



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ
(ШКОЛА)

Департамент математического и компьютерного моделирования

ОТЧЁТ

к лабораторной работе №1 «Модель нагревателя»
по дисциплине
«Математическое и компьютерное моделирование»

Направление подготовки
01.03.02 «Прикладная математика и информатика»

Выполнил студент гр.

Б9122-01.03.02мкт

Вершинин Д.А.

(Ф.И.О.)

(подпись)

Проверил профессор д.ф.-м. н.

Пермяков М. С.

(Ф.И.О.)

(подпись)

« 31 » марта 2025 г.

г. Владивосток

2025

Оглавление

Введение	3
1 Постановка задачи	4
1.1 Формализация	4
2 Построение математической модели	5
2.1 Модель без терморегулятора	5
2.2 Модель с терморегулятором	6
3 Анализ математической модели	7
3.1 Вычисление точек покоя	7
4 Вычислительные эксперименты	9
4.1 Алгоритм решения	9
4.2 Программа для ЭВМ	9
4.3 Модели без терморегулятора	13
4.4 Модели с терморегулятором	13
5 Заключение	17
Список использованных источников	18

Введение

Каждый день люди встречают нагревательные приборы. Это может быть паяльник, электрическая кухонная плита, радиатор отопления или обычный электрический утюг.

Каждый из приборов имеет свои характеристики, и человеку, как пользователю, хочется знать, как быстро будет нагреваться тот или иной прибор. Для этого можно построить математическую модель, которая сможет спрогнозировать поведение прибора. Кроме того, почти каждый нагревательный прибор оснащён терморегулятором.

В данной работе рассмотрим построение математической модели нагревателя на примере паяльника с терморегулятором и без него.

1 Постановка задачи

Целью данной работы является построение математической модели нагревательного прибора, анализирующего его тепловые процессы. Рассмотрим две модели: первую — без терморегулятора, и вторую — с терморегулятором, который предотвращает перегрев устройства.

Исходные данные:

- Температура нагревателя $T(t)$ в зависимости от времени t .
- Постоянная температура окружающей среды T_{env} .
- Масса нагревателя m_H и удельная теплоёмкость c .
- Коэффициент теплопередачи k и площадь нагревательного элемента S .
- Постоянная Стефана-Больцмана σ .

1.1 Формализация

Необходимо вывести уравнение теплового баланса, учитывающее изменение внутренней энергии нагревателя из-за электрической мощности, входящие и исходящие тепловые потоки, а также излучение, согласно закону Стефана-Больцмана[1].

- Получить дифференциальное уравнение теплового баланса с начальными условиями $T(0) = T_0$.
- Во второй модели необходимо учесть работу терморегулятора для предотвращения перегрева нагревателя.
- Ввести функцию, которая будет управлять включением и выключением нагревателя в зависимости от температуры.
- Добавить функцию управления в уравнение теплового баланса.

Решение обеих моделей позволит проанализировать динамику изменения температуры нагревательного прибора.

2 Построение математической модели

2.1 Модель без терморегулятора

Основной характеристикой нагревательного прибора является температура. При включенном нагревателе она изменяется со временем. Нас интересует зависимость изменения температуры ($[T] = \text{K}$) от времени ($[t] = \text{с}$): $T(t)$.

Предположим, что нагреватель состоит из одного материала, температура окружающей среды постоянная и равна T_{env} . Также отметим, что масса окружающей среды намного больше массы нагревательного прибора (паяльника): $m_{env} \gg m_H$.

Процесс нагревания описывается [2] изменением количества внутренней энергии тела (ΔQ , $[Q] = \text{Дж}$) от изменении температуры (ΔT):

$$\Delta Q = cm\Delta T, \quad (1)$$

где c - удельная теплоёмкость тела ($\frac{\text{Дж}}{\text{кг}\cdot\text{K}}$), m - масса нагревателя (кг).

