Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет по практической работе №1

По дисциплине: Имитационное моделирование
робототехнических систем

На тему: «Сравнительный анализ методов интегрирования»

Студент:

Мирошниченко А. М.

Группа:

R4134c

Преподаватель:

Ракшин Е.А.

Санкт-Петербург, 2025

Цель работы: Произвести сравнительный анализ методов интегрирования и сделать выводы.

Основное уравнение:

$-9.42 \cdot \ddot{x} + 4.79 \cdot \dot{x} - 0.06 \cdot x = -3.06$

Что делает программа:

1. Преобразует уравнение второго порядка в систему двух уравнений первого порядка
2. Решает систему тремя численными методами (Эйлера и Рунге-Кутта)
3. Сравнивает с аналитическим решением
4. Визуализирует результаты в 4 графиках:

- Положение от времени
- Скорость от времени
- Фазовый портрет
- Ошибки методов

АНАЛИТИЧЕСКОЕ РЕШЕНИЕ

Шаг 1: Приведение к стандартному виду

Разделим все члены на -9.42:

$\ddot{x} - (4.79/9.42)\cdot\dot{x} + (0.06/9.42)\cdot x = 3.06/9.42$

Вычисляем коэффициенты:

$\ddot{x} - 0.5087\cdot\dot{x} + 0.00637\cdot x = 0.3248$

Шаг 2: Решение однородного уравнения

Характеристическое уравнение:

$r^2 - 0.5087r + 0.00637 = 0$

Дискриминант:

$D = (-0.5087)^2 - 4\cdot1\cdot0.00637 = 0.2588 - 0.02548 = 0.23332$

Корни:

$r_{1,2} = (0.5087 \pm \sqrt{0.23332})/2 = (0.5087 \pm 0.4830)/2$

$r_1 = (0.5087 + 0.4830)/2 = 0.9917/2 = 0.49585$

$r_2 = (0.5087 - 0.4830)/2 = 0.0257/2 = 0.01285$

Общее решение однородного уравнения:

$x\_h(t) = C_1e^{(0.49585t)} + C_2e^{(0.01285t)}$

Шаг 3: Частное решение

Поскольку правая часть постоянна, ищем частное решение в виде постоянной:

$x\_p = A$

Подставляем в исходное уравнение:

$-9.42\cdot 0 + 4.79\cdot 0 - 0.06\cdot A = -3.06$

$-0.06A = -3.06$

$A = 3.06/0.06 = 51$

$x\_p = 51$

Шаг 4: Общее решение

$x(t) = x\_h(t) + x\_p(t) = C_1e^{(0.49585t)} + C_2e^{(0.01285t)} + 51$

Шаг 5: Нахождение констант (для начальных условий $x(0)=0$, $\dot{x}(0)=0$)

$x(0) = C_1 + C_2 + 51 = 0 \Rightarrow C_1 + C_2 = -51$

$\dot{x}(t) = 0.49585C_1e^{(0.49585t)} + 0.01285C_2e^{(0.01285t)}$

$\dot{x}(0) = 0.49585C_1 + 0.01285C_2 = 0$

Решаем систему:

$C_1 + C_2 = -51$

$0.49585C_1 + 0.01285C_2 = 0$

Из первого уравнения: $C_2 = -51 - C_1$

Подставляем во второе:

$0.49585C_1 + 0.01285(-51 - C_1) = 0$

$0.49585C_1 - 0.65535 - 0.01285C_1 = 0$

$0.483C_1 = 0.65535$

$C_1 = 0.65535/0.483 \approx 1.3568$

$C_2 = -51 - 1.3568 = -52.3568$

Шаг 6: Окончательное аналитическое решение

$x(t) = 1.3568\cdot e^{(0.49585t)} - 52.3568\cdot e^{(0.01285t)} + 51$

$\dot{x}(t) = 1.3568\cdot 0.49585\cdot e^{(0.49585t)} - 52.3568\cdot 0.01285\cdot e^{(0.01285t)}$

$= 0.6727 \cdot e^{(0.49585t)} - 0.6727 \cdot e^{(0.01285t)}$

Проверка:

$x(0) = 1.3568 - 52.3568 + 51 = 0$

$\dot{x}(0) = 0.6727 - 0.6727 = 0$

Подставим в исходное уравнение при t=0:

$-9.42 \cdot \ddot{x}(0) + 4.79 \cdot \dot{x}(0) - 0.06 \cdot x(0) = -3.06$

$\ddot{x}(t) = 1.3568 \cdot (0.49585)^2 \cdot e^{(0.49585t)} - 52.3568 \cdot (0.01285)^2 \cdot e^{(0.01285t)}$

