Оглавление

Введение

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

Каждый из приборов имеет своих характеристики, и человеку, как пользователю, хочется знать, как быстро нагреется тот или иной прибор. Для этого можно потроить математическую модель, которая и 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 Построение математической модели

1.1 Модель без термо регулятора

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

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): $m_{env} >> m_H$.

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

$$\Delta Q = cm\Delta T, \tag{1}$$

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

Нагревательный прибор использует электрический ток для увеличения внутренней энергии:

$$\Delta Q_1 = P\Delta t, \tag{2}$$

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, характерынй для данной конструкции нагревательного прибора $\left(\frac{\text{Вт}}{\text{м}^2\text{К}}\right)$. С учётом этих явлений, внутренняя энергия будет изменяться на следующую величину:

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

Кроме этих явлений, согласно закону Стефана-Больцмана, любое тело, нагретое выше абсолютного нуля за единицу веремени на единицу площади излучает энергию равную $-\sigma T^4$, где $\sigma \approx 5.68 \cdot 10^{-8} \left(\frac{\text{Вт}}{\text{м}^2\cdot\text{К}^2}\right)$ - постоянная Стефана-Больцмана. Аналогично, излучение поступает из кружающей среды, равное σT_{env}^4. Тогда, измненение внутренней энергии, вызванного этим процессом, равно:

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

Суммируя все потоки энергии, получаем уравнение теплового баланса (см. 1, 2, 3, 4):

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

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

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

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

1.2 Модель с терморегулятором

Для предотвращения перегрева нагревателя, целесообразно установить терморегулятор, которые 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}).

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

Добавляя (7) в (6) получим:

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

2 Анализ математической модели

Сперва найдём точки равновесия уравнения (6).

$$\frac{dT}{dt} = \frac{P - kS(T - T_{env}) - \sigma S(T^4 - T_{env}^4)}{cm} \tag{9}$$

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:

$$\frac{dT}{dt} = \frac{P}{cm} > 0. \tag{10}$$

На основе (10) можно сделать вывод, что температура будет увеличиваться. Со временем, отрицательные слагаемые будут увеличиваться по мудулю, 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 соответсвующим ростом по модулю отрицательных слагаемых.

2.1 Вычисление точек покоя

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

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

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, использовав метод первого приближения. Обозначив правую часть дифференциального уравнения за R, получим:

$$\frac{dR}{dT} = -0.0017. \tag{11}$$

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

3 Вычислительные эксперименты

3.1 Алгоритм решения

Для численного решения дифференциального уравнения бдум использовать алгоритм Рунге-Кутты четвертого порядка. Результатом будет являться массив значений, каждое из которых будет являться решением дифференциального уравнения (6) с заданными параметрами.

