР.

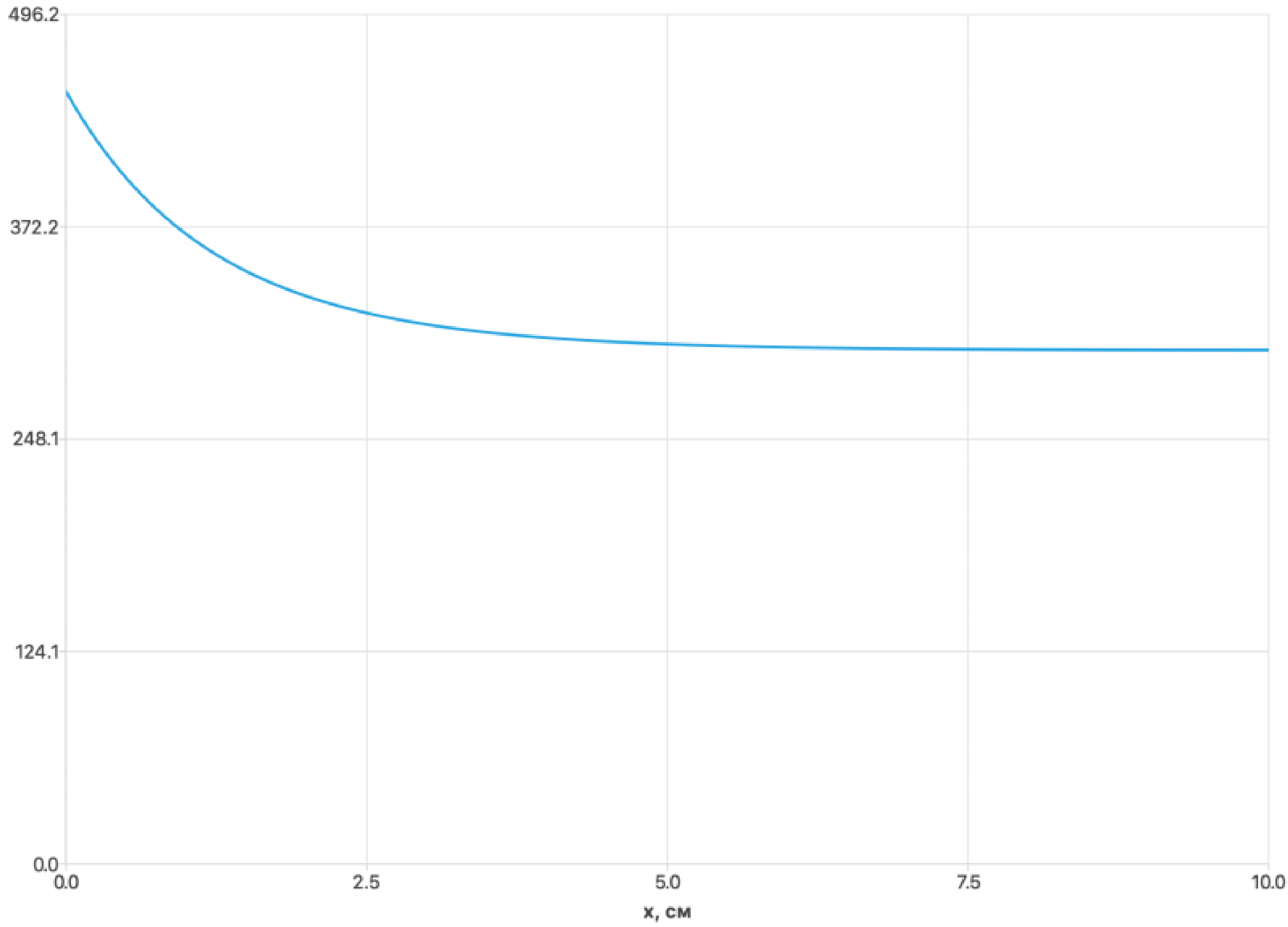

Рисунок 6. График температуры из 3й ЛР.

Как видим, они идентичны.