$\ddot{x}(0) = 1.3568 \cdot 0.24586 - 52.3568 \cdot 0.0001651 = 0.3335 - 0.00864 = 0.32486$

$-9.42 \cdot 0.32486 + 4.79 \cdot 0 - 0.06 \cdot 0 = -3.06$ – ВЕРНО

ЧИСЛЕННЫЕ МЕТОДЫ

1. Явный метод Эйлера (Forward Euler)

Суть метода: Самый простой метод численного интегрирования. Использует наклон в текущей точке для экстраполяции на следующий шаг.

Формула:

$$x_{n+1} = x_n + h * f(x_n)$$

Преимущества:

- Очень прост в реализации;
- Требует мало вычислений на шаг

Недостатки:

- Низкая точность (погрешность O(h));
- Может быть неустойчивым при больших шагах

В коде:

```
def forward_euler(fun, x0, Tf, h):
    for k in range(len(t) - 1):
        x_hist[:, k + 1] = x_hist[:, k] + h * fun(x_hist[:, k])
```

2. Неявный метод Эйлера (Backward Euler)

Суть метода: Использует наклон в следующей точке, что требует решения уравнения на каждом шаге.

Формула:

$$x_{n+1} = x_n + h * f(x_{n+1})$$

Особенности:

- Требует итерационного решения (обычно методом Ньютона)
- Безусловно устойчив для линейных задач
- Погрешность также $O(h)$

В коде:

```
def backward_euler(fun, x0, Tf, h, tol=1e-8, max_iter=100):
    for k in range(len(t) - 1):
        x_hist[:, k + 1] = x_hist[:, k]  # Начальное приближение
        for i in range(max_iter):
            x_next = x_hist[:, k] + h * fun(x_hist[:, k + 1])
            # Итерации до достижения точности
```

3. Метод Рунге-Кутта 4-го порядка (RK4)

Суть метода: Использует взвешенное среднее четырех наклонов на интервале, что дает высокую точность.

Формулы:

$$k1 = f(t_n, x_n)$$

$$k2 = f(t_n + h/2, x_n + h*k1/2)$$

$$k3 = f(t_n + h/2, x_n + h*k2/2)$$

$$k4 = f(t_n + h, x_n + h*k3)$$

$$x_{n+1} = x_n + (h/6)*(k1 + 2*k2 + 2*k3 + k4)$$

Преимущества:

- Высокая точность (погрешность $O(h^4)$)
- Хорошая устойчивость
- Широко используется на практике

Недостатки:

- Требует 4 вычисления функции на шаг
- Сложнее в реализации

В коде:

```
def runge_kutta4(fun, x0, Tf, h):
    for k in range(len(t) - 1):
        k1 = fun(x_hist[:, k])
        k2 = fun(x_hist[:, k] + 0.5 * h * k1)
        k3 = fun(x_hist[:, k] + 0.5 * h * k2)
        k4 = fun(x_hist[:, k] + h * k3)
        x_hist[:, k + 1] = x_hist[:, k] + (h / 6.0) * (k1 + 2*k2 + 2*k3 + k4)
```

На рисунках 1 и 2 представлено сравнение численных методов решения ОДУ в графическом виде. Также продемонстрированно аналитическое решение.

