

## Цель работы

1. Составить ОДУ для системы с массой, пружиной и демпфером по рисунку 1.

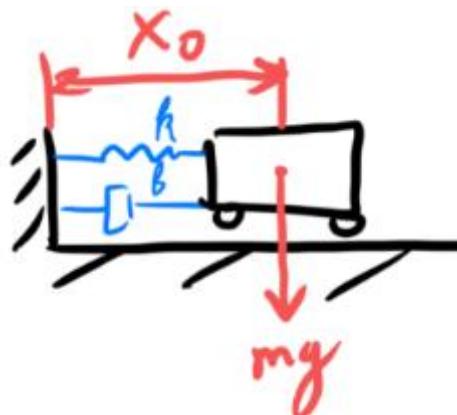


Рисунок 1. Вариант 2, система с массой, пружиной и демпфером

2. Решить составленное ОДУ аналитически.
3. Решить ОДУ в численном виде.
4. Сравнить результаты методов с аналитическим решением.

## Аналитическое решение ОДУ второго порядка

Для решения системы в аналитическом виде нужно составить лагранжиан системы:

$$\mathcal{L}(x, \dot{x}) = \mathcal{K}(x, \dot{x}) - \mathcal{P}(x) = \frac{1}{2}m\dot{x}^2 - \frac{1}{2}kx^2 \quad (1)$$

Уравнение Лагранжа выглядит следующим образом:

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{x}} \right) - \frac{\partial \mathcal{L}}{\partial x} = Q, \quad (2)$$

где:

$$\frac{\partial \mathcal{L}}{\partial \dot{x}} = m\dot{x}; \quad (3)$$

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{x}} \right) = m\ddot{x}; \quad (4)$$

$$\frac{\partial \mathcal{L}}{\partial x} = -kx; \quad (5)$$

$$Q = -b\dot{x}. \quad (6)$$

Подставив уравнения (3 – 6) в (2) получим:

$$m\ddot{x} + b\dot{x} + kx = 0 \quad (7)$$

Согласно данным таблицы принимаем следующие значения:  
 $m = 0.3, b = 0.005, k = 12.4.$

Решение будет производиться в общем виде. Составляется характеристический многочлен:

$$\begin{aligned} a \cdot \lambda^2 + b \cdot \lambda + c &= 0. \\ \lambda_{1,2} &= \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}. \\ \lambda_1 &= -0.00835 - 6.43i; \\ \lambda_2 &= -0.00835 + 6.43i; \end{aligned} \quad (8)$$

Так как нет действительных корней, то общее решение ОДУ выглядит следующим образом:

$$x = e^{\alpha t} (C_1 \cos(\beta x) + C_2 \sin(\beta x)), \quad (9)$$

где:  $C_1$  и  $C_2$  – постоянные величины,  $\alpha = -0.00835$ ,  $\beta = 6.43$ .

Нахождение значений коэффициентов  $C_1$  и  $C_2$  при начальных условиях:  $x_0 = 0.67, \dot{x}_0 = 0$ .

$$\begin{aligned} x(0) &= C_1 = x_0; \\ \dot{x}(0) &= \alpha C_1 + \beta C_2 = 0; \\ \begin{cases} C_1 = x_0 \\ C_2 = -\frac{\alpha C_1}{\beta} \end{cases} \end{aligned} \quad (10)$$

Подставим значения в (10) и получим:

$$\begin{cases} C_1 = 0.67 \\ C_2 = 0.0067 \end{cases}$$

Подставляя все полученные значения в (9), получим решение системы при заданных начальных условиях:

$$x = e^{-0.00835t} (0.67 \cdot \cos(6.43x) + 0.0067 \cdot \sin(6.43x)).$$

## Решение ОДУ второго порядка численными методами

В файле *Integrators.ipynb* функция *pendulum\_dynamics* была заменена на функцию *ode\_system*, реализующую систему первого порядка для уравнения (рис. 2). Также были добавлены функции аналитического решения координаты и её производной и вызовы трёх интеграторов для сравнения.

```
4  m = 0.3
5  k = 12.4
6  b = 0.005
7  x0_val = 0.67
8  v0_val = 0
9  Tf = 10.0
10 h = 0.01
11 t = np.arange(0, Tf + h, h)
12
13 def ode_system(x):
14     x1 = x[0]
15     x2 = x[1]
16     dx1 = x2
17     dx2 = (-b * x2 - k *x1)/m
18     return np.array([dx1, dx2])
```