Также возможный метод тестирования — при разогретом стрежне температура перестаёт расти, обнулить тепловой поток ($F_0=0$):

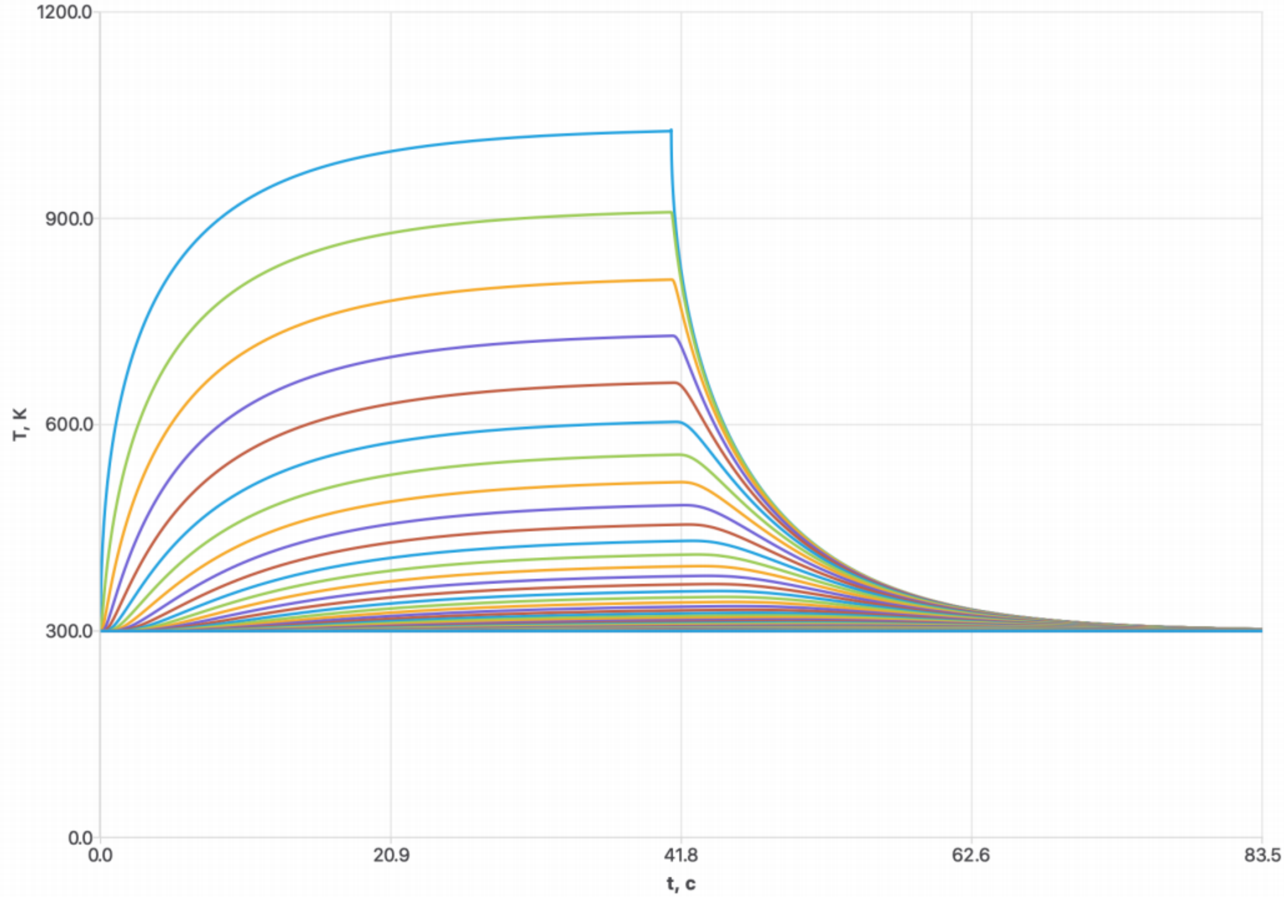

Рисунок 7. График нагрева и остывания стрежня после обнуления теплового потока.

Как видим по графику, после отсутствия нагрева левого конца, начинается процесс остывания до температуры окружающей среды (T_0).

2. Выполните линеаризацию уравнения (9), по Ньютону, полагая для простоты, что все коэффициенты зависят только от одной переменной $\widehat{y_n}$. Приведите линеаризованный вариант уравнения и опишите алгоритм его решения.

$$\begin{cases} \widehat{K_0}\,\widehat{y_0}+\widehat{M_0}\,\widehat{y_1}=\widehat{P_0}, \\ \widehat{A_n}\,\widehat{y_{n-1}}-\widehat{B_n}\,\widehat{y_n}+\widehat{D_n}\,\widehat{y_{n+1}}=-\widehat{F_n}, \quad 1\le n\le N-1 \\ \widehat{K_N}\,\widehat{y_N}+\widehat{M_{N-1}}\,\widehat{y_{N-1}}=\widehat{P_N} \end{cases} \qquad \textbf{(9)}$$