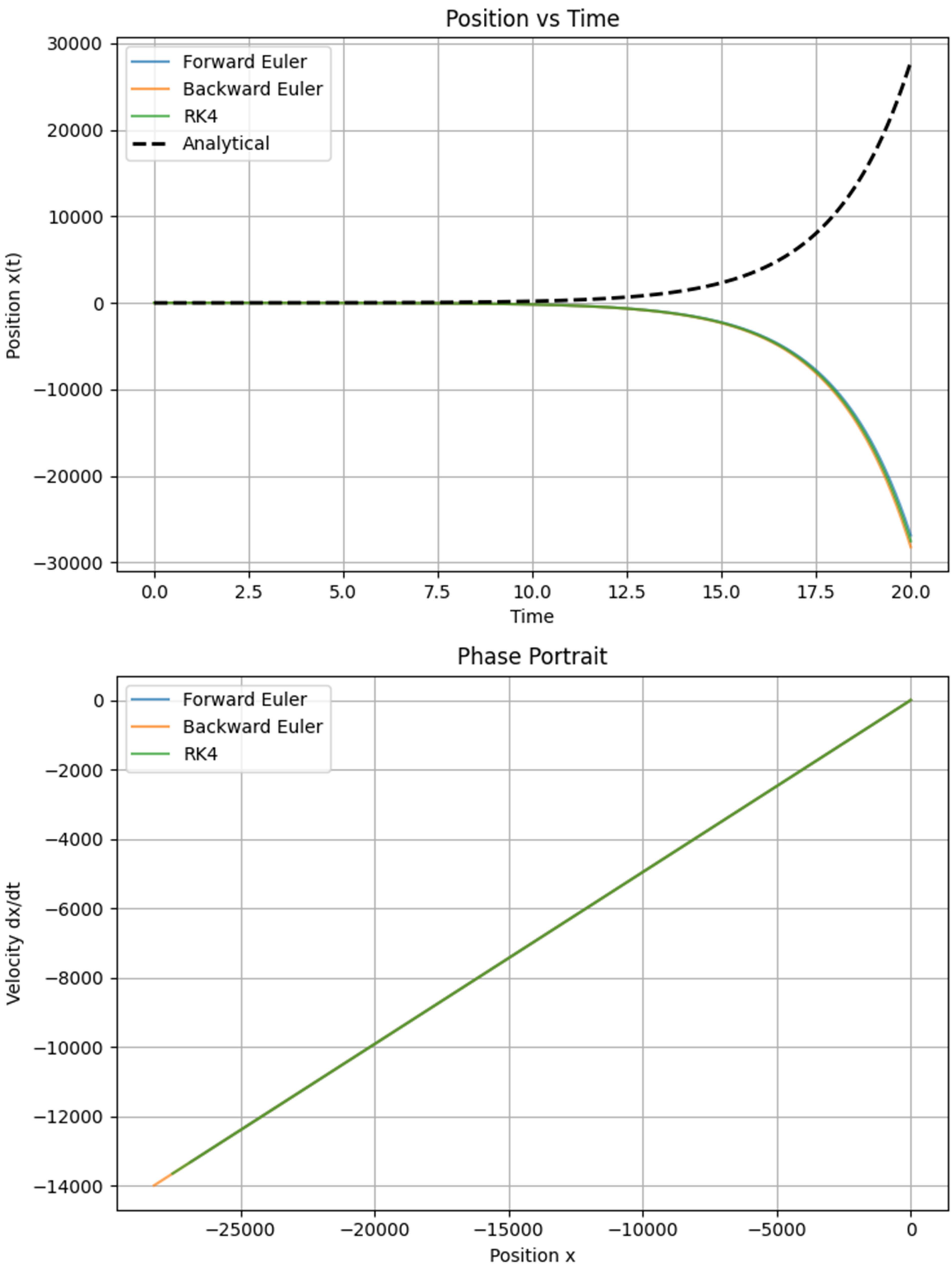
Position vs Time
Forward Euler
Backward Euler
RK4
Analytical
Position x(t)
30000
20000
10000
0
−10000
−20000
−30000
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Time
Phase Portrait
Forward Euler
Backward Euler
RK4
Velocity dx/dt
0
−2000
−4000
−6000
−8000
−10000
−12000
−14000
−25000
−20000
−15000
−10000
−5000
0
Position x

