Тестовое задание

Канюкова София Сергеевна
МФТИ, 3 курс
+7(995)788-19-17, kaniukova.ss@phystech.edu

Постановка задачи

Дан идеальный математический маятник – груз массой m = 1.0, прикрепленный к шарниру невесомым нерастяжимым стержнем длиной L = 5.0. На маятник действует сила сопротивления стержня T и сила тяжести mg(t), где g(t) = 9.81 + 0.05sin(2πt) – переменное ускорение свободного падения, t – время. Пренебрегая трением в шарнире и сопротивлением воздуха, получаем систему уравнений, описывающую движение груза в декартовой системе координат:

$$\begin{cases} m\ddot{x} = -\frac{x}{L}T \\ m\ddot{y} = -\frac{y}{L}T - mg \\ g = 9.81 + 0.05\sin(2\pi t) \\ x^2 + y^2 = L^2 \end{cases}$$

Начальные условия:

$$\begin{cases} x(0) = 3, y(0) = -4 \\ x'^{(0)} = -1, y'(0) = -1 \end{cases}$$

Аналитическое решение

Решим уравнения сначала аналитически, чтобы получить базу для сравнения с численным решением:

решение задачи Коши без НУ:

НУ:

$$\ddot{x} + \frac{T}{mL}x = 0 \longrightarrow x(t) = C_1 \cos\left(t\sqrt{\frac{T}{Lm}}\right) + C_2 \sin\left(t\sqrt{\frac{T}{Lm}}\right) \longrightarrow \begin{cases} x(0) = 3 \\ x'(0) = -1 \end{cases}$$

решение задачи Коши с НУ:

$$\longrightarrow \quad \boxed{x(t) = 3\cos\left(t\sqrt{\frac{T}{Lm}}\right) - \sqrt{\frac{Lm}{T}}\sin\left(t\sqrt{\frac{T}{Lm}}\right)}$$

Аналитическое решение

Решим уравнения сначала аналитически, чтобы получить базу для сравнения с численным решением:

$$\ddot{y} + \frac{T}{mL} y + 9.81 + 0.05\sin(2\pi t) = 0 \quad \longrightarrow$$

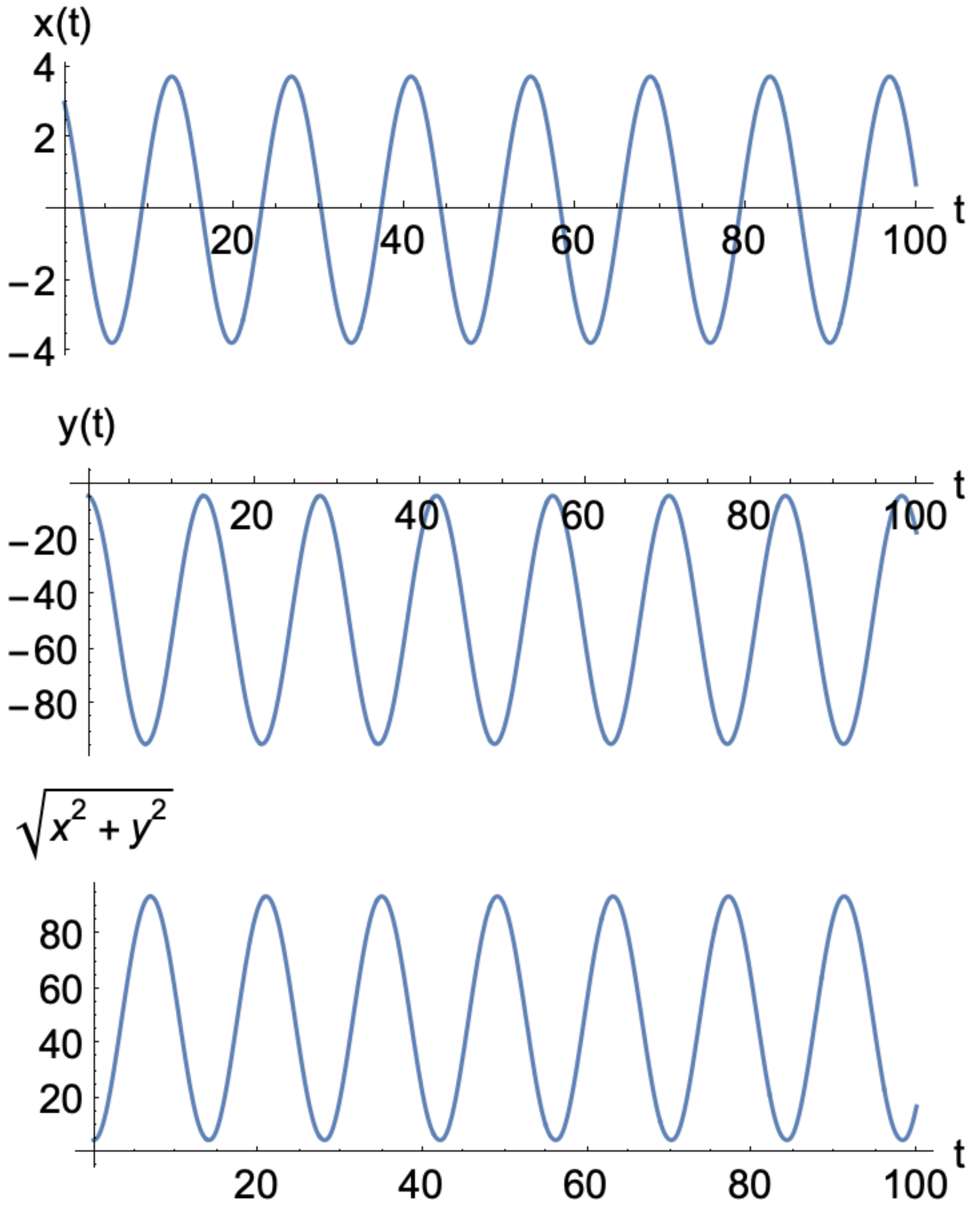
решение задачи Коши без НУ:

$$y(t) = C_1 \cos\left(t\sqrt{\frac{T}{Lm}}\right) + C_2 \sin\left(t\sqrt{\frac{T}{Lm}}\right) - \frac{9.81Lm}{T} + \frac{0.00126651Lm}{Lm - 0.0253303T}\sin(6.28319t) \quad \longrightarrow$$

