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	Степанов А. О.
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 квазиинейном уравнении параболического типа.

ИСХОДНЫЕ ДАННЫЕ

Задана математическая модель

Уравнение для функции $T(x,t)$ (формула 1).

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

Краевые условия

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

Функция $\alpha(x)$

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

, где

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

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

Разностная схема

Систему квазилинейных растностных уравнений видно на формуле 2.

$$
\begin{cases}
\widehat{K}_0 \widehat{y}_0 + \widehat{M}_0 \widehat{y}_1 = \widehat{P}_0 \\
\widehat{A}_n \widehat{y}_{n-1} - \widehat{B}_n \widehat{y}_n + \widehat{D}_n \widehat{y}_{n+1} = -\widehat{F}_n, \quad 1 \le n \le N-1 \\
\widehat{K}_N \widehat{y}_N + \widehat{M}_{N-1} \widehat{y}_{N-1} = \widehat{P}_N
\end{cases} \tag{2}
$$

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

Краевые условия

Обозначим:

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

Разностный аналог краевого условия при $x = 0$

$$
\left(\frac{h}{8}\widehat{c}_{\frac{1}{2}} + \frac{h}{4}\widehat{c}_0 + \widehat{\chi}_{\frac{1}{2}}\frac{\tau}{h} + \frac{\tau h}{8}p_{\frac{1}{2}} + \frac{\tau h}{4}p_0\right)\widehat{y}_0 + \left(\frac{h}{8}\widehat{c}_{\frac{1}{2}} - \widehat{\chi}_{\frac{1}{2}}\frac{\tau}{h} + \frac{\tau h}{8}p_{\frac{1}{2}}\right)\widehat{y}_1 = \\
= \frac{h}{8}\widehat{c}_{\frac{1}{2}}\left(y_0 + y_1\right) + \frac{h}{4}\widehat{c}_0 y_0 + \widehat{F}\tau + \frac{\tau h}{4}\left(\widehat{f}_{\frac{1}{2}} + \widehat{f}_0\right) \tag{3}
$$

Получим разностный аналог краевого условия $x = l$. Проинтегрируем уравнение 1 на отрезке $[x_{N-\frac{1}{2}}, x_N]$ и на временном интервале $[t_m, t_{m+1}]$.

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

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

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

Подставим 4, 5, 6 и 7.

$$\widehat{y}_{N-\frac{1}{2}} = \frac{\widehat{y}_{N-1} + \widehat{y}_N}{2} \tag{4}$$

$$y_{N-\frac{1}{2}} = \frac{y_{N-1} + y_N}{2} \tag{5}$$

$$\widehat{F}_N = \alpha_N(\,\widehat{y}_N - T_0) \tag{6}$$

$$\widehat{F}_{N-\frac{1}{2}} = \widehat{\chi}_{N-\frac{1}{2}}\,\frac{\widehat{y}_{N-1} - \widehat{y}_N}{h} \tag{7}$$

Получим

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

С помощью формул 2 и 3 получим коэффициенты $\widehat{K}_0, \widehat{M}_0, \widehat{P}_0$, а с помощью 2 и 8 – $\widehat{K}_N, \widehat{M}_{N-1}, \widehat{P}_N$.

Метод простых итераций

Для решения системы 2 используется метод простых итераций. Обозначим текущую итерацию за s, тогда предыдущая – $s-1$. С данными обозначениями итерационный процесс организуется следующим образом

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

Решение данной схемы осуществляется методом прогонки. Итерации прекращаются при условии

$$\max \left| \frac{y_n^s - y_n^{s-1}}{y_n^s} \right| \leq \varepsilon, n = \overline{0; N}$$