Нагревательный прибор использует электрический ток для увеличения внутренней энергии:

$$\Delta Q_1 = P\Delta t, \quad (2)$$

где P - мощность (Вт).

На изменение внутренней энергии также влияют входящие и исходящие тепловые потоки. На единицу площади за единицу времени исходящий поток будет изменять энергию на величину $-kT$, а входящий - на величину kT_{env} , где k - коэффициент теплопередачи, характерный для данной конструкции нагревательного прибора ($\frac{\text{Вт}}{\text{м}^2\cdot\text{K}}$). С учётом этих явлений, внутренняя энергия будет изменяться на следующую величину:

$$\Delta Q_2 = -kS(T - T_{env})\Delta t, \quad (3)$$

где S - площадь нагревательного элемента (м^2).

Кроме этих явлений, согласно закону Стефана-Больцмана, любое тело, нагретое выше абсолютного нуля за единицу времени на единицу площади излучает энергию равную $-\sigma T^4$, где $\sigma \approx 5.68 \cdot 10^{-8} \left(\frac{\text{Вт}}{\text{м}^2\cdot\text{K}^4} \right)$

- постоянная Стефана-Больцмана. Аналогично, излучение поступает из окружающей среды, равное σT_{env}^4 . Тогда, изменение внутренней энергии, вызванного этим процессом, равно:

$$\Delta Q_3 = -\sigma S(T^4 - T_{env}^4)\Delta t. \quad (4)$$

Суммируя все потоки энергии, получаем уравнение теплового баланса (см. 1 - 4):

$$cm\Delta T = P\Delta t - kS(T - T_{env})\Delta t - \sigma S(T^4 - T_{env}^4)\Delta t. \quad (5)$$

Разделим, обе части уравнения (5) на $cm\Delta t$ и совершим предельный переход $\Delta t \rightarrow 0$:

$$\frac{dT}{dt} = \frac{P - kS(T - T_{env}) - \sigma S(T^4 - T_{env}^4)}{cm} \quad (6)$$

Таким образом, мы получили дифференциальное уравнение теплового баланса, которое описывает поведение температуры нагревателя. Для нахождения единственного решения достаточно ввести начальное условие: $T(0) = T_0$.

2.2 Модель с терморегулятором

Для предотвращения перегрева нагревателя целесообразно установить терморегулятор, который будет выключать нагреватель при достижении максимальной температуры. Для этого достаточно ввести функцию, которая будет отключать нагреватель, когда температура больше максимально установленной (T_{max}), и включать, при достижении минимальной установленной температуры (T_{min}).

$$I(T, T_{min}, T_{max}) = \begin{cases} 1, & T < T_{min}, \\ 0, & T > T_{max}. \end{cases} \quad (7)$$

Добавляя (7) в (6) получим:

$$\frac{dT}{dt} = \frac{P \cdot I(T, T_{min}, T_{max}) - kS(T - T_{env}) - \sigma S(T^4 - T_{env}^4)}{cm}, \quad (8)$$

3 Анализ математической модели

Сперва найдём точки равновесия уравнения (6).

Заметим, что удельная теплоёмкость (c) и масса (m) нагревателя находятся в знаменателе. Это значит, что эти параметры не влияют на точки равновесия, но регулируют скорость изменения температуры в самом уравнении.

В начальный момент времени уравнение будет иметь следующий вид:

$$\frac{dT}{dt} = \frac{P}{cm} > 0. \quad (9)$$

На основе (9) можно сделать вывод, что температура будет увеличиваться. Со временем, отрицательные слагаемые будут увеличиваться по модулю, что вызовет замедление увеличения скорости роста температуры.

Это уменьшение скорости роста температуры, будет продолжаться до тех пор, пока отрицательные слагаемые, отвечающие за изменения внутренней энергии (3, 4), не уравновесят увеличение внутренней энергии от работы электрического тока (1). Достигнутая температура будет максимальной и постоянной до тех пор, пока нагреватель будет включен.