решение задачи Коши с НУ:

$$y(t) = \frac{(-387.283L^2m^2 + 167.724LmT - 4T^2)\cos\left(t\sqrt{\frac{T}{Lm}}\right)}{T(-39.4784Lm + T)} + \frac{Lm(387.283Lm - 9.81T - 0.05T - 0.05T\sin(6.28319t)) + \sqrt{Lm}(39.7926Lm - T)\sin\left(t\sqrt{\frac{T}{Lm}}\right)}{\sqrt{T}(-39.4784Lm + T)}$$

Аналитическое решение

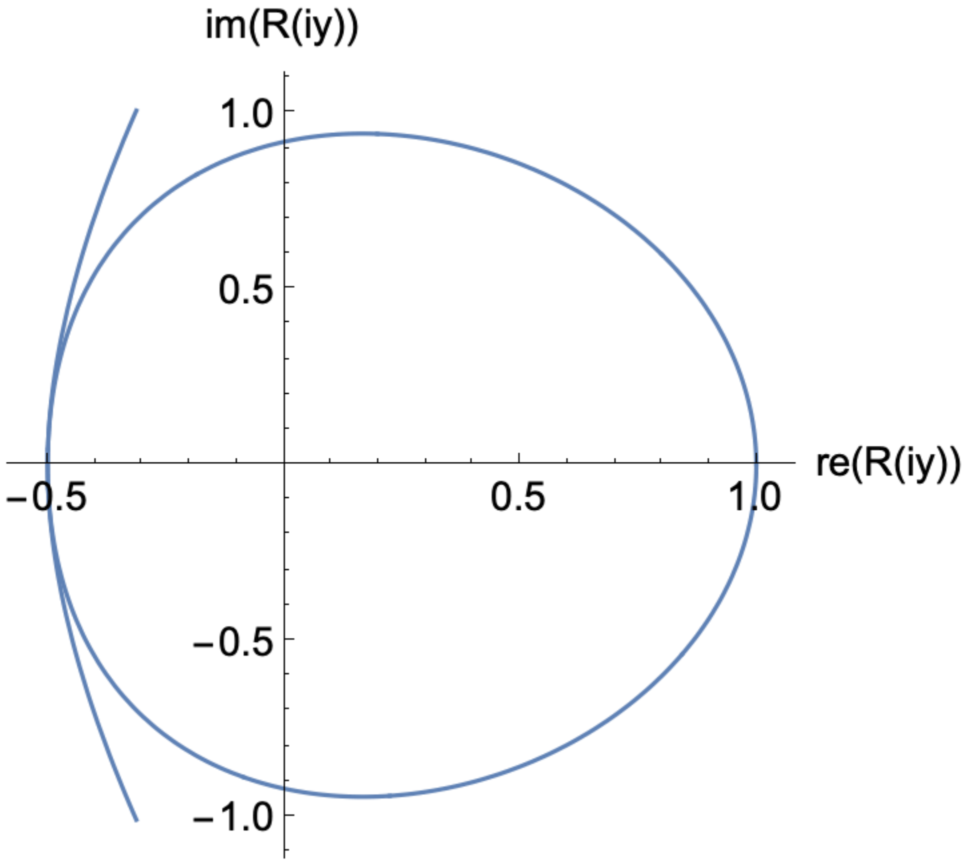


В дальнейшем мы будем сравнивать численное решение с аналитическим решением, чтобы посмотреть эволюцию нормы ошибки

Численное решение

Для численного решения был выбран метод Рунге-Кутты 4-го порядка аппроксимации. Таблица Бутчера этого метода выглядит следующим образом:

$$
\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array}
$$



Функция устойчивости метода: $R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}$

$R(iy) < 1$ → устойчиво → $y \in (-2.85, 2.85)$

На графике изображена граница области строгой устойчивости

Реализация

Необходимые библиотеки:

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

Реализация

Разбиваем исходную систему на набор линейных уравнений

```
def equations(t, y, m, L, T):
    x, u, y, v = y # Распаковка вектора состояния y на координаты и скорости
    g = 9.81 + 5 *  np.sin(2 * np.pi * t)
    dxdt = u # Производная координаты по времени равна скорости
    dudt = -x * T / (m * L) # Уравнение для производной скорости по времени
    dydt = v # Производная координаты по времени равна скорости
    dvdt = -y * T / (m * L) - g # Уравнение для производной скорости по времени
    return [dxdt, dudt, dydt, dvdt]
```

Реализация

Параметры задачи из условия тестового задания:

```
m = 1.0  # Масса
L = 5.0  # Длина
T = 1.0  # Сила сопротивления стержня
```

Реализация

Начальные условия из формулировки тестового задания

```
                                                # x(0) = 3, y(0) = -4, x'(0) = -1, y'(0) = -1
initial_conditions = [3, -4, -1, -1]
t_span = (0, 100)  # Диапазон по времени
dt = 0.001         # Шаг временной сетки
```

Реализация

Решение системы уравнений метода Рунге-Кутты

```
t_values = np.arange(t_span[0], t_span[1], dt)
y_values = np.zeros((len(t_values), len(initial_conditions)))

y_values[0] = initial_conditions    # Установка начальных условий

for i in range(1, len(t_values)):
        k1 = equations(t_values[i-1], y_values[i-1], m, L, T)
        k2 = equations(t_values[i-1] + dt/2, y_values[i-1] + dt/2*np.array(k1), m, L, T)
        k3 = equations(t_values[i-1] + dt/2, y_values[i-1] + dt/2*np.array(k2), m, L, T)
        k4 = equations(t_values[i-1] + dt, y_values[i-1] + dt*np.array(k3), m, L, T)
        y_values[i] = y_values[i-1] + dt/6 * (np.array(k1) + 2*np.array(k2) + 2*np.array(k3) +
np.array(k4))

                    # Расчет коэффициентов метода Рунге-Кутты
```

