МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»
(Университет ИТМО)

Факультет систем управления и робототехники

ОТЧЕТ
по лабораторной работе №1
по дисциплине
«Имитационное моделирование робототехнических систем»

Студент: Петрекеев К.С.

Группа № R4133c 505881

Преподаватель: Ракшин Егор Александрович

Санкт-Петербург 2025

Численное решение

Уравнение динамики представлено в виде:

$$a \cdot x'' + b \cdot x' + c \cdot x = d$$

$$-2.28\,x'' + 2.98\,x' + 7.58\,x = 9.11$$

Где параметры согласно индивидуальному варианту:

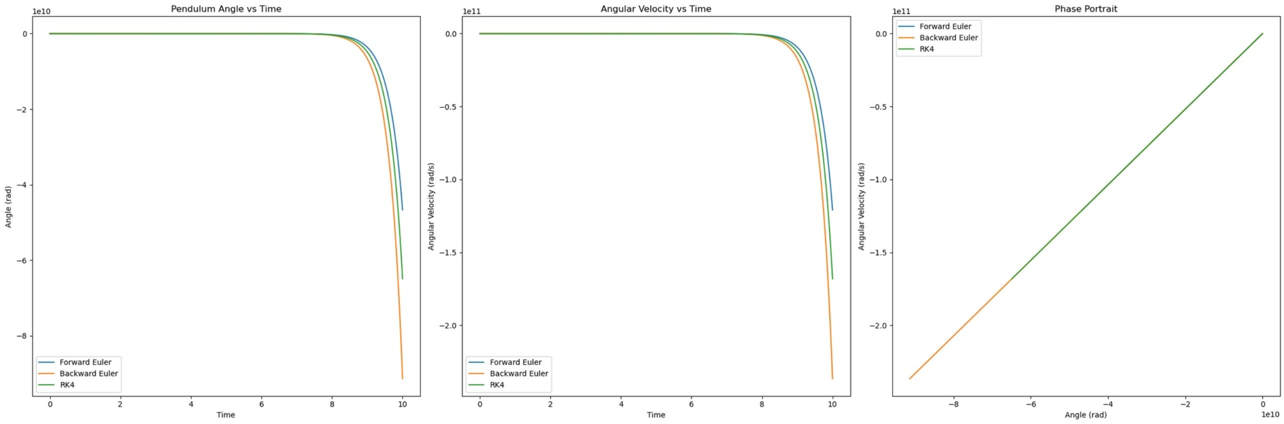
$$a = -2.28$$
$$b = 2.98$$
$$c = 7.58$$
$$d = 9.11$$

Код программы, решающий уравнение с использованием разных интеграторов:

```
a = -2.28
b = 2.98
c = 7.58
d = 9.11

def dynamics(y):
    dydt = np.array([y[1], (d - b * y[1] - c * y[0]) / (a)])
    return dydt

# Initial states
x0 = np.array([0.1, 0.0]) # [angle, angular_velocity]
Tf = 1.0
h = 0.01

# Forward Euler
x_fe, t_fe = forward_euler(dynamics, x0, Tf, h)

# Backward Euler
x_be, t_be = backward_euler(dynamics, x0, Tf, h)

# Runge-Kutta 4
x_rk4, t_rk4 = runge_kutta4(dynamics, x0, Tf, h)
```

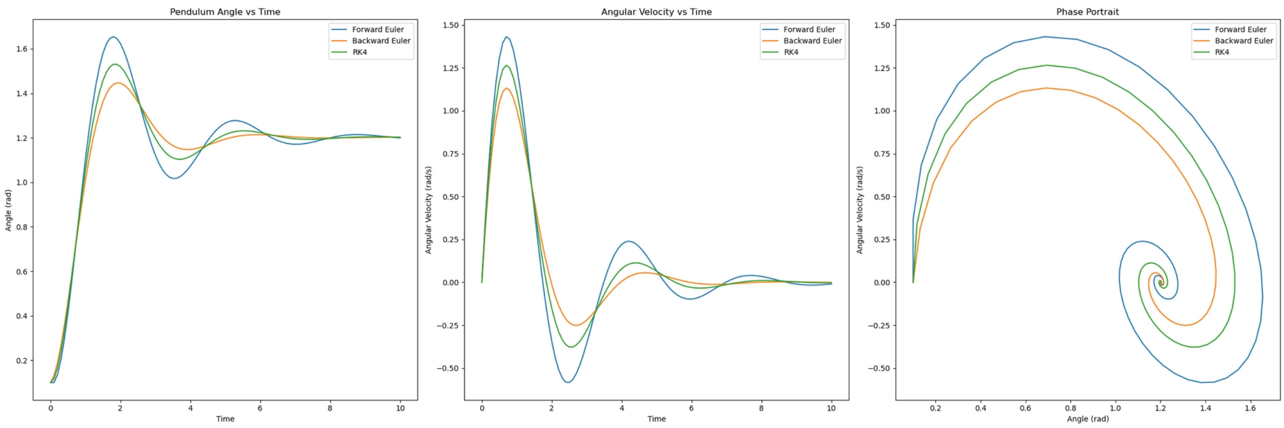
По заданным параметрам системы, а, точнее, отрицательному значению $a = -2.28$ можно сразу сделать вывод, что система может оказаться неустойчивой, то есть такая конфигурация делает уравнение нефизичным, так как в обычном представлении коэффициент должен быть положительным.