Значения параметров для отладки

$$
\begin{aligned}
&k(T) = a_1(b_1 + c_1 T^{m_1}), \tfrac{\text{Вт}}{\text{см К}},\\
&c(T) = a_2 + b_2 T^{m_2} - \tfrac{c_2}{T^2}, \tfrac{\text{Дж}}{\text{см}^3\text{К}},\\
&a_1 = 0.0134,\ b_1 = 1,\ c_1 = 4.35 \cdot 10^{-4},\ m_1 = 1,\\
&a_2 = 2.049,\ b_2 = 0.563 \cdot 10^{-3},\ c_2 = 0.528 \cdot 10^5,\ m_2 = 1\\
&\alpha(x) = \tfrac{c}{x-d},\\
&\alpha_0 = 0.05\ \tfrac{\text{Вт}}{\text{см}^2\text{К}},\\
&\alpha_N = 0.01\ \tfrac{\text{Вт}}{\text{см}^2\text{К}},\\
&l = 10 \text{ см},\\
&T_0 = 300 \text{ К},\\
&R = 0.5 \text{ см}\\
&F(t) = 50 \tfrac{\text{Вт}}{\text{см}^2}.
\end{aligned}
$$

ЛИСТИНГИ

Листинг 1: Теплоемкость стержня

```
double Mathematics::c(const double T)
{
    return _a2 + _b2 * std::pow(T, _m2) - _c2 / (T * T);
}
```

Листинг 2: Коэффициент теплопроводности стержня

```
double Mathematics::k(const double T)
{
    return _a1 * (_b1 + _c1 * std::pow(T, _m1));
}
```

Листинг 3: Коэффициент теплоотдачи при обдуве

```
double Mathematics::alpha(const double x)
{
    double d = (_alphaN * _l) / (_alphaN - _alpha0);
    double c = -_alpha0 * d;
    return c / (x - d);
}
```

Листинг 4: Замены p и f

```
double Mathematics::p(const double x)
{
    return (2.0 / _R) * alpha(x);
}

double Mathematics::f(const double x)
{
    return (2.0 * _T0 / _R) * alpha(x);
}
```

Листинг 5: Метод средних для всех функций

```
double Mathematics::chi1_2(const double T, const double tau)
{
    return (k(T) + k(T + tau)) / 2.0;
}

double Mathematics::c1_2(const double T, const double tau)
{
    return (c(T) + c(T + tau)) / 2.0;
}

double Mathematics::p1_2(const double x, const double h)
{
    return (p(x) + p(x + h)) / 2.0;
}

double Mathematics::f1_2(const double x, const double h)
```

```
{
    return (f(x) + f(x + h)) / 2.0;
}
```

Листинг 6: Параметры разностной схемы

```
double Mathematics::A(const double T)
{
    return chi1_2(T, -_tau) * _tau / _h;
}

double Mathematics::B(const double T, const double x)
{
    return A(T) + D(T) + c(T) * _h + p(x) * _h * _tau;
}

double Mathematics::D(const double T)
{
    return chi1_2(T, _tau) * _tau / _h;
}

double Mathematics::F(const double T, const double x)
{
    return f(x) * _h * _tau + c(T) * T * _h;
}
```