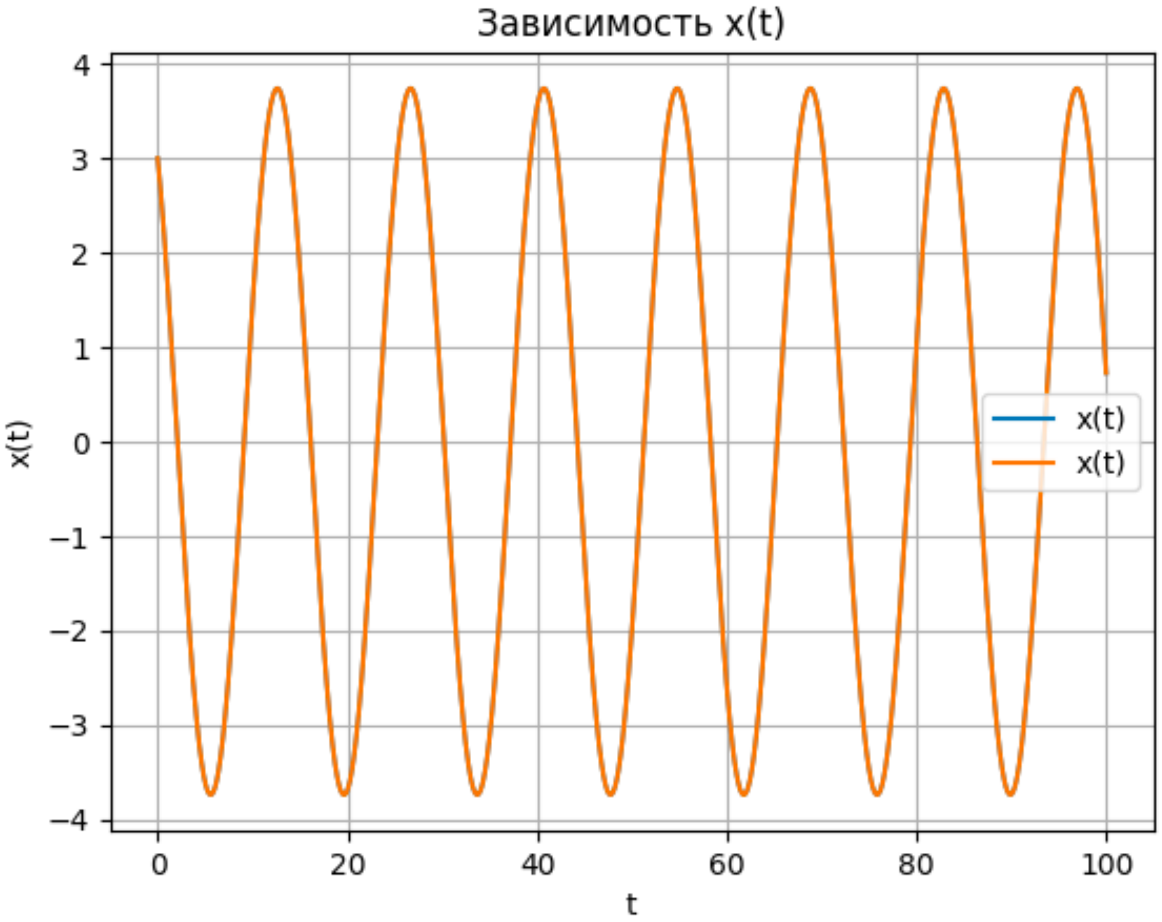
Реализация

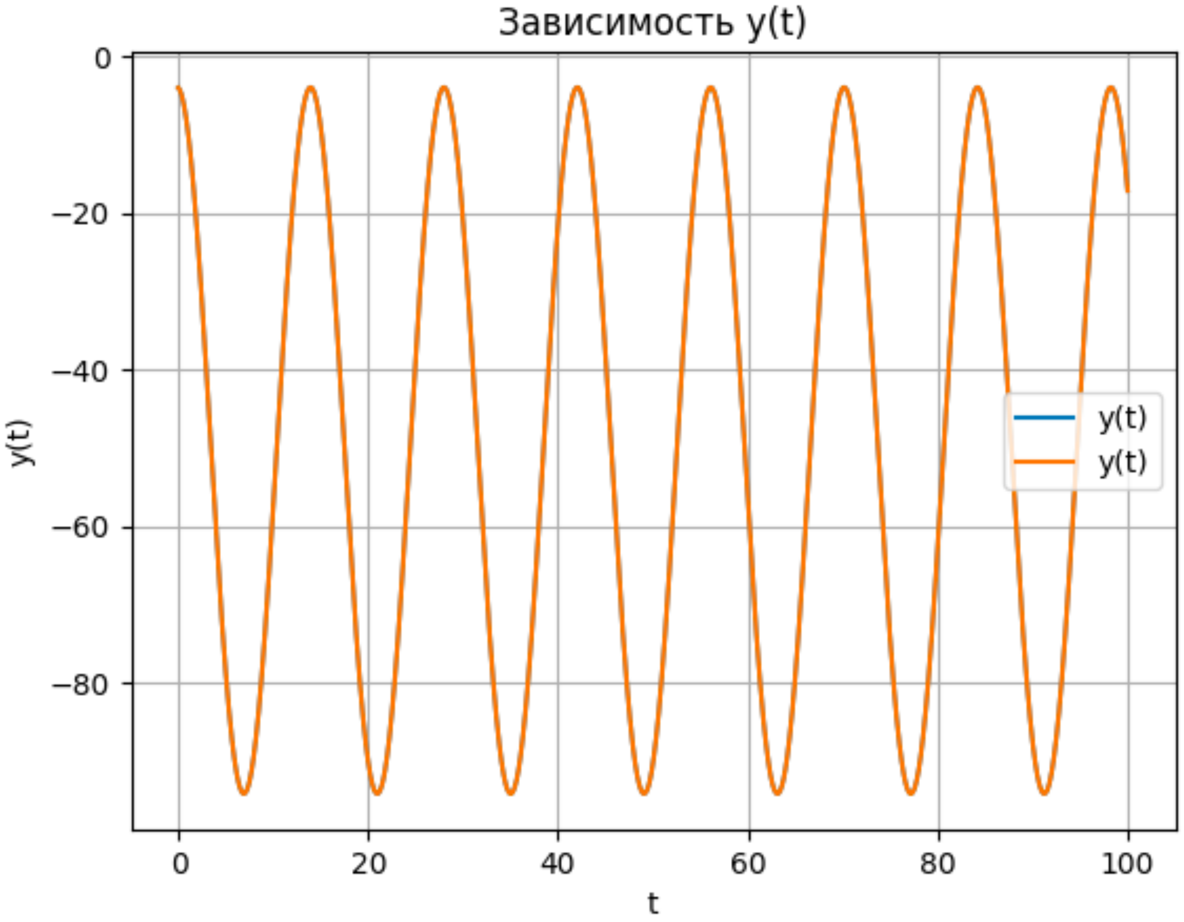
Построение графика

```
plt.figure()
plt.plot(t_values, np.sqrt(y_values[:, 1]**2 + y_values[:, 2]**2))
plt.xlabel('t')
plt.ylabel('sqrt(x^2 + y^2)')
plt.title('Зависимость sqrt(x^2 + y^2)')
plt.grid(True)
plt.legend()
plt.show()
```