Отметим, что достигнутая максимальная температура будет единственной. Это связано с тем, что дальнейший прирост внутренней энергии компенсируется соответствующим ростом по модулю отрицательных слагаемых.

3.1 Вычисление точек покоя

Для вычисления точек покоя выберем параметры нагревательного элемента. В качестве него возьмём паяльник с радиусом 0.003м и длиной 0.05м.

$$P = 35\text{Вт}, m = 0.25\text{кг}, c = 375 \frac{\text{Дж}}{\text{кг} \cdot \text{К}}, k = 2,$$

$$S = 2\pi rh = 0.00094\text{м}^2, T_{env} = 296\text{К}$$

Точные значения точек равновесия для этих параметров будут такие:

$$T_1 = -917, T_2 = 895.53, T_{3,4} = 10.74 \pm 906.4i.$$

Получили только одно положительное значение. Проверим его, используя метод первого приближения. Обозначив правую часть дифференциального уравнения за R , получим:

$$\frac{dR}{dT} = -0.0017. \quad (10)$$

Значение отрицательное, следовательно, положение устойчиво (система возвращается в равновесие при малых возмущениях).

4 Вычислительные эксперименты

4.1 Алгоритм решения

Для численного решения дифференциального уравнения будем использовать алгоритм Рунге-Кутты [3] четвертого порядка.

Метод Рунге-Кутты четвертого порядка (RK4) обладает порядком точности $O(h^4)$, что означает, что ошибка метода уменьшается пропорционально четвертой степени размера шага h . Локальная ошибка на каждом шаге составляет $O(h^5)$, а глобальная ошибка, накапливаясь после N шагов, составляет $O(h^3)$. Это делает метод RK4 более точным по сравнению с методами более низкого порядка, такими как метод Эйлера и метод Рунге-Кутты второго порядка. Благодаря высокой точности и простоте реализации, RK4 широко используется для численного решения обыкновенных дифференциальных уравнений в различных областях науки и техники.

Результатом будет являться массив значений, каждое из которых будет являться решением дифференциального уравнения (6) с заданными параметрами.

4.2 Программа для ЭВМ

В качестве языка программирования для расчётов и визуализации был выбран Python с использованием библиотек `numpy` (вычисления) и `matplotlib` (визуализация).

```
1 import numpy as np
2 from matplotlib import pyplot as plt
3
4
5 class SolderingIron:
6     def __init__(self, P, c, m, T0, Tenv, Tmax, Tmin, k, R, h, sigma):
7         self.P = P
8         self.c = c
9         self.m = m
10        self.T0 = T0
11        self.Tenv = Tenv
12        self.Tmax = Tmax
13        self.Tmin = Tmin
14        self.k = k
```

```

15         self.S = 2 * np.pi * R * h
16         self.sigma = sigma
17         self.isOn = True
18
19     def dTdt(self, T):
20         return (self.P - self.k * self.S * (T - self.Tenv) - self.sigma *
21                 self.S * (T ** 4 - self.Tenv ** 4)) / (
22                 self.m * self.c)
23
24     def I(self, T):
25         if T > self.Tmax:
26             self.isOn = False
27         if T < self.Tmin:
28             self.isOn = True
29
30     def dTdt_with_controller(self, T):
31         self.I(T)
32         return (self.P * self.isOn - self.k * self.S * (T - self.Tenv) -
33                 self.sigma * self.S * (
34                     T ** 4 - self.Tenv ** 4)) / (self.m * self.c)
35
36     def solve(self, t0, tn, n):
37         h = (tn - t0) / n
38         T = self.T0
39         t_values, T_values = [t0], [self.T0]
40
41         for _ in range(n):
42             k1 = h * self.dTdt(T)
43             k2 = h * self.dTdt(T + 0.5 * k1)
44             k3 = h * self.dTdt(T + 0.5 * k2)
45             k4 = h * self.dTdt(T + k3)
46             T += (k1 + 2 * k2 + 2 * k3 + k4) / 6
47             t_values.append(t_values[-1] + h)
48             T_values.append(T)
49
50         return t_values, T_values
51
52     def solve_with_controller(self, t0, tn, n):
53         h = (tn - t0) / n
54         T = self.T0
55         t_values, T_values = [t0], [self.T0]
56
57         for _ in range(n):
58             k1 = h * self.dTdt_with_controller(T)
59             k2 = h * self.dTdt_with_controller(T + 0.5 * k1)
60             k3 = h * self.dTdt_with_controller(T + 0.5 * k2)