3.2 Программа для ЭВМ

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

```python
import numpy as np
from matplotlib import pyplot as plt


class SolderingIron:
    def __init__(self, P, c, m, T0, Tenv, Tmax, Tmin, k, R, h, sigma):
        self.P = P
        self.c = c
        self.m = m
        self.T0 = T0
        self.Tenv = Tenv
        self.Tmax = Tmax
        self.Tmin = Tmin
        self.k = k
        self.S = 2 * np.pi * R * h
        self.sigma = sigma
        self.isOn = True

    def dTdt(self, T):
        return (self.P - self.k * self.S * (T - self.Tenv) - self.sigma * self.S * (T ** 4 - self.Tenv ** 4)) / (
                self.m * self.c)

    def I(self, T):
        if T > self.Tmax:
            self.isOn = False
        if T < self.Tmin:
            self.isOn = True

    def dTdt_with_controller(self, T):
```

```
            self.I(T)
        return (self.P * self.isOn - self.k * self.S * (T - self.Tenv) -
            self.sigma * self.S * (
                T ** 4 - self.Tenv ** 4)) / (self.m * self.c)

    def solve(self, t0, tn, n):
        h = (tn - t0) / n
        T = self.T0
        t_values, T_values = [t0], [self.T0]

        for _ in range(n):
            k1 = h * self.dTdt(T)
            k2 = h * self.dTdt(T + 0.5 * k1)
            k3 = h * self.dTdt(T + 0.5 * k2)
            k4 = h * self.dTdt(T + k3)
            T += (k1 + 2 * k2 + 2 * k3 + k4) / 6
            t_values.append(t_values[-1] + h)
            T_values.append(T)

        return t_values, T_values

    def solve_with_controller(self, t0, tn, n):
        h = (tn - t0) / n
        T = self.T0
        t_values, T_values = [t0], [self.T0]

        for _ in range(n):
            k1 = h * self.dTdt_with_controller(T)
            k2 = h * self.dTdt_with_controller(T + 0.5 * k1)
            k3 = h * self.dTdt_with_controller(T + 0.5 * k2)
            k4 = h * self.dTdt_with_controller(T + k3)
            T += (k1 + 2 * k2 + 2 * k3 + k4) / 6
            t_values.append(t_values[-1] + h)
            T_values.append(T)

        return t_values, T_values


def plot_temperature_vs_time(P, c, m, T0, Tenv, k, R, h, sigma):
    soldering_iron = SolderingIron(P, c, m, T0, Tenv, 0, 0, k, R, h, sigma)
    t_values, T_values = soldering_iron.solve(0, 3600, 1000)
    plt.plot(t_values, T_values, label=fr"P={P}, c={c}, m={m}, k={k},
        S={soldering_iron.S:.4f}")
    plt.xlabel('Time (s)')
    plt.ylabel('Temperature (K)')
    plt.title('Temperature vs Time with Different Boundary Conditions')
```

```
    plt.legend()
    plt.grid(True)


def make_plot_for_controller(soldering_iron1, soldering_iron2, ):
    t_values, T_values = soldering_iron1.solve_with_controller(0, 3600, 1000)
    plt.plot(t_values, T_values)
    plt.axhline(y=soldering_iron1.Tmax, linestyle='--', color='red', label=f'Tmax_1={soldering_iron1.Tmax}')
    plt.axhline(y=soldering_iron1.Tmin, linestyle='--', color='blue', label=f'Tmin_1={soldering_iron1.Tmin}')

    t_values, T_values = soldering_iron2.solve_with_controller(0, 3600, 1000)
    plt.plot(t_values, T_values)
    plt.axhline(y=soldering_iron2.Tmax, linestyle='--', color='green', label=f'Tmax_2={soldering_iron1.Tmax}')
    plt.axhline(y=soldering_iron2.Tmin, linestyle='--', color='purple', label=f'Tmin_2={soldering_iron1.Tmin}')


def plot_temperature_vs_time_with_controller(C0, P, c, m, T0, Tenv, k, R, h, sigma):
    soldering_iron_1 = SolderingIron(P, c, m, T0, Tenv, 250 + C0, 200 + C0, k, R, h, sigma)
    soldering_iron_2 = SolderingIron(P, c, m, T0, Tenv, 190 + C0, 180 + C0, k, R, h, sigma)

    make_plot_for_controller(soldering_iron_1, soldering_iron_2)
    plt.xlabel('Time (s)')
    plt.ylabel('Temperature (K)')
    plt.title('Temperature vs Time with Controller')
    plt.legend()
    plt.grid(True)
    plt.show()


def main():
    C0 = 276
    P = 35
    c = 375
    m = 0.25
    T0 = 25.0 + C0
    Tenv = 25.0 + C0
    k = 2
```

```
    R = 0.003
    h = 0.05
    sigma = 5.67e-8

    plt.figure(figsize=(12, 8))
    plot_temperature_vs_time(P, c, m, T0, Tenv, k, R, h, sigma)
    plot_temperature_vs_time(P * 2, c, m, T0, Tenv, k, R, h, sigma)
    plot_temperature_vs_time(P, c / 2, m, T0, Tenv, k, R, h, sigma)
    plot_temperature_vs_time(P, c, m * 2, T0, Tenv, k, R, h, sigma)
    plot_temperature_vs_time(P, c, m, T0, Tenv, k * 2, R, h, sigma)
    plot_temperature_vs_time(P, c, m, T0, Tenv, k, R, h * 2, sigma)
    plt.axhline(y=895.53, linestyle='--')

    plt.show()
    plot_temperature_vs_time_with_controller(C0, P * 2, c, m, T0, Tenv, k,
        R, h, sigma)
    plot_temperature_vs_time_with_controller(C0, P, c / 2, m, T0, Tenv, k,
        R, h, sigma)
    plot_temperature_vs_time_with_controller(C0, P, c, m, T0, Tenv, k, R,
        h, sigma)


if __name__ == '__main__':
    main()
```

Класс SolderingIron возвращает массив значений численного решения.