Результаты

Графики x(t) и y(t) вместе с аналитическим решением. Пока что видно, что решения идентичны.

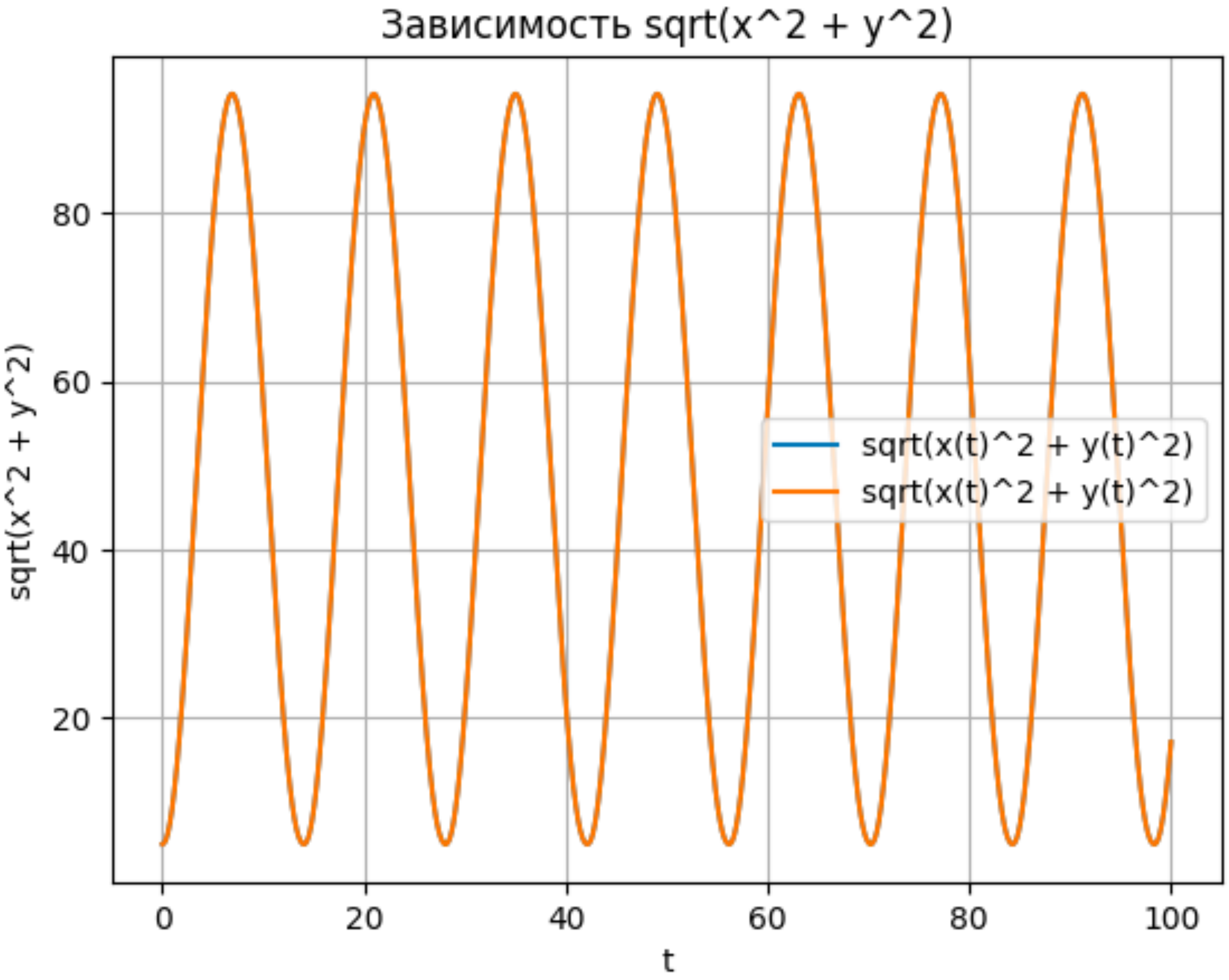




После демонстрации результатов приведем эволюцию ошибки по мере уменьшения шага

Результаты

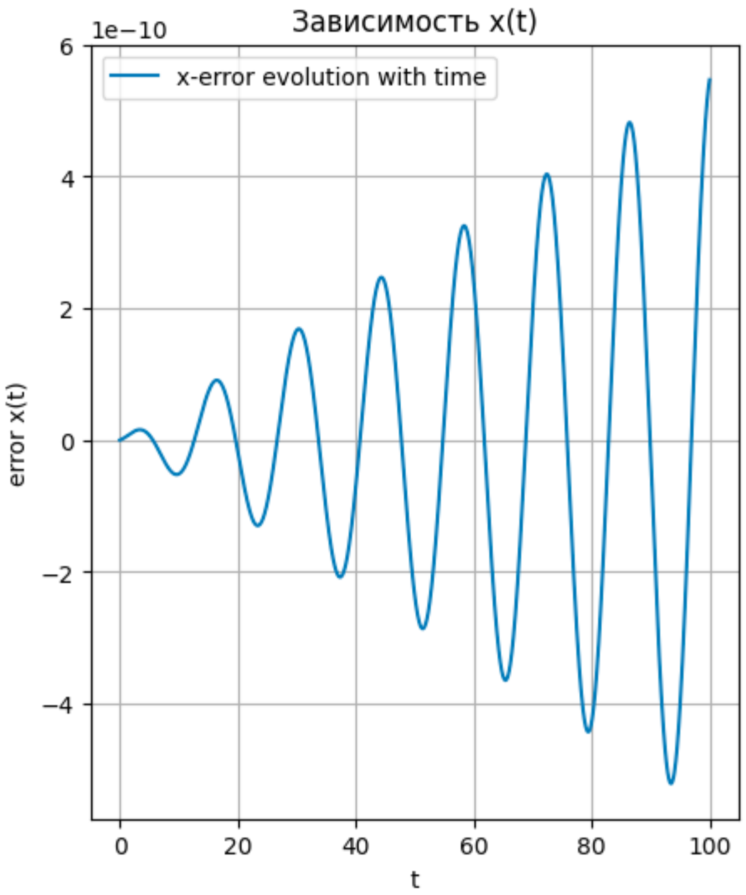
График $\sqrt{x^2 + y^2}$, наблюдаем то же самое визуальное сходство

