Параллельное программирование
Лабораторная работа №3. “Простецкая”
Численные методы решения ОДУ

Цель работы

Реализация численных методов решения обыкновенных дифференциальных уравнений (ОДУ). Использование встроенного в MatLAB функции `ode45`, реализация метода Рунге-Кутты 4-го порядка и метода Адамса-Башфорта 5-го порядка. Оценка точности численного решения ОДУ.

Задание

1. Создать функцию `lab03_func`, которая возвращает вектор дифференциалов первой степени ОДУ в зависимости от $x(t_i)$ и t_i следующего ОДУ описывающего движение маятника

   $$m\,l^2\,\ddot{\theta} + m\,g\,l\,\sin(\theta) + k\,l^2\,\dot{\theta} = u,$$

   где масса $m = 1.2$ кг, длина $l = 1.2$ м, $g = 9.81$ м/с^2 , $k = 2.2$ Н/м, $u = 1.3$ Н/м - постоянно действующий момент. Все переменные задать как глобальные. Интервал времени для решения дифференциального уравнения выбрать таким, при котором решение можно считать установившимся.

   Примечание: см. пример документации MatLAB по функции `ode45`.

2. Создать функцию `lab03_rk4` реализующую численное решение ОДУ методом Рунге-Кутты 4-го порядка, возвращающую матрицу решения X, и принимающую в качестве входных параметров вектор времени t, указатель на функцию `lab03_func` и вектор начальных значений $x(t_0)$.

   Примечание: т.к. под решением ОДУ понимается решение относительно всех производных до $n-1$ порядка включительно, то X получается матрицей, где i-ая строка содержит решение относительно каждой производной $x^{(0)}(t)\ldots x^{(n-1)}(t)$ в момент времени t_i, а количество строк определяется длиной вектора времени t.

3. Создать функцию `lab03_ab5` реализующую численное решение ОДУ методом Адамса-Башфорта 5-го порядка, возвращающую матрицу решения X, и принимающую в качестве входных параметров вектор времени t, указатель на функцию `lab03_func` и вектор начальных значений $x(t_0)$.

   Примечание: вычисление первых четырех точек осуществить методом Рунге-Кутты 4-го порядка непосредственно внутри функции `lab03_ab5`.

4. Задать вектор времени t значениями от 0.0 до 10 секунд с шагом $h = 0.01$ секунды. Используя `ode45`, `lab03_rk4`, `lab03_ab5` получить численные решения ОДУ соответствующими методами.

5. Построить на одной канве три вертикально расположенных графика для каждого найденного решения решения. На каждом графике построить кривые решений $x^{(0)}(t)$ и $x^{(1)}(t)$ и кривую решения $x(t)$ предварительно полученную аналитически. Результат `ode45` использовать в качестве проверочного. Т.к. размерности $x^{(0)}(t)$ и $x^{(1)}(t)$ не совпадают (разный физический смысл), графики построить с двумя осями (`yyaxis`), где ось слева соответствует $x^{(0)}(t)$, а ось справа $x^{(1)}(t)$. Цвета кривых и осей сделать идентичными (через настройку осей).

6. Найти решение ОДУ с различными значениями шага h равными 6.0×10^{-1}, 3.0×10^{-1} и 1.0×10^{-3} для метода Рунге-Кутты 4-го порядка и 7.2×10^{-2}, 7.0×10^{-1} и 10^{-3} для метода Адамса-Башфорта 5-го порядка.

7. Построить на одной канве два (каждый метод - свой графика) вертикально расположенных графика на каждом показав решение $x^{(0)}(t)$ полученное для соответствующих значений h. Сделать выводы об устойчивости алгоритмов численного интегрирования дифференциальных уравнений.

8. Оценить точность решения ОДУ методом Рунге-Кутты 4-го порядка используя правилом Рунге приняв h равное 6.0×10^{-3} и 3.0×10^{-3}. Результаты вывести в командную строку.

<u>БОНУС:</u>

(+2 балла в итог семестра). Реализовать метод Рунге-Кутты 4-го порядка с динамическим (автоматическим) выбором шага h (оформить как отдельную функцию). Сравнить точность решения ОДУ правилом Рунге используя h равное 6.0×10^{-3} и 3.0×10^{-3}. Результаты вывести в командную строку.