Групповой проект Хищник-жертва

Алгоритмы решения задачи

Беличева Д. М., Демидова Е. А.,
Самигуллин Э. А., Смирнов-Мальцев Е. Д.

Содержание

Список иллюстраций

1 Цель работы

Рассмотреть численные методы решения дифференциальных уравнений для построения модели Хищник-жертва и обосновать выбор Octave для программной реализации.

2 Задачи

- Описать метод Эйлера
- Описать метод Рунге-Кутта второго порядка
- Описать метод Рунге-Кутта четвёртого порядка
- Сравнить рассматриваемые методы

3 Метод Эйлера

Метод Эйлера играет важную роль в теории численных методов решения ОДУ, хотя и не часто используется в практических расчетах из-за невысокой точности.

Рассмотрим задачу Коши для дифференциального уравнения

$$y'(x) = f(x, y) \tag{1}$$

удовлетворяющее начальному условию

$$y(x_0) = y_0 \tag{2}$$

Требуется найти функцию $y = y(x)$, которая удовлетворяет уравнению (1) на интервале $(x_0; x_n)$ и начальному условию (2).

Проведём разбиение отрезка $[x_0; x_n]$ на n равных частей[1]:

$$x_i = x_0 + ih, i = \overline{1, n}$$
$$h = \frac{x_n - x_0}{n}$$

Для вычисления значения функции в точке x_1 разложим функцию $y = y(x)$ в окрестности точки x_0 в ряд Тейлора:

$$y(x_1) = y(x_0 + h) = y(x_0) + y'(x_0)h + y''(x_0)\frac{h^2}{2} + \dots$$

При достаточном малом значении h членами выше второго порядка можно пренебречь и с учетом $y'(x_0) = f(x_0, y_0)$ получим следующую формулу для вычисления приближенного значения функции $y(x)$ в точке x_1:

$$y_2 = y_1 + hf(x_1, y_1)$$

Повторяя этот процесс, сформируем последовательность значений x_i в точках t_i по формуле:

$$y_{i+1} = y_i + hf(x_i, y_i) \tag{3}$$

Процесс нахождения значений функции x_i в узловых точках t_i по формуле (3) называется методом Эйлера. Так как точность определяется отброшенными членами ряда, то точность метода Эйлера на каждом шаге составляет $O(h^2)$. в целом точность этого метода $O(h)$.

Алгоритм метода Эйлера можно построить в виде двух программных модулей: основной программы и подпрограммы Euler, реализующей метод(рис. 3.1).