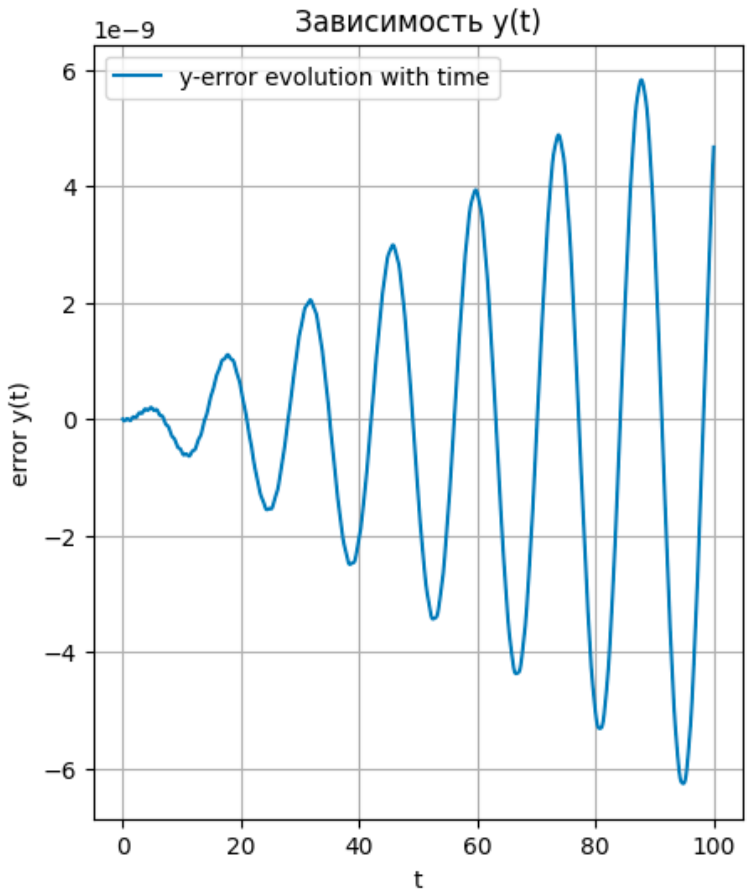


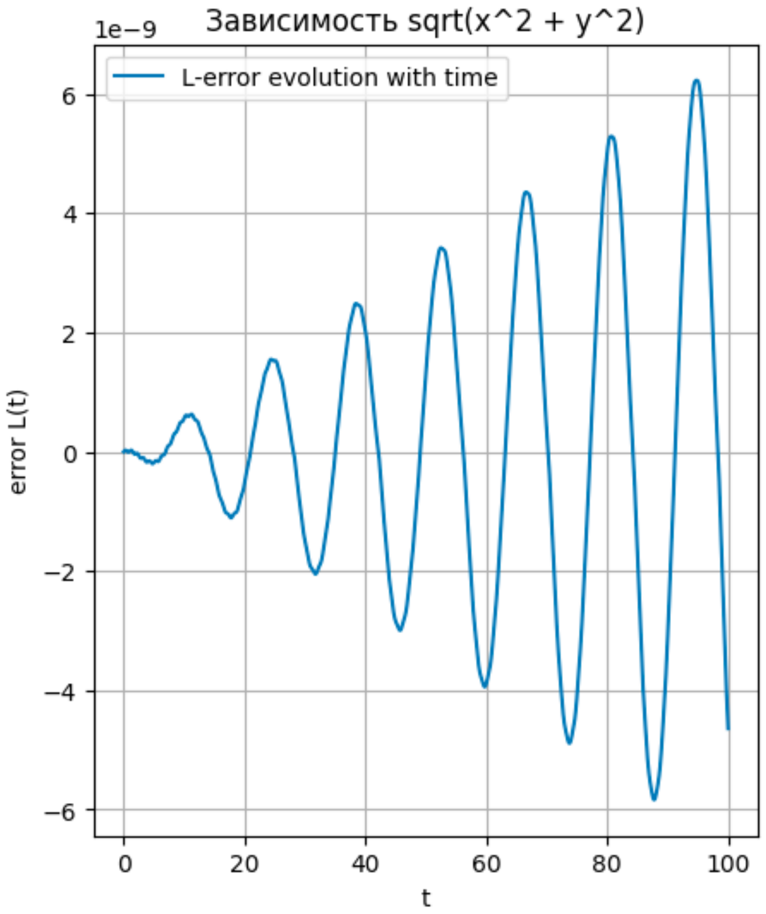
После демонстрации результатов приведем эволюцию ошибки по мере уменьшения шага

Ошибка в ходе решения

Ошибка решения рассчитывалась как разность численного и аналитического значения в моменте времени