```

```

59         k4 = h * self.dTdt_with_controller(T + k3)
60         T += (k1 + 2 * k2 + 2 * k3 + k4) / 6
61         t_values.append(t_values[-1] + h)
62         T_values.append(T)
63
64     return t_values, T_values
65
66
67 def plot_temperature_vs_time(P, c, m, T0, Tenv, k, R, h, sigma):
68     soldering_iron = SolderingIron(P, c, m, T0, Tenv, 0, 0, k, R, h, sigma)
69     t_values, T_values = soldering_iron.solve(0, 3600, 1000)
70     plt.plot(t_values, T_values, label=fr"P={P}, c={c}, m={m}, k={k},
71             S={soldering_iron.S:.4f}")
72     plt.xlabel('Time (s)')
73     plt.ylabel('Temperature (K)')
74     plt.title('Temperature vs Time with Different Boundary Conditions')
75     plt.legend()
76     plt.grid(True)
77
78 def make_plot_for_controller(soldering_iron1, soldering_iron2, ):
79     t_values, T_values = soldering_iron1.solve_with_controller(0, 3600,
80         1000)
81     plt.plot(t_values, T_values)
82     plt.axhline(y=soldering_iron1.Tmax, linestyle='—', color='red',
83         label=f'Tmax_1={soldering_iron1.Tmax}')
84     plt.axhline(y=soldering_iron1.Tmin, linestyle='—', color='blue',
85         label=f'Tmin_1={soldering_iron1.Tmin}')
86
87     t_values, T_values = soldering_iron2.solve_with_controller(0, 3600,
88         1000)
89     plt.plot(t_values, T_values)
90     plt.axhline(y=soldering_iron2.Tmax, linestyle='—', color='green',
91         label=f'Tmax_2={soldering_iron1.Tmax}')
92     plt.axhline(y=soldering_iron2.Tmin, linestyle='—', color='purple',
93         label=f'Tmin_2={soldering_iron1.Tmin}')
94
95 def plot_temperature_vs_time_with_controller(C0, P, c, m, T0, Tenv, k, R,
96     h, sigma):
97     soldering_iron_1 = SolderingIron(P, c, m, T0, Tenv, 250 + C0, 200 +
98         C0, k, R, h, sigma)
99     soldering_iron_2 = SolderingIron(P, c, m, T0, Tenv, 190 + C0, 180 +
100         C0, k, R, h, sigma)
101
102     make_plot_for_controller(soldering_iron_1, soldering_iron_2)

```

```

95     plt.xlabel('Time (s)')
96     plt.ylabel('Temperature (K)')
97     plt.title('Temperature vs Time with Controller')
98     plt.legend()
99     plt.grid(True)
100    plt.show()
101
102
103    def main():
104        C0 = 276
105        P = 35
106        c = 375
107        m = 0.25
108        T0 = 25.0 + C0
109        Tenv = 25.0 + C0
110        k = 2
111        R = 0.003
112        h = 0.05
113        sigma = 5.67e-8
114
115        plt.figure(figsize=(12, 8))
116        plot_temperature_vs_time(P, c, m, T0, Tenv, k, R, h, sigma)
117        plot_temperature_vs_time(P * 2, c, m, T0, Tenv, k, R, h, sigma)
118        plot_temperature_vs_time(P, c / 2, m, T0, Tenv, k, R, h, sigma)
119        plot_temperature_vs_time(P, c, m * 2, T0, Tenv, k, R, h, sigma)
120        plot_temperature_vs_time(P, c, m, T0, Tenv, k * 2, R, h, sigma)
121        plot_temperature_vs_time(P, c, m, T0, Tenv, k, R, h * 2, sigma)
122        plt.axhline(y=895.53, linestyle='—')
123
124        plt.show()
125        plot_temperature_vs_time_with_controller(C0, P * 2, c, m, T0, Tenv, k,
126        R, h, sigma)
127        plot_temperature_vs_time_with_controller(C0, P, c / 2, m, T0, Tenv, k,
128        R, h, sigma)
129        plot_temperature_vs_time_with_controller(C0, P, c, m, T0, Tenv, k, R,
130        h, sigma)
131
132    if __name__ == '__main__':
133        main()