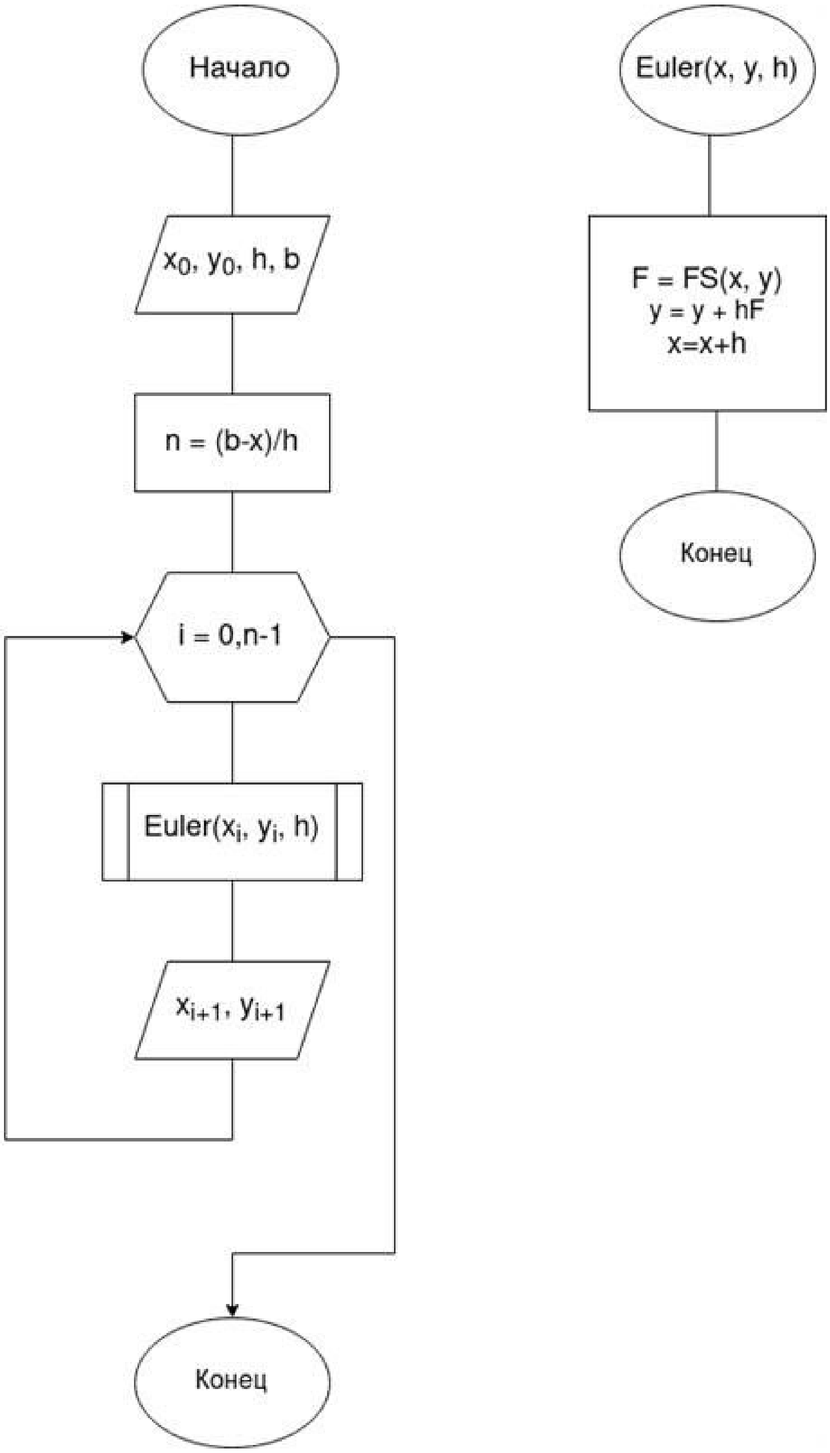

Рис. 3.1: Блок-схема алгоритма метода Эйлера

4 Метод Рунге-Кутта второго порядка

Рассмотрим рассчётные формулы метода Рунге-Кутта второго порядка[1]:

$$\begin{cases} x_{i+1} = x_i + \Delta x_i, i = \overline{1,n} \\ \Delta x_i = \frac{h}{2}(K_1^i + K_2^i) \\ K_1^i = f(x_i, y_i) \\ K_2^i = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_1^i) \end{cases}$$

Этот метод имеет второй порядок точности. Это значит, что ошибка на одном шаге имеет порядок $O(h^3)$, а суммарная ошибка на конечном интервале интегрирования имеет порядок $O(h^2)$.

Алгоритм метода Рунге-Кутта второго порядка можно построить в виде двух программных модулей: основной программы и подпрограммы Rk2, реализующей метод. (рис. 4.1).