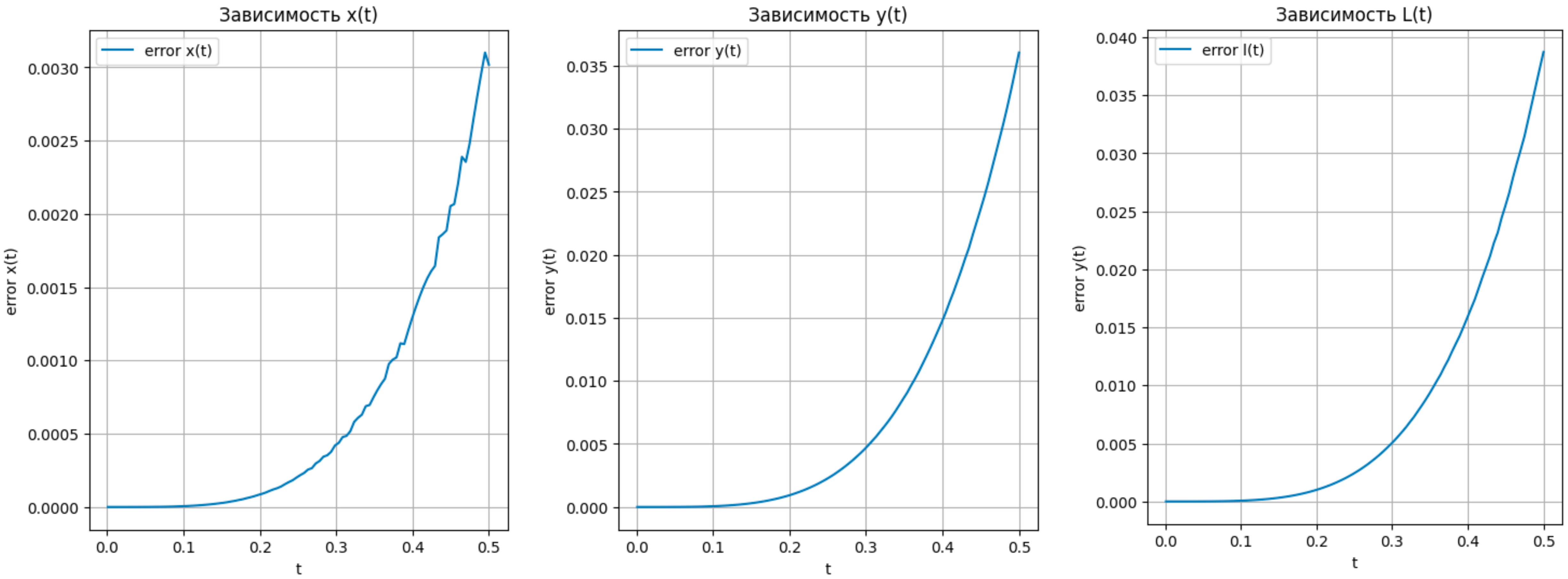




max_error_x = 5.466701535894458e−10 max_error_y = 5.822883508699306e−09 max_error_L = 6.226898108252499e−09

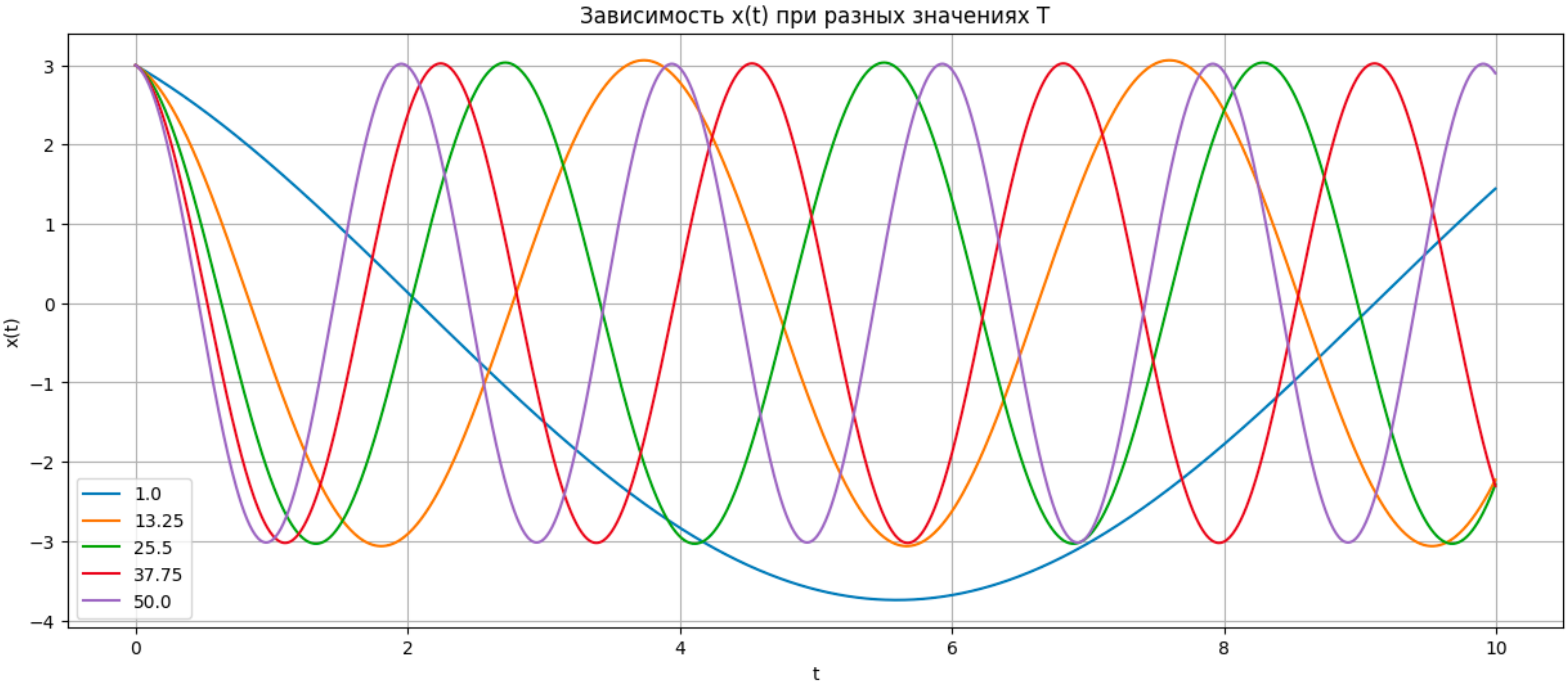
Эволюция ошибки от dt

Ошибка решения в зависимости от шага временной сетки рассчитывалась по первой норме (максимальное значение ошибки при текущем шаге на промежутке от 0 до 100 по t)