3.3 Модели без терморегулятора

Построим решения нескольких моделей для нагревателя без использования терморегулятора.

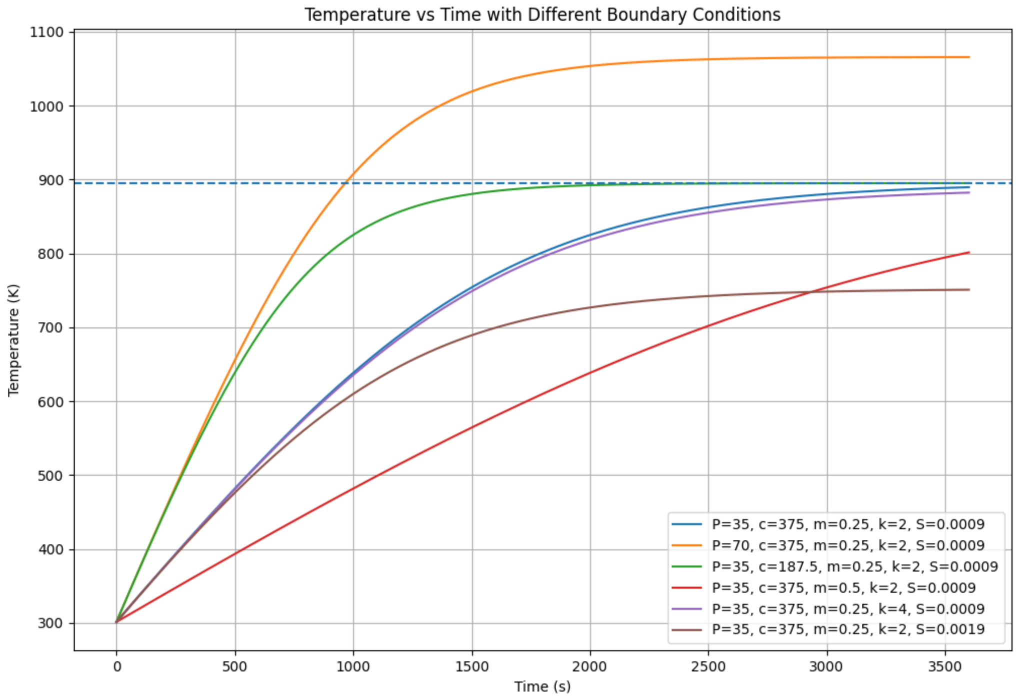

Рисунок 1 — Моделирование для $T_0 = 301$.

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.

3.4 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 (изображены пунктиром).

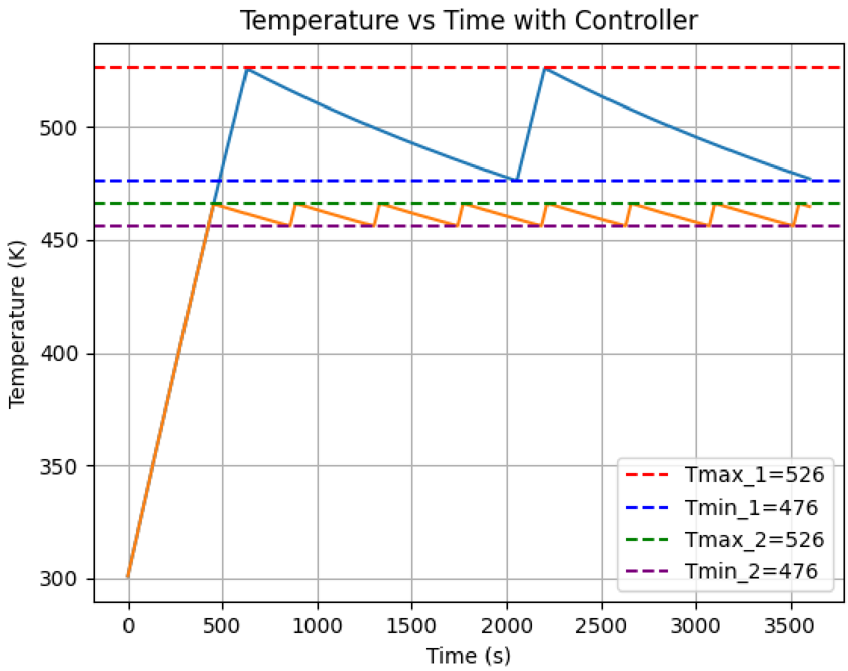

Рисунок 2 — Моделирование для

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

.