Листинг 7: Метод прогонки

```
QVector<double> Mathematics::runTrought(const QVector<double> &prev)
{
    const double K0 = _h / 8.0 * c1_2(prev[0], _tau) +
        _h / 4.0 * c(prev[0]) + chi1_2(prev[0], _tau) *
        _tau / _h + _tau * _h / 8.0 * p1_2(0, _h) +
        _tau * _h / 4.0 * p(0);
    const double M0 = _h / 8.0 * c1_2(prev[0], _tau) -
        chi1_2(prev[0], _tau) * _tau / _h +
        _tau * _h / 8.0 * p1_2(0, _h);
    const double P0 = _h / 8.0 * c1_2(prev[0], _tau) * (prev[0] + prev[1]) +
        _h / 4.0 * c(prev[0]) * prev[0] +
        _Ft * _tau +
        _tau * _h / 4.0 * (f1_2(0, _h) + f(0));

    const double KN = _h / 4.0 * c(prev.last()) +
        _h / 8.0 * c1_2(prev.last(), -_tau) +
```

```
            _alphaN * _tau +
            chi1_2(prev.last(), -_tau) * _tau / _h +
            p(_l) * _tau * _h / 4.0 +
            p1_2(_l, -_h) * _tau * _h / 8.0;
    const double MN = _h / 8.0 * c1_2(prev.last(), -_tau) -
            chi1_2(prev.last(), -_tau) * _tau / _h +
            p1_2(_l, -_h) * _tau * _h / 8.0;
    const double PN = _h / 4.0 * c(prev.last()) * prev.last() +
            _h / 8.0 * c1_2(prev.last(), -_tau) * prev[prev.count() - 2] +
            _h / 8.0 * c1_2(prev.last(), -_tau) * prev.last() +
            _T0 * _alphaN * _tau +
            (f(_l) + f1_2(_l, -_h)) * _tau * _h / 4.0;

    QVector<double> eps;
    eps.append(-M0 / K0);

    QVector<double> eta;
    eta.append(P0 / K0);

    int n = 1;
    for (double x = _h; x + _h < _l; x += _h, n += 1) {
        double epsN = eps.last();
        double etaN = eta.last();
        eps.append(D(prev[n]) / (B(prev[n], x) - A(prev[n]) * epsN));
        eta.append((F(prev[n], x) + A(prev[n]) * etaN) /
            (B(prev[n], x) - A(prev[n]) * epsN));
    }

    QVector<double> t(eps.count());
    t[t.count() - 1] = (PN - MN * eta.last()) / (KN + MN * eps.last());

    for (int i = t.count() - 2; i >= 0; --i) {
        t[i] = eps[i + 1] * t[i + 1] + eta[i + 1];
    }

    return t;
}
```

Листинг 8: Метод простых итераций

```
void Mathematics::iterations()
{
    if (!_secondRun) {
```

```
        QVector<double> tZero;
        int n = int(_l / _h) + 1;
        for (int i = 0; i < n; ++i) {
            if (_anotherStart) {
                tZero.append(1000);
            } else {
                tZero.append(_T0);
            }
        }

        temp.append(tZero);
    }

    do {
        QVector<double> prev;
        QVector<double> curr = temp.last();

        do {
            prev = curr;
            curr = runTrought(prev);
        } while (!endRunTrought(prev, curr));

        temp.append(curr);
    } while (!endIterations());
}
```

Листинг 9: Условие завершения прогонки

```
bool Mathematics::endRunTrought(
    const QVector<double> &prev,
    const QVector<double> &current
)
{
    double max = std::fabs((current[0] - prev[0]) / current[0]);

    for (int i = 1; i < std::min(current.count(), prev.count()); ++i) {
        double e = std::fabs((current[i] - prev[i]) / current[i]);

        if (e > max)
            max = e;
    }

    return max < _eps;
```

```
}
```

Листинг 10: Условие завершения метода простых итераций

```
bool Mathematics::endIterations()
{
    int last = temp.count() - 1;
    for (int i = 0; i < temp[last].count(); ++i) {
        if (std::fabs(
                (temp[last][i] - temp[last - 1][i]) / temp[last][i]
            ) > _eps)
            return false;
    }

    return true;
}
```

Листинг 11: Запуск вычислений

```
Mathematics::Mathematics(
    const double alpha0, const double alphaN, const double l,
    const double T0, const double R, const double Ft,
    const bool again, const bool another
) : _alpha0(alpha0), _alphaN(alphaN), _l(l),
    _T0(T0), _R(R), _Ft(Ft),
    _again(again), _anotherStart(another)
{
    if (_anotherStart) {
        _Ft = 0;
    }

    iterations();

    if (_again) {
        _Ft = 0;
        _secondRun = true;
        iterations();
    }
}
```

РЕЗУЛЬТАТЫ РАБОТЫ

На рисунке 1 представлен график зависимости температуры от координаты стержня при фиксированных значениях времени t. На этом графике последняя кривая, оранжевая, соответствет установившемуся режиму, когда поле перестанет меняться с точностью 10^{-4}.