Тогда получены графики:



Как и предполагалось, система неустойчива. По полученным значениям однозначного вывода о качестве различных интеграторов сделать нельзя.

Однако, для того, чтобы убедиться в работоспособности программы и сделать вывод, можно *для примера* принять первый коэффициент положительным $a=2.28$. Тогда вывод программы будет:



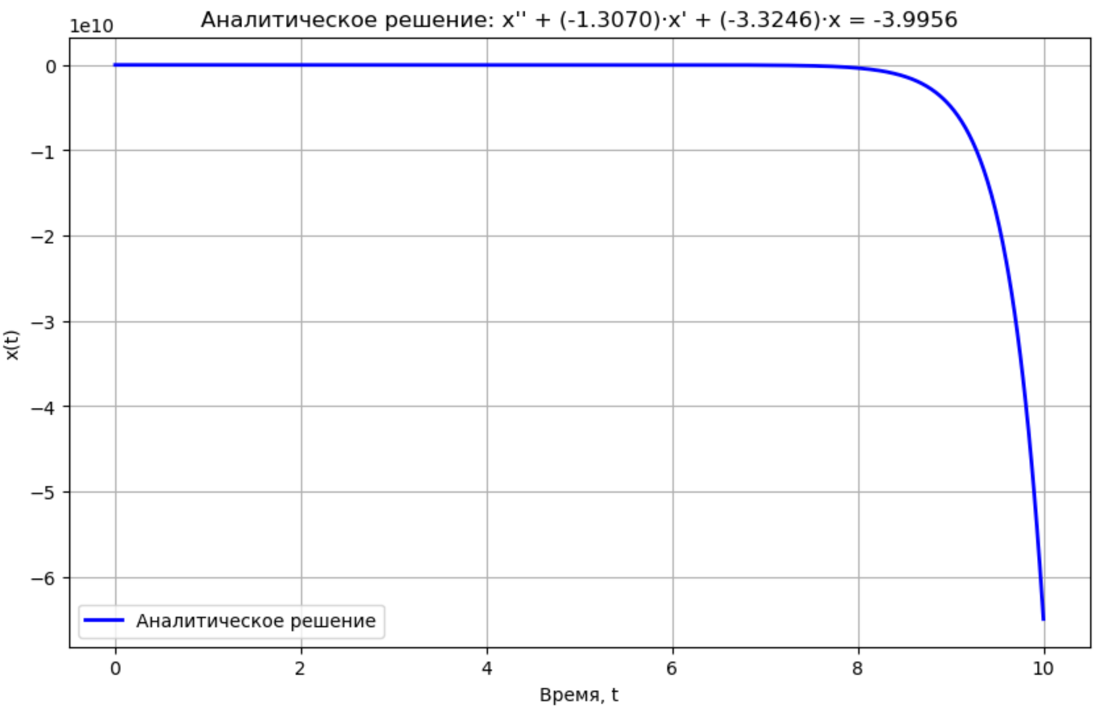
где при параметрах интегрирования $Tf=10.0\,;\,h=0.1$ можно увидеть, что явный Эйлер — наименее точный метод.

Аналитическое решение

Для решения аналитическим способом используем скрипт:

```
# λ² + B*λ + C = 0
D = B**2 - 4*C
lambda1 = (-B + np.sqrt(D)) / 2
lambda2 = (-B - np.sqrt(D)) / 2
x_particular = D / C

if D > 0:
    A_matrix = np.array([[1, 1], [lambda1, lambda2]])
    b_vector = np.array([x0 - x_particular, x0_dot])
    C1, C2 = np.linalg.solve(A_matrix, b_vector)
```

```
    def analytical_solution(t):
        return C1 * np.exp(lambda1 * t) + C2 * np.exp(lambda2 * t) + x_particular
elif D == 0:
    C1 = x0 - x_particular
    C2 = x0_dot - lambda1 * C1
    def analytical_solution(t):
        return (C1 + C2 * t) * np.exp(lambda1 * t) + x_particular
else:
    alpha = -B / 2
    beta = np.sqrt(-D) / 2
    C1 = x0 - x_particular
    C2 = (x0_dot - alpha * C1) / beta
    def analytical_solution(t):
        return np.exp(alpha * t) * (C1 * np.cos(beta * t) + C2 * np.sin(beta * t)) + x_particular

t = np.linspace(0, 10, 1000)
x_analytical = analytical_solution(t)
```



В сравнении видно, что численные методы выдают аналогичный аналитическому решению ответ.

Как и предполагалось, при корнях характеристического уравнения $\lambda 1 = 2.5904, \lambda 2 = -1.2834$, где один из корней положительный, система действительно оказывается неустойчивой.