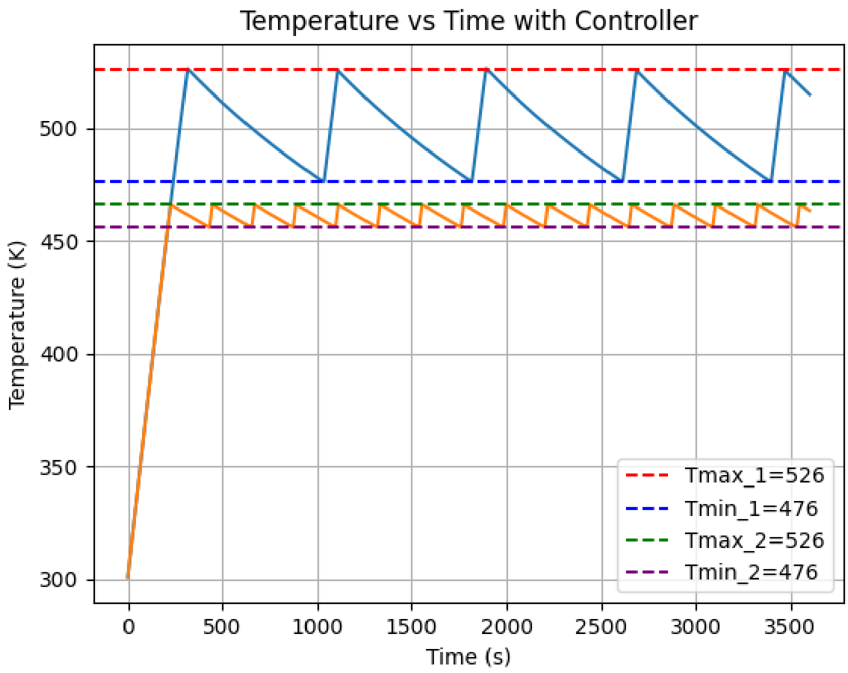

Рисунок 3 — Моделирование для

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

.

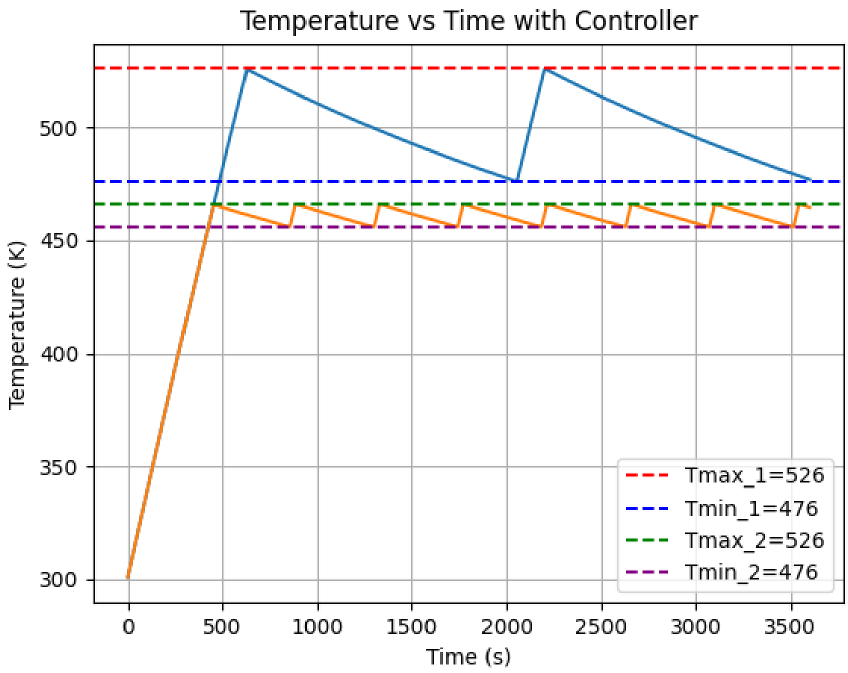

Рисунок 4 — Моделирование для

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

.

На рисунках (2, 3, 4) показано, что при достижении максимальной температуры, нагревательный элемент отключается, путём работы контроллера. Аналогично, при достижении минимальной температуры нагреватель включается, и температура растёт.

4 Заключение

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

Проведен 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, определенный с помощью знака производной.