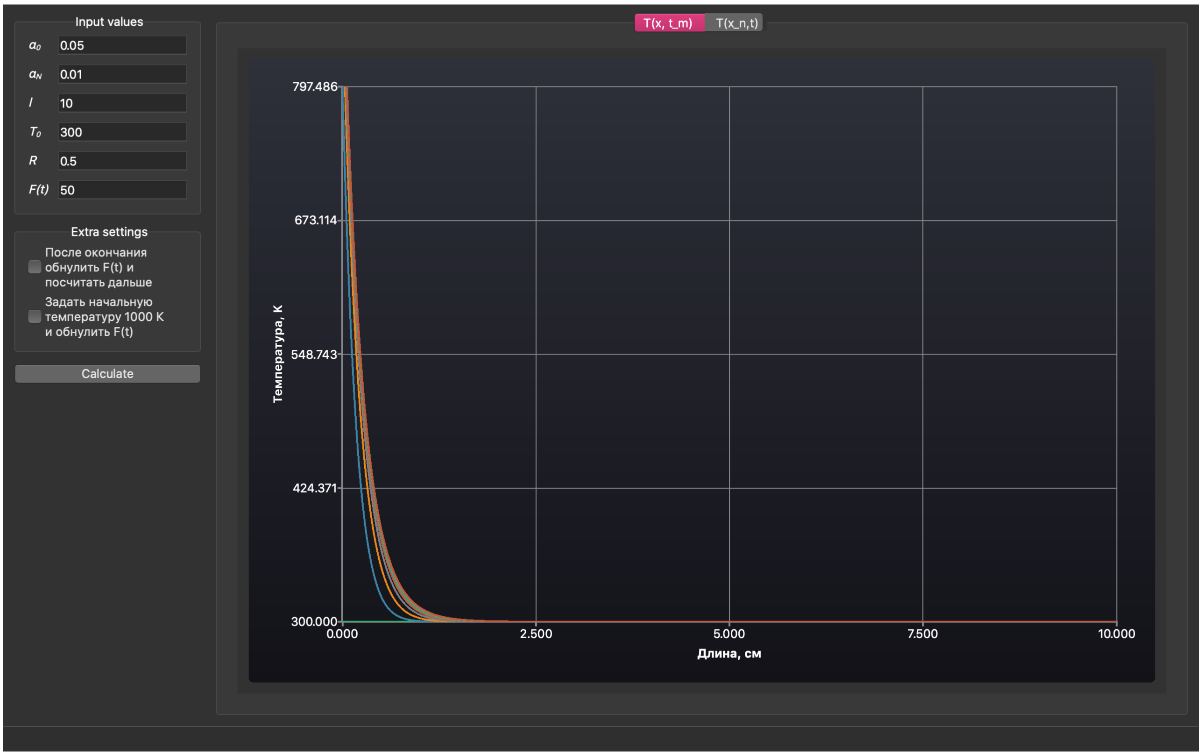

Рис. 1: Зависимость температуры от координаты стержня

На рисунке 2 представлен график зависимости температуры от времени при нескольких фиксированных значениях координаты x.