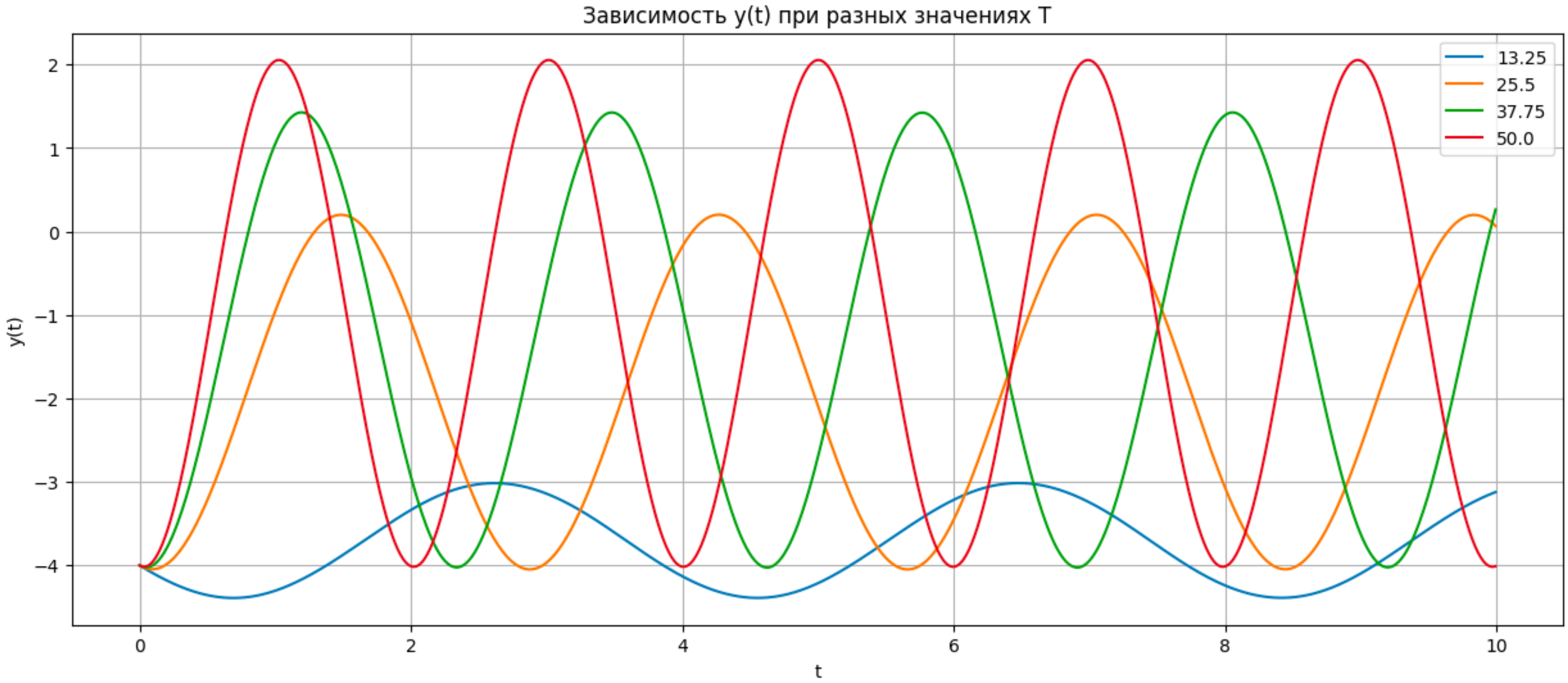
Для реализации этого был написан солвер ДУ в зависимости от шага сетки, который в последствии возвращал максимальную ошибку на всем временном промежутке

Исследование поведения x(t) от T



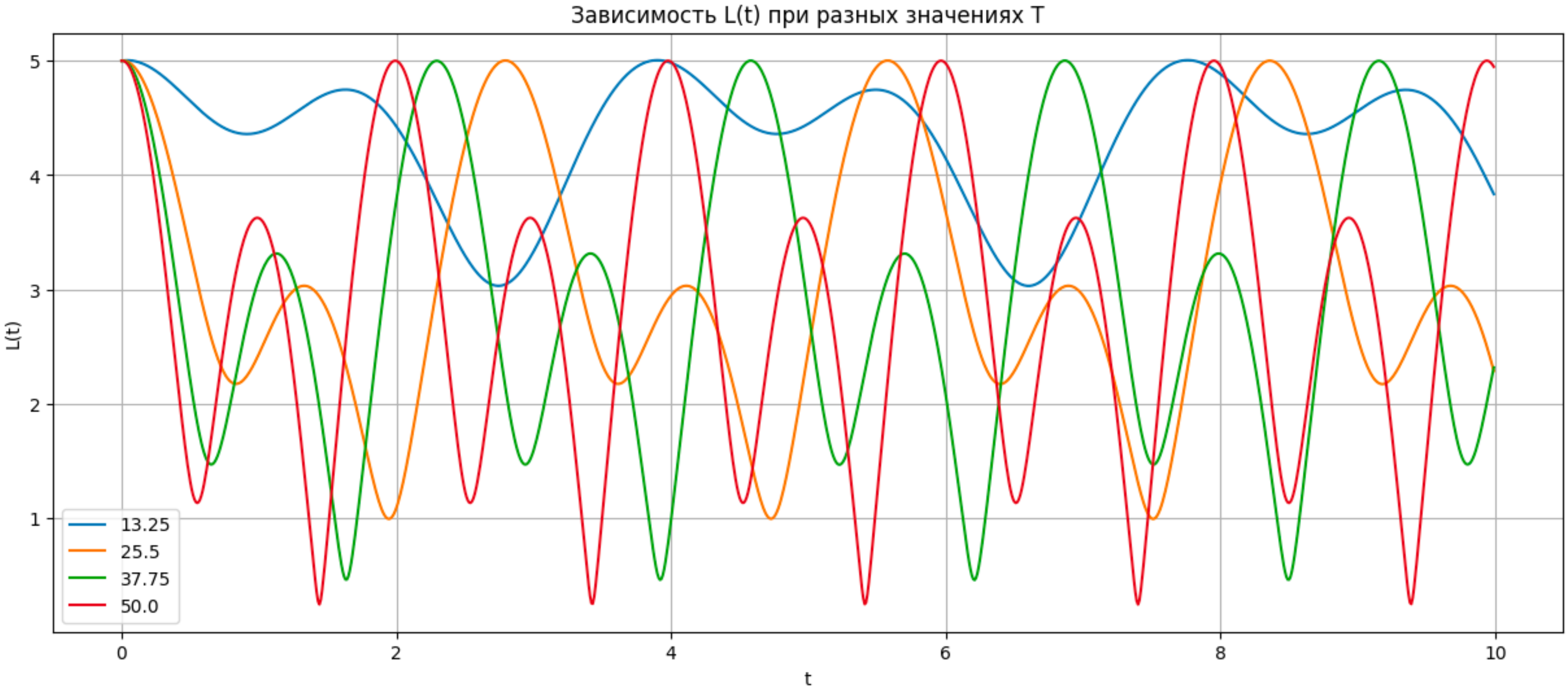
Используем отдельный солвер для зависимостей от параметров. У траектории увеличивается частота, как и ожидается из аналитического решения