```

Класс SolderingIron возвращает массив значений численного решения.

4.3 Модели без терморегулятора

Построим решения нескольких моделей для нагревателя без использования терморегулятора.

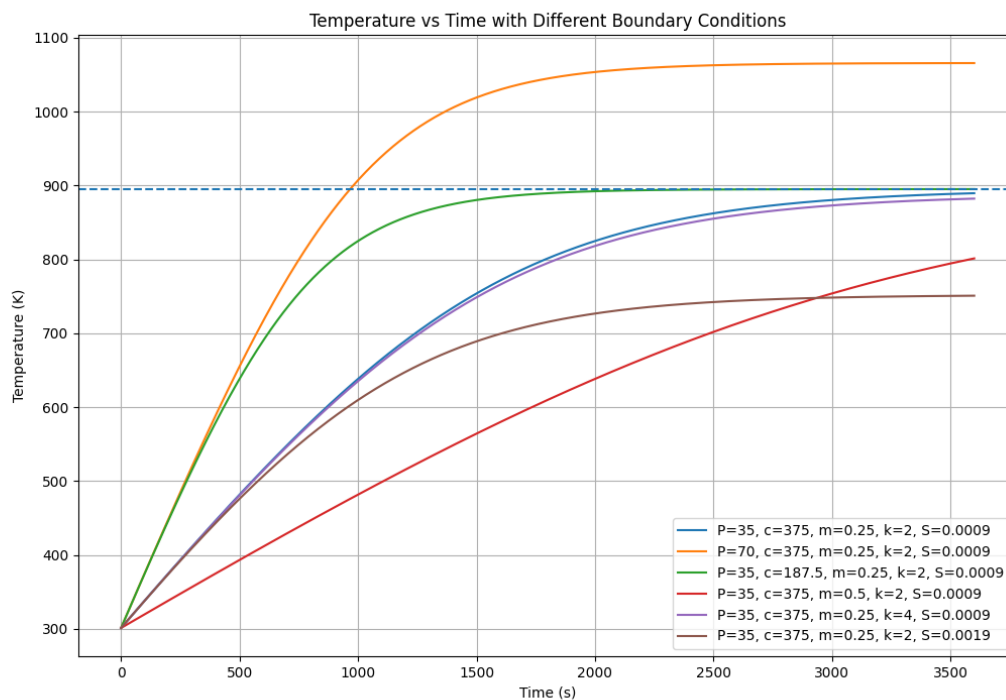


Рисунок 1 — Моделирование для $T_0 = 301$.

На рис. 1 изображены графики решений при разных параметрах, которые указаны в легенде графика. Также отмечена теоретически предсказанная максимальная температура при параметрах из предыдущей главы $T = 895.53$.

Заметим, что два графика, у которых меняется только масса и удельная теплоёмкость, возрастают до предсказанной точки равновесия.

Остальные же отличаются в параметрах, влияющих на точку равновесия. Для них, это значение тоже можно увидеть.

4.4 Модели с терморегулятором

Построим решения уравнения с терморегулятором для моделей с разными параметрами. Каждый рисунок - изображение решения

уравнения 8 с одним набором параметров, но разными минимальными и максимальными температурами, отмеченными пунктиром.