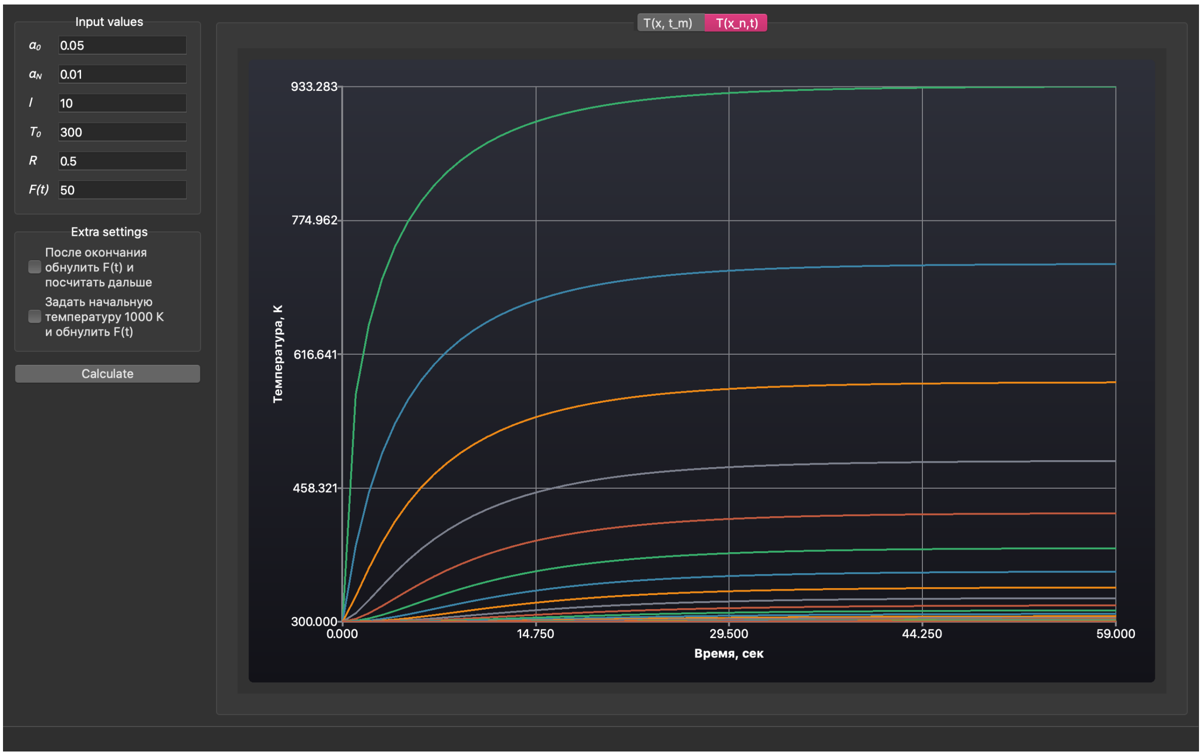

Рис. 2: Зависимость температуры от времени

ОТВЕТЫ НА ВОПРОСЫ

1. **Приведите результаты тестирования программы (графики, общие соображения, качественный анализ). Учесть опыт выполнения лабораторной работы №3.**

   На рисунке 3 виден график (зависимость температуры от времени при фиксированных координатах стержня) в случае, когда после нагрева стержня мы устанавливаем поток равный 0 и запускаем вычисления заново с учетом последней температуры. В таком случае стержень после нагрева начнет остывать до температуры окружающей среды, которая равна 300 K.

Рис. 3: Сначала нагрев стержня, затем остывание

На рисунке 4 виден график (зависимость температуры от времени при фиксированных координатах стержня) в случае, если нагреть стержень до температуры 1000 K, а поток установить равным 0. В такой ситуации стержень будет остывать до тех пор, пока температура не станет равной температуре окружающей среды, которая равна 300 K.