Рисунок 2. функция *ode\_system*

Реализация аналитического решения представлена на рисунке 3.

```
64  p = b / m
65  q = k / m
66  s = 0
67  x_p = 0
68
69  D = p**2 - 4.0 * q
70
71 if D > 1e-12:
72
73     r1 = (-p + np.sqrt(D)) / 2.0
74     r2 = (-p - np.sqrt(D)) / 2.0
75
76     A = np.array([[1.0, 1.0],
77                   [r1, r2]])
78     b_ic = np.array([x0_val - x_p,
79                      v0_val])
80     C1, C2 = np.linalg.solve(A, b_ic)
81
82     x_an = C1 * np.exp(r1 * t) + C2 * np.exp(r2 * t) + x_p
83
84 elif abs(D) <= 1e-12:
85
86     r = -p / 2.0
87
88     C1 = x0_val - x_p
89     C2 = v0_val - r * C1
90
91     x_an = (C1 + C2 * t) * np.exp(r * t) + x_p
92
```

Рисунок 3. Аналитическое решение

Далее будут представлены графики решений. На рисунке 4 будет представлена зависимость координаты  $x(t)$  от времени, на рисунке 5 – зависимость скорости  $x'(t)$  от времени, а на рисунке 6 графически изображен фазовый портрет данной системы.

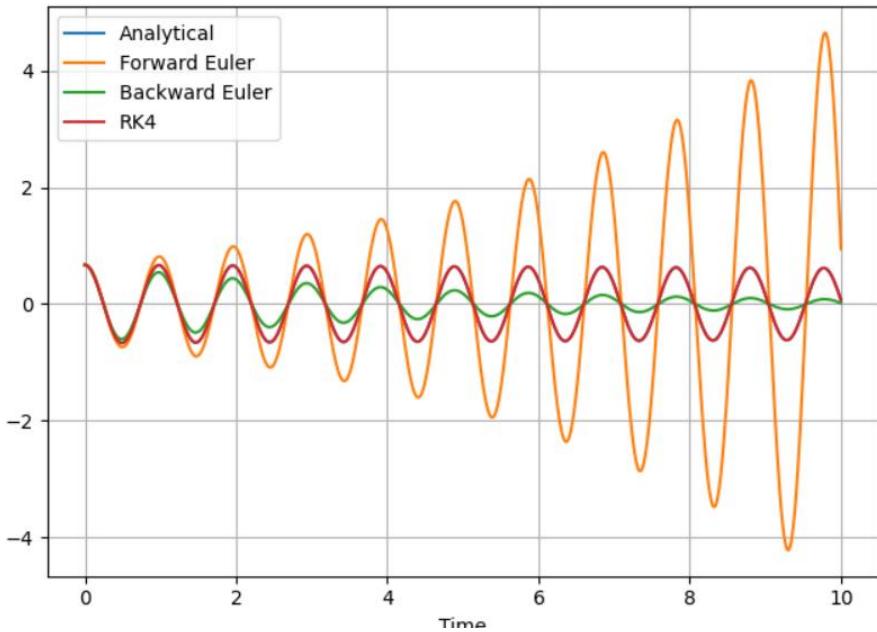


Рисунок 4. Зависимость координаты  $x(t)$  от времени с  $h = 0.01$

Из рисунка 4 видно, что аналитический метод совпадает с численным методом Рунге – Кутты (идентичен) при этом метод Явного Эйлера ошибочен, так как используется малый шаг (0.01). Изменим шаг до 0.001 и повторим запуск кода. Дальнейшие графики будут изображены с меньшим шагом.