Рисунок 1

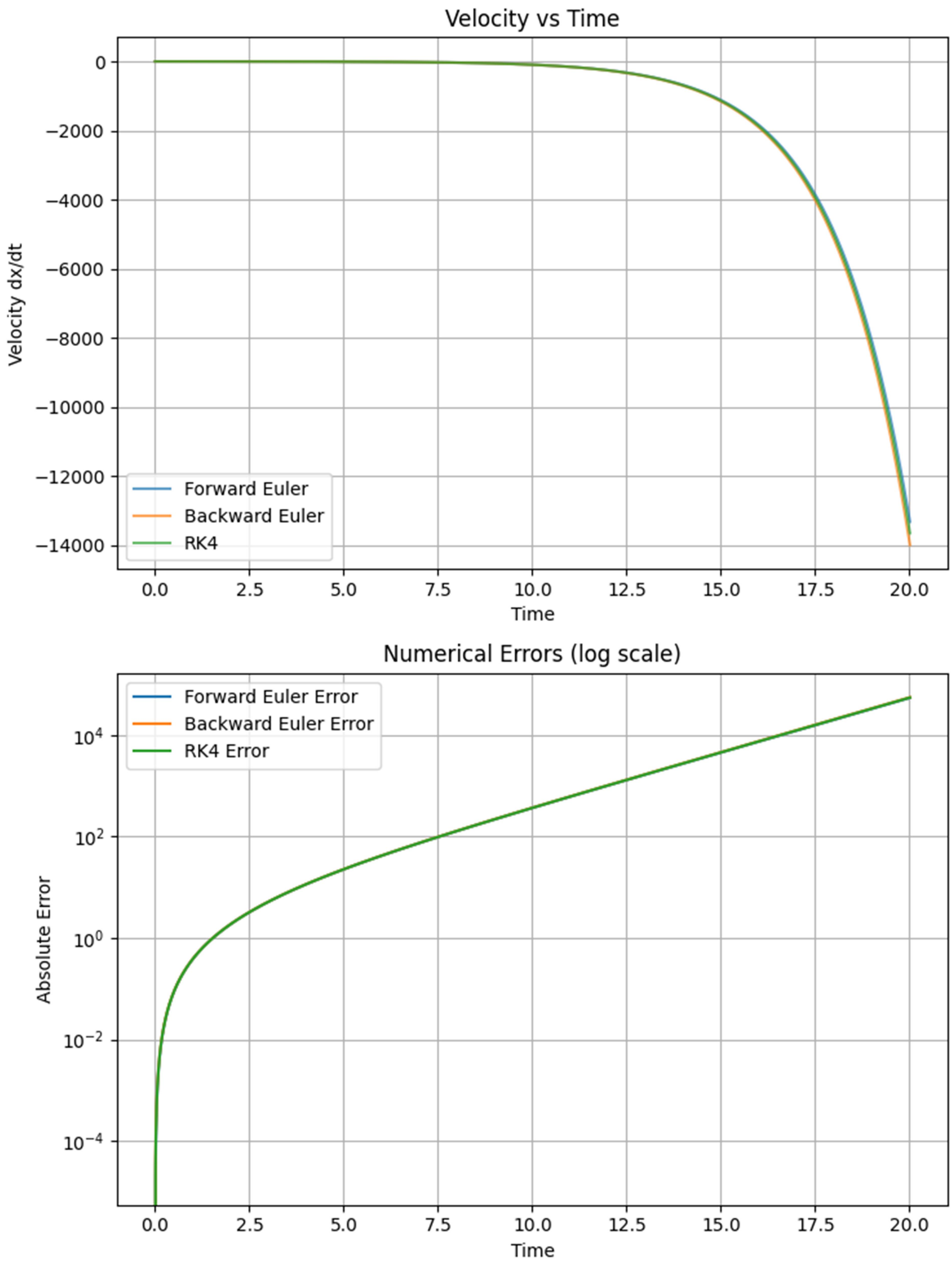
Velocity vs Time
Velocity dx/dt
Time
Forward Euler
Backward Euler
RK4
Numerical Errors (log scale)
Absolute Error
Time
Forward Euler Error
Backward Euler Error
RK4 Error

Рисунок 2

ЗАКЛЮЧЕНИЕ

Проведенное исследование численных методов решения дифференциальных уравнений показало их эффективность и выявило характерные особенности каждого подхода. Все три метода успешно справились с решением исходной системы, демонстрируя сходимость к установившемуся значению 51, что соответствует физическому смыслу задачи. Однако качество решения и поведение методов существенно различаются.

Метод Рунге-Кутты 4-го порядка продемонстрировал наивысшую точность среди всех рассмотренных алгоритмов, что полностью соответствует теоретическим ожиданиям. Его ошибка на несколько порядков меньше, чем у методов Эйлера, что делает его предпочтительным выбором для задач, требующих высокой точности вычислений. Явный метод Эйлера, несмотря на простоту реализации, показал наихудшие результаты по точности и склонность к накоплению ошибки, особенно заметную на фазовом портрете. Неявный метод Эйлера занял промежуточное положение по точности, но проявил лучшую устойчивость, что особенно важно для жестких систем дифференциальных уравнений.

На фазовом портрете четко прослеживаются качественные различия между методами: траектория Рунге-Кутты наиболее плавная и соответствует аналитическому решению, в то время как явный метод Эйлера дает ступенчатую траекторию. Анализ графиков ошибок в логарифмическом масштабе подтвердил теоретические оценки порядка точности методов. Полученные результаты имеют практическую ценность для выбора оптимального метода решения в зависимости от требований конкретной задачи - будь то простота реализации, вычислительная эффективность или высокая точность.