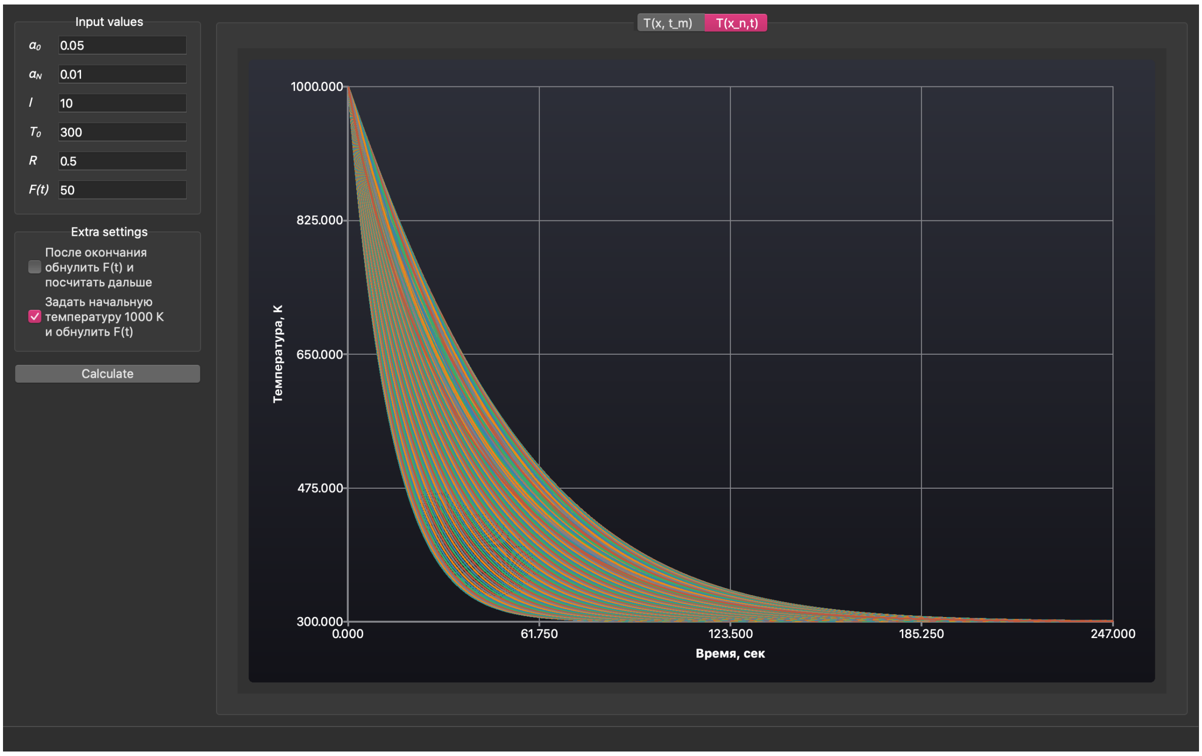

Рис. 4: Стержень остывает до температуры 300 K

Если для коэффициента теплопроводности стержня поменять зависимость от T на зависимость от x как в третьей лабораторной, а теплоемкость стержня задать равной нулю, то график $T(x,t)$ из этой лабораторной работы (рисунок 6) будет совпадать с графиком $T(x)$ из прошлой лабораторной работы (рисунок 5).