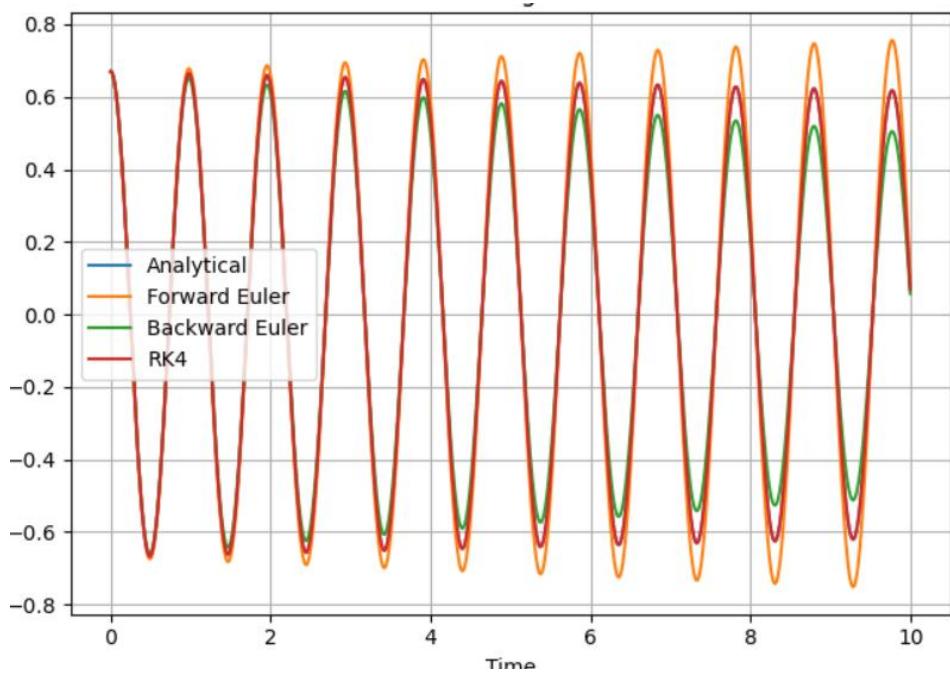


Рисунок 5. Зависимость координаты  $x(t)$  от времени с  $h = 0.001$

Из рисунка 5 также видно, что аналитический метод больше всего совпадает с численным методом Рунге – Кутты.

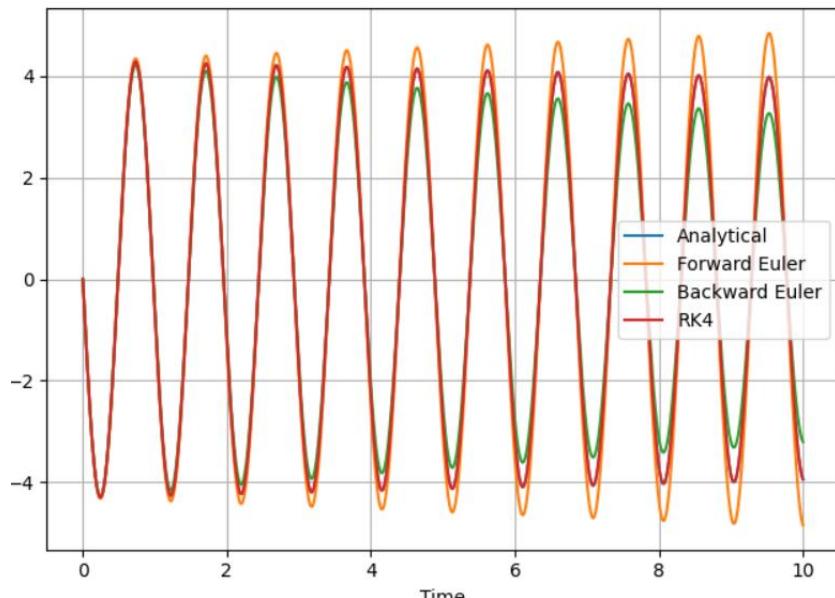


Рисунок 6. Зависимость скорости  $x'(t)$  от времени

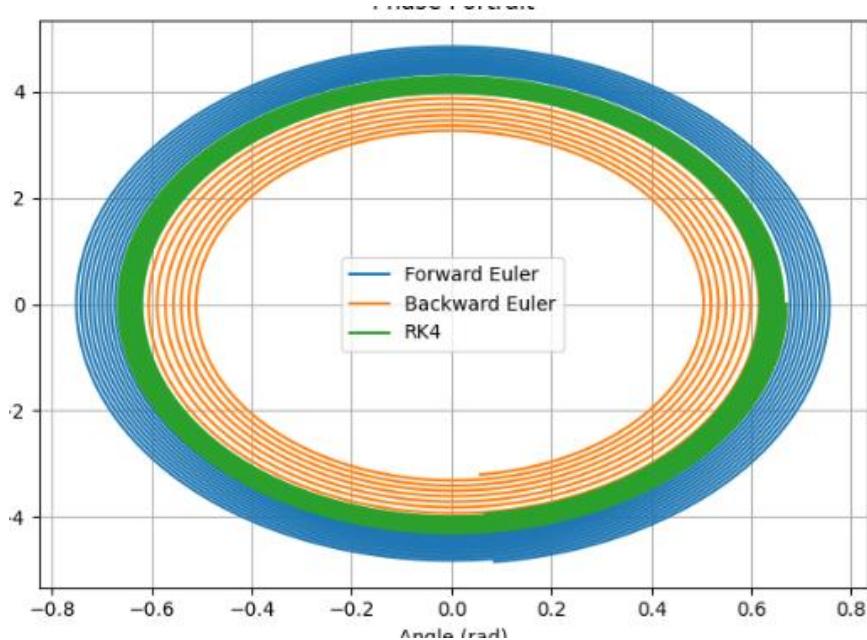


Рисунок 7. Фазовый портрет системы

### Вывод

В ходе выполнения данной практической работы можно сделать вывод о том, что из приведенных выше методов численного интегрирования наибольшую точность имеет метод Рунге – Кутты, явный Эйлер менее точен и имеет свойство накапливать ошибку, также его точность критически зависит от шага, а неявный более устойчив, но при данном шаге может давать менее точное приближение, чем RK4.