Выполним линеаризацию по переменным $\widehat{y_{n-1}}, \widehat{y_n}, \widehat{y_{n+1}}$:

$$\left(\widehat{A_n}\,\widehat{y_{n-1}}-\widehat{B_n}\,\widehat{y_n}+\widehat{D_n}\,\widehat{y_{n+1}}\,\widehat{F_n}\right)\Big|_{s-1}+\widehat{A^{s-1}}\,\Delta\,\widehat{y^s_{n-1}}+\left(\frac{\partial\widehat{A_n}}{\partial\widehat{y_n}}y_{n-1}-\frac{\partial\widehat{B_n}}{\partial\widehat{y_n}}\widehat{y_n}-\widehat{B_n}+\frac{\partial\widehat{D_n}}{\partial\widehat{y_n}}\widehat{y_{n+1}}+\frac{\partial\widehat{F_n}}{\partial\widehat{y_n}}\right)\Big|_{s-1}$$

$$\Delta\,\widehat{y^s_n}+\widehat{D^{s-1}_n}\,\Delta\,\widehat{y^s_{n+1}}=0$$

Приведя к каноническому виду, получим:

$$A_n\Delta\,\widehat{y^s_{n-1}}-B_n\Delta\,\widehat{y^s_n}+D_n\Delta\,\widehat{y^s_{n+1}}=-F_n, \quad 1\le n\le N-1 \qquad \textbf{(10)}$$

где,

$$A_n=\widehat{A^{s-1}_n},$$
$$D_n=\widehat{D^{s-1}_n},$$
$$B_n=\left(-\frac{\partial\widehat{A_n}}{\partial\widehat{y_n}}y_{n-1}+\frac{\partial\widehat{B_n}}{\partial\widehat{y_n}}\widehat{y_n}+\widehat{B_n}-\frac{\partial\widehat{D_n}}{\partial\widehat{y_n}}\widehat{y_{n+1}}+\frac{\partial\widehat{F_n}}{\partial\widehat{y_n}}\right)\Big|_{s-1}$$
$$F_n=\left(A_n\,\widehat{y_{n-1}}-B_n\,\widehat{y_n}+D_n\,\widehat{y_{n+1}}+\widehat{F_n}\right)\Big|_{s-1}$$

Полученная система уравнений с трехдиагональной матрицей решаются методом прогонки с краевыми условиями:

$$\Delta\,\widehat{y^s_0}=0, \quad \Delta\,\widehat{y^s_N}$$

В результате находятся все $\Delta\,\widehat{y^s_n}$, после чего находятся все значения функции для текущей итерации (*S*) по формуле:

$$\widehat{y^s_n}=\widehat{y^{s-1}_n}+\Delta\,\widehat{y^s_n}$$

В качестве начального приближения $\widehat{y^0_n}$ можно задать в сошедшееся решение y_n с предыдущего временного шага $t=t_m$.

Итерационный процесс завершается при выполнении условия

$$max\left|\frac{\Delta\,\widehat{y^s_n}}{\widehat{y^s_n}}\right|\le\varepsilon \quad \text{для всех } n=0,1,\ldots,N.$$

ЗАКЛЮЧЕНИЕ

Были получены навыки разработки алгоритмов решения смешанной краевой задачи при реализации моделей, построенных на квазилинейном уравнении параболического типа. Была выполнена линеаризация уравнения (9) и описан алгоритм его решения. Был получен разностный аналог краевого условия при $x=l$ интегро-интерполяционным методом. Произведено сравнение показаний графиков с 3й ЛР.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Градов В.М. Методические указания: «04-05-2020-Задание_на_лаб_раб__4.doc » (дата обращения 16.05.2020)
2. Градов В.М. Компьютерные технологии в практике математического моделирования часть 2 URL: http://ebooks.bmstu.ru/secret/html/bikqxzugca/files/assets/basic-html/page-1.html (дата обращения 21.05.2020)
3. Градов В.М. Лекция №14 «04-05-2020-Лекция__14_Модели_ДУЧП_Методы_постр_разност_схем_Интегро_интерп.pdf » (дата обращения 20.05.2020)
4. Градов В.М. Лекция №13 «04-05-2020-Лекция__13_Модели_ДУЧП_Методы_постр_разност_схем_Разност_аппроксим.pdf » (дата обращения 19.05.2020)
5. Градов В.М. Лекция №8 «30-03-2020-Лекция_№8_Модели_ОДУ_краевая_задача.pdf» (дата обращения 16.05.2020)