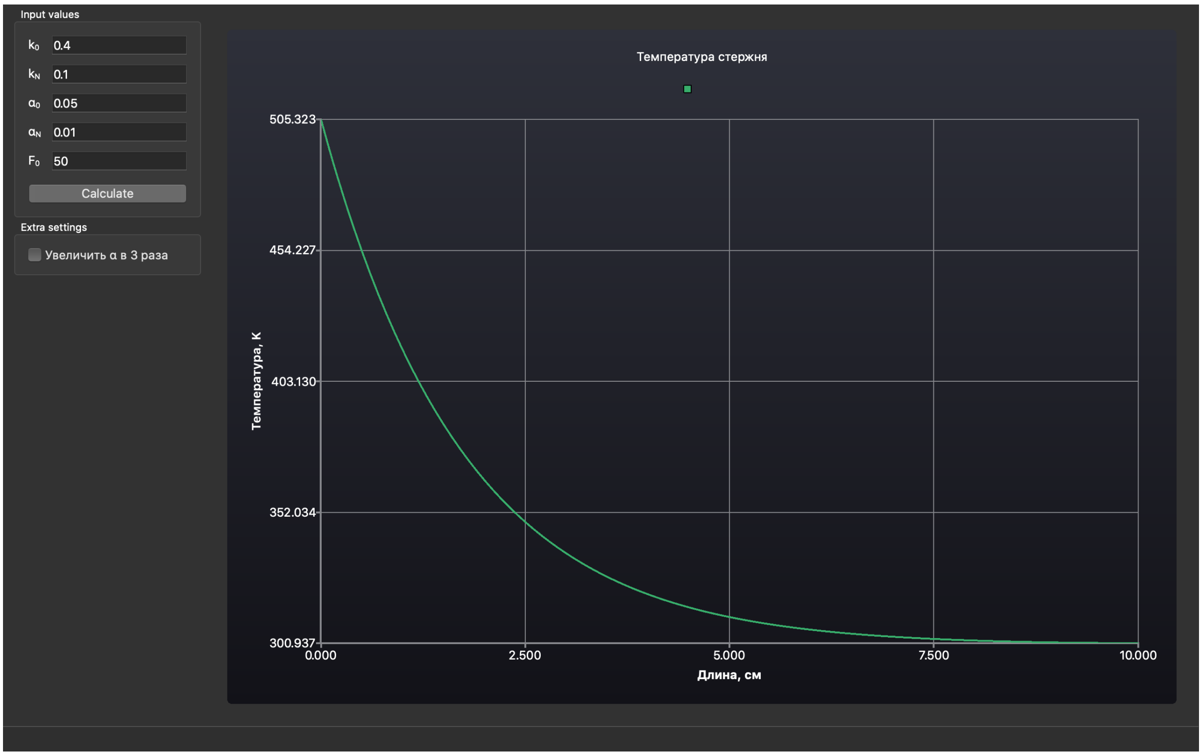

Рис. 5: График из третьей лабораторной работы

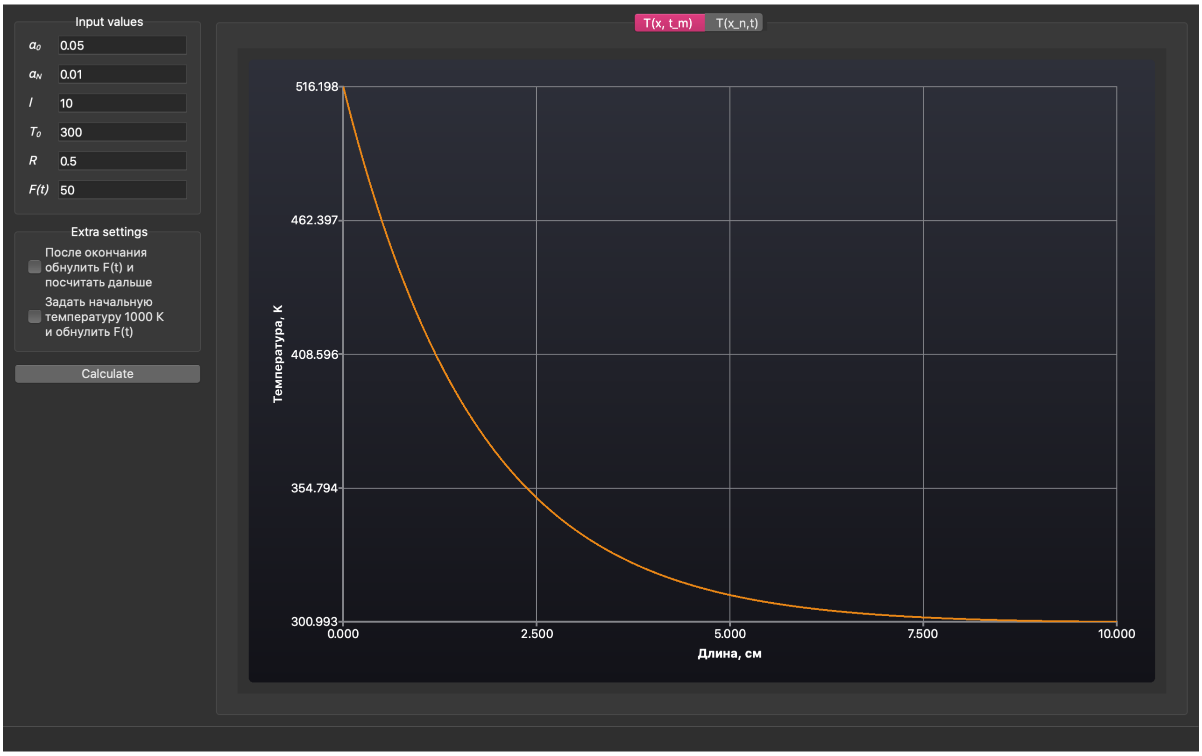

Рис. 6: График из данной лабораторной работы

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

$$\widehat{A}_n \widehat{y}_{n-1} - \widehat{B}_n \widehat{y}_n + \widehat{D}_n \widehat{y}_{n+1} = - \widehat{F}_n$$

Все коэффициенты зависят только от одной переменной, тогда

$$\left(\widehat{A}_n \widehat{y}_{n-1} - \widehat{B}_n \widehat{y}_n + \widehat{D}_n \widehat{y}_{n+1} + \widehat{F}_n \right)\Big|_{s-1} + \widehat{A}_n^{s-1} \Delta \widehat{y}_{n-1}^{s} +$$
$$+ \left(\frac{\partial \widehat{A}_n}{\partial \widehat{y}_n} \widehat{y}_{n-1} - \frac{\partial \widehat{B}_n}{\partial \widehat{y}_n} \widehat{y}_n - \widehat{B}_n + \frac{\partial \widehat{D}_n}{\partial \widehat{y}_n} \widehat{y}_{n+1} + \frac{\partial \widehat{F}_n}{\partial \widehat{y}_n} \right)\Big|_{s-1} \Delta \widehat{y}_n^{s} +$$
$$+ \widehat{D}_n^{s-1} \Delta \widehat{y}_{n+1}^{s} = 0$$

Каноничный вид

$$A_n \Delta \widehat{y}_{n-1}^{s} - B_n \Delta \widehat{y}_n^{s} + D_n \Delta \widehat{y}_{n+1}^{s} = -F_n$$

Тогда получаем, что

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

Краевое условие для y_0

$$\widehat{K}_0 \widehat{y}_0 + \widehat{M}_0 \widehat{y}_1 = \widehat{P}_0$$