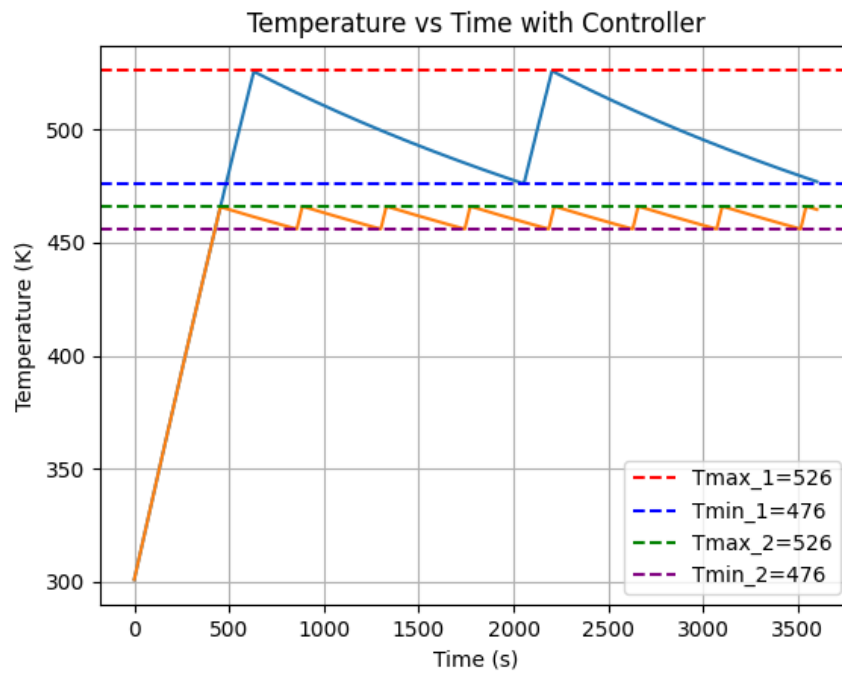


Рисунок 2 — Моделирование для

$$P = 35 \text{ Вт}, m = 0.25 \text{ кг}, c = 375 \frac{\text{Дж}}{\text{кг} \cdot \text{К}}, k = 2, S = 0.00094 \text{ м}^2, T_{env} = 296 \text{ К}$$