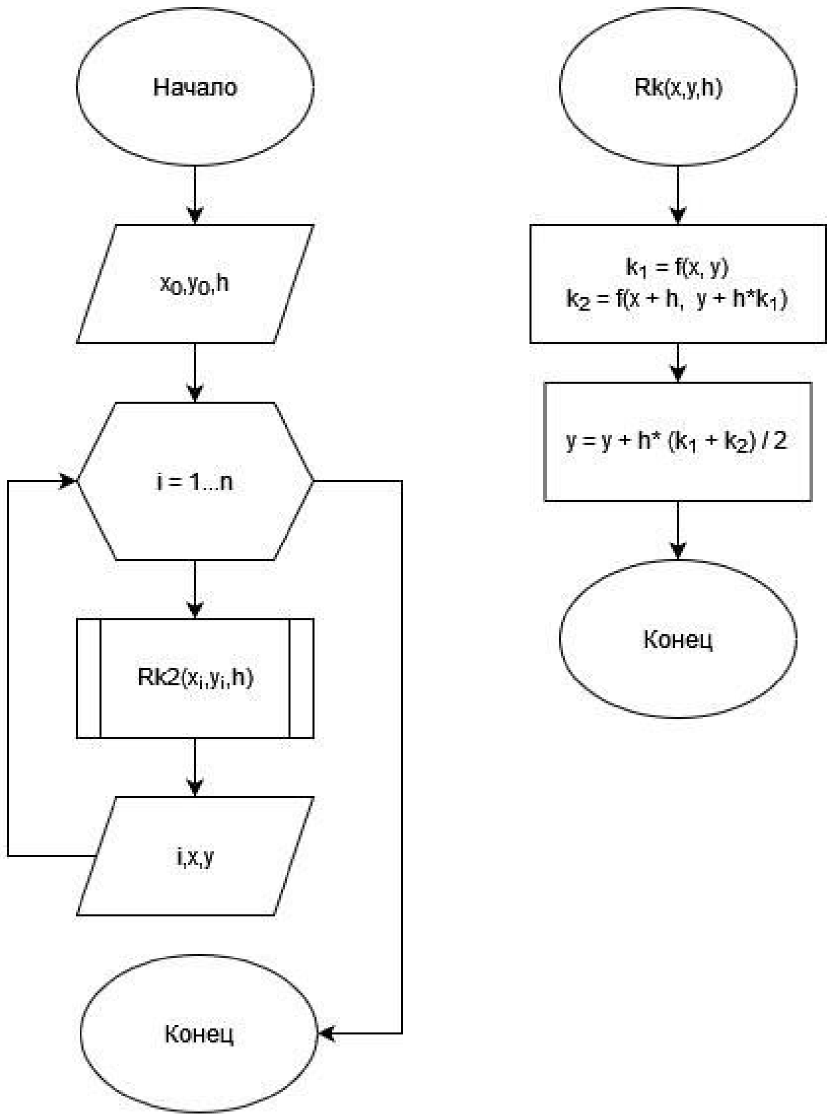

Рис. 4.1: Блок-схема алгоритма метода Рунге-Кутта второго порядка

5 Метод Рунге-Кутта четвёртого порядка

Рассмотрим рассчётные формулы метода Рунге-Кутта четвертого порядка[1]:

$$\begin{cases} x_{i+1} = x_i + \Delta x_i, i = \overline{1, n} \\ \Delta x_i = \frac{h}{6}(K_1^i + 2K_2^i + 2K_3^i + K_4^i) \\ K_1^i = f(x_i, y_i) \\ K_2^i = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_1^i) \\ K_3^i = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_2^i) \\ K_4^i = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_3^i) \end{cases}$$

Этот метод имеет четвёртый порядок точности. Это значит, что ошибка на одном шаге имеет порядок $O(h^5)$, а суммарная ошибка на конечном интервале интегрирования имеет порядок $O(h^4)$.

Алгоритм метода Рунге-Кутта четвёртого порядка можно построить в виде двух программных модулей: основной программы и подпрограммы Rk4, реализующей метод(рис. 5.1).