Тогда

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

Каноничный вид

$$K_0 \Delta \widehat{y}_0^{s} + M_0 \Delta \widehat{y}_1^{s} = P_0$$

Тогда получаем, что

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

Краевое условие для y_N

$$\overset{\frown}{K}_N \overset{\frown}{y}_N + \overset{\frown}{M}_{N-1} \overset{\frown}{y}_{N-1} = \overset{\frown}{P}_N$$

Тогда

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

Каноничный вид

$$K_N \Delta \overset{\frown}{y}{}_N^{s} + M_{N-1} \Delta \overset{\frown}{y}{}_{N-1}^{s} = P_N$$

Тогда получается, что

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

Система

$$\begin{cases}
K_0 \Delta \overset{\frown}{y}{}_0^{s} + M_0 \Delta \overset{\frown}{y}{}_1^{s} = P_0\\
A_n \Delta \overset{\frown}{y}{}_{n-1}^{s} - B_n \Delta \overset{\frown}{y}{}_n^{s} + D_n \Delta \overset{\frown}{y}{}_{n+1}^{s} = -F_n\\
K_N \Delta \overset{\frown}{y}{}_N^{s} + M_{N-1} \Delta \overset{\frown}{y}{}_{N-1}^{s} = P_N
\end{cases}$$

Решается методом прогонки, в результате находятся все $\Delta \overset{\frown}{y}{}_n^{s}$, 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 $\max\left|\frac{\Delta\overset{\frown}{y}{}^{s}_{n}}{\overset{\frown}{y}{}^{s}_{n}}\right|\leq\varepsilon$.