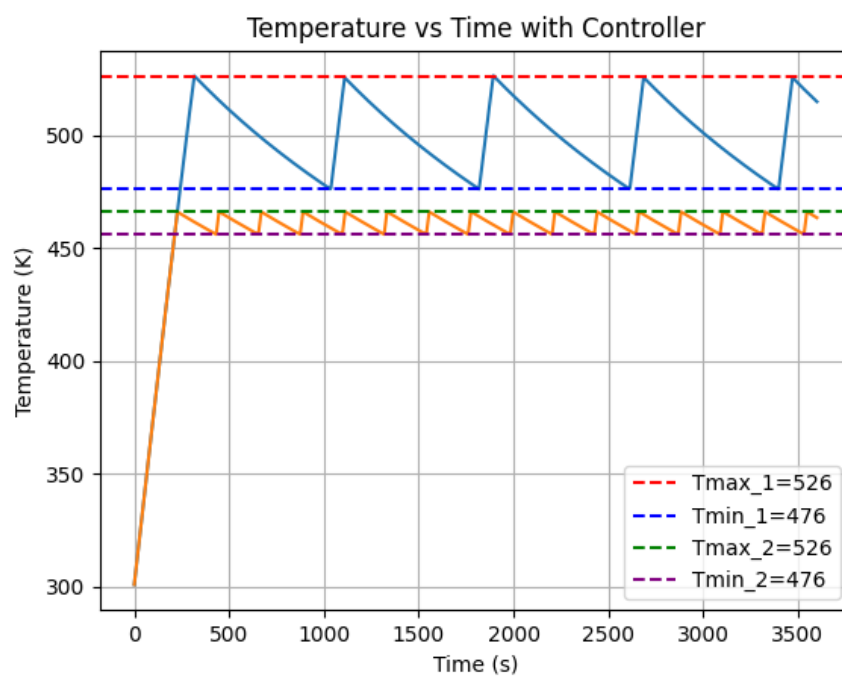


Рисунок 3 — Моделирование для

$$P = 35\text{Вт}, m = 0.25\text{кг}, c = 180 \frac{\text{Дж}}{\text{кг} \cdot \text{К}}, k = 2, S = 0.00094\text{м}^2, T_{env} = 296\text{К}$$

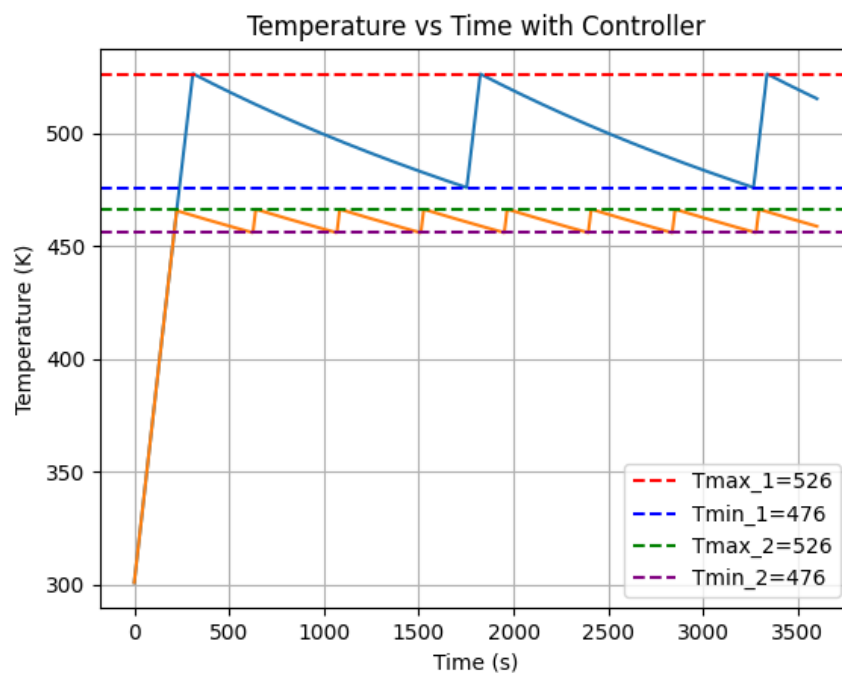


Рисунок 4 — Моделирование для

$$P = 70\text{Вт}, m = 0.25\text{кг}, c = 375 \frac{\text{Дж}}{\text{кг} \cdot \text{К}}, k = 2, S = 0.00094\text{м}^2, T_{env} = 296\text{К}$$

На рис. 2 - 4 показано, что при достижении максимальной температуры, нагревательный элемент отключается, благодаря работе контроллера. Аналогично, при достижении минимальной температуры нагреватель включается, и температура растёт.

5 Заключение

Была сформулирована математическая модель электрического нагревателя с терморегулятором и без него, представленная в виде дифференциального уравнения.

Проведён анализ стационарных решений, определены точки равновесия, удовлетворяющие уравнению.

Реализован численный метод решения задачи для различных параметров системы. Численные вычисления подтвердили результаты теоретического анализа, в частности, характер устойчивости найденных точек равновесия, определённый с помощью знака производной.

Список использованных источников

1. *Масалов, А. В.* Закон Стефана – Больцмана излучения / А. В. Масалов // Большая российская энциклопедия. — Москва, 2016.
2. *А., Черноуцан.* Уравнение теплового баланса / Черноуцан А. — Квант, номер 2, 42-47 edition. — 2015.
3. *Н.С., Бахвалов.* Численные методы / Бахвалов Н.С. — Наука, 1975.