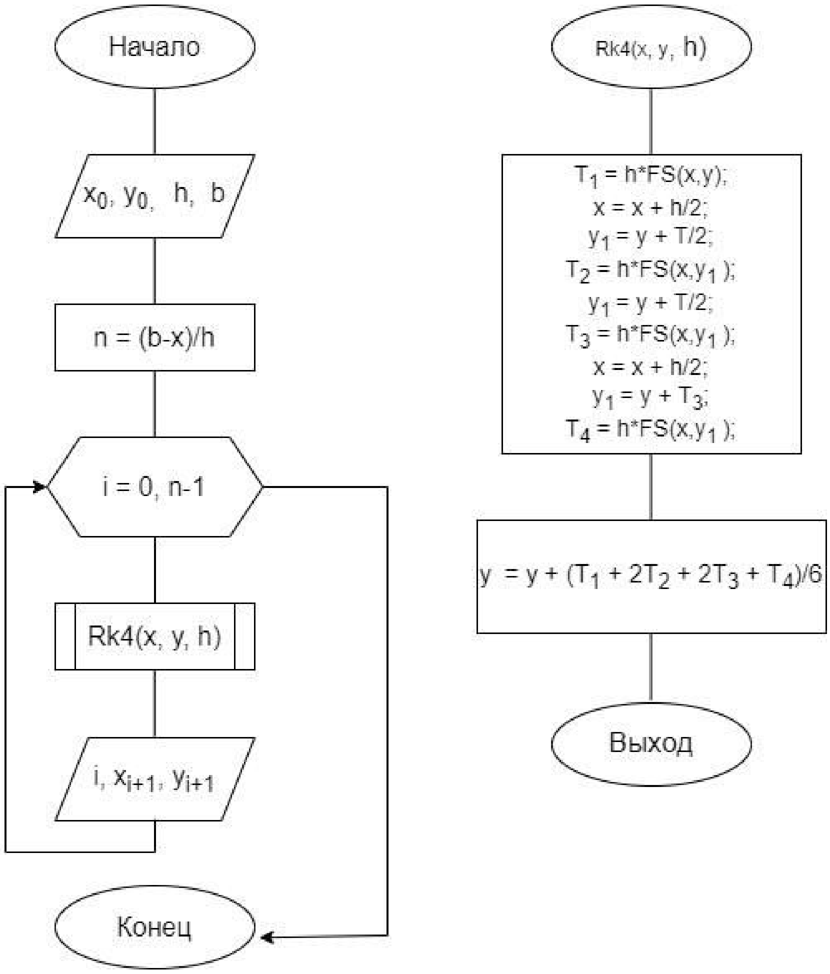

Рис. 5.1: Блок-схема алгоритма метода Рунге-Кутта четвёртого порядка

6 Выбор системы для математических вычислений Octave

В нашей работе будет использована система для математических вычислений Octave. Octave совместим с Matlab на уровне интерфейса и языка программирования. Также есть все базовые функции Matlab. Кроме того эта система совместима как с Linux, так и с Windows. Ещё одна причина выбора именно Octave состоит в том, что в этой системе есть программная реализация метода Эйлера и методов Рунге-Кутта[2].

Исследователи из Университета Мэриленда в США провели сравнительный анализ математических вычислений, используя MATLAB, Octave, SciLab и FreeMat в простом сценарии и в сложном[3]. В первом случае решали систему линейных уравнений а в втором — конечно-разностную дискретизацию уравнения Пуассона в двухмерном пространстве. Основной вывод — GNU Octave справляется с задачами лучше остальных открытых математических пакетов, демонстрируя результат, сопоставимый с матлабовским.

7 Заключение

Для исследования модели Хищник-жертва в нашей работе будут использованы метод Эйлера и методы Рунге-Кутта, а программная реализация будет выполнена в системе математических вычисленый Octave.

Список литературы

1. Кулакова С.В. Численные методы. гос. хим.-технол. ун-т. Иваново, 2018. 124 с.
2. GNU Octave Documentation [Электронный ресурс]. Free Software Foundation, 2023. URL: https://docs.octave.org/latest/.
3. Sharma N., Gobbert M.K. A comparative evaluation of Matlab, Octave, Freemat, and Scilab for research and teaching. Department of Mathematics; Statistics, University of Maryland, Baltimore County, 2010. 37 c.