Исследование поведения y(t) от T



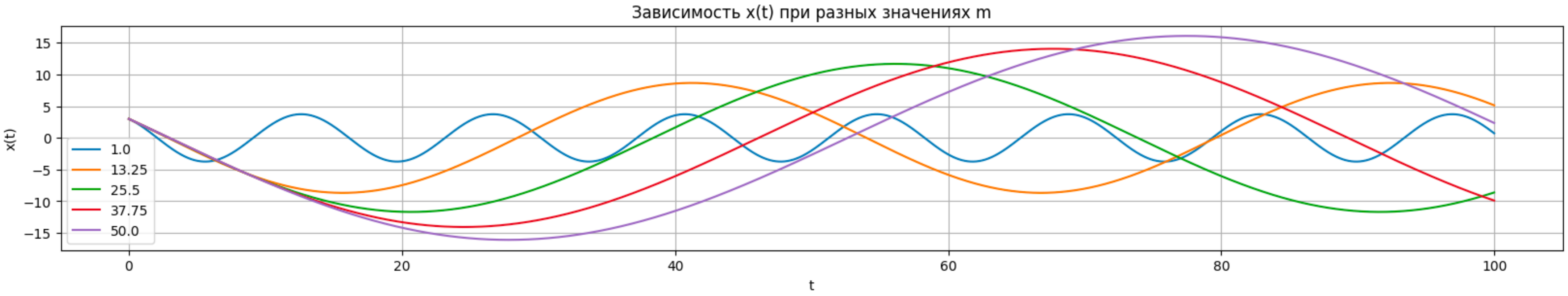
Используем отдельный солвер для зависимостей от параметров. Траектория ведет себя уже не так предсказуемо, как для x(t), что следует из неоднородности ДУ, ее описывающего.

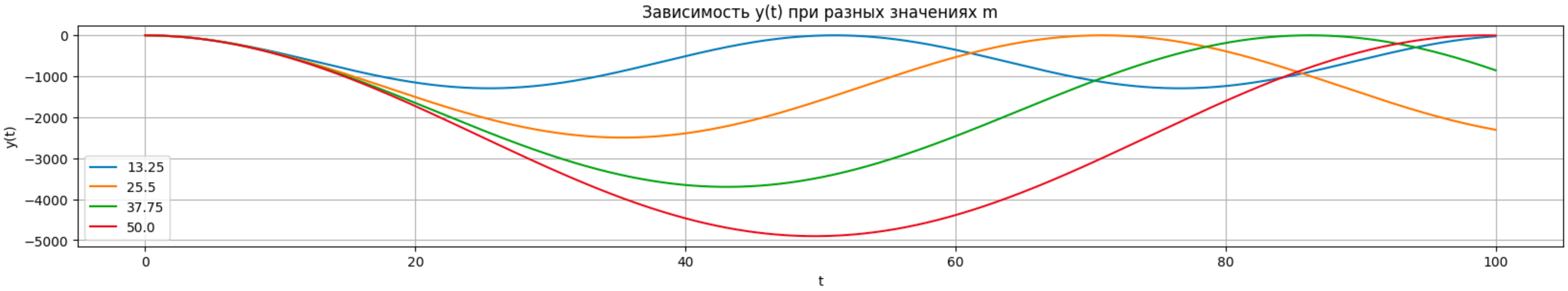
Исследование поведения L(t) от T

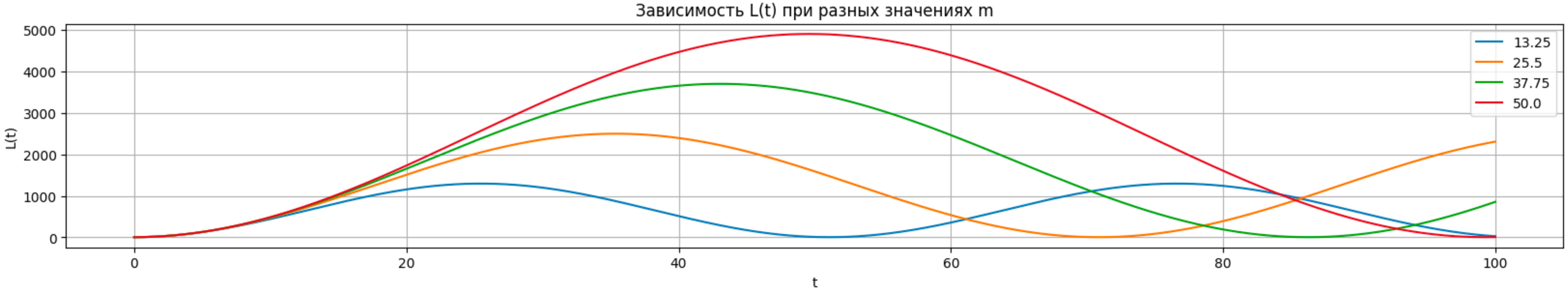


Используем отдельный солвер для зависимостей от параметров. Траектория ведет себя уже не так предсказуемо, как для x(t), что следует из неоднородности ДУ, ее описывающего.

Исследование поведения траекторий от m







Исследование поведения траекторий от L