МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ (ШКОЛА)

Департамент математического и компьютерного моделирования

ОТЧЁТ

к лабораторной работе №2 по дисциплине

«Математическое и копмьютерное моделирование»

Направление подготовки
01.03.02 «Прикладная математика и информатика»

Выполнил студент гр.
Б9121-01.03.02сп
Держапольский Ю.В. ________
(Ф.И.О.) *(подпись)*

Профессор к.ф.-м. н.
Пермяков М. С. ________
(Ф.И.О.) *(подпись)*

« 19 » апреля 2024 г.

г. Владивосток

2024

Содержание

1. Введение

Хищник-жертва

2. Построение математической модели

Мы будем строить модель для двух популяций и рассматривать их взаимодействие во времени. Пусть $N(t)$ – популяция жертв, $M(t)$ – популяция хищников в зависимости от времени. Единицы измерения выберем безразмерные и скажем, что это некоторый «объём» популяции. В общем случае будем рассматривать при $N(t), M(t) > 0$. Время измеряется в секундах.

В отсутствии взаимодействия между ними популяция хищников будет расти, а популяция жертв убывать в зависимости от объёмов самих популяций:

$$\begin{cases} \dfrac{dN}{dt} = a \cdot N, \\ \dfrac{dM}{dt} = -b \cdot M, \end{cases}$$

где $a, b > 0$ – коэффициенты, обозначающие скорость изменения популяции.

При взаимодействии популяции будут влиять друг на друга. Хищники будут уменьшать популяцию жертв, а жертвы будут увеличивать популяцию хищников:

$$\begin{cases} \dfrac{dN}{dt} = (a - c \cdot M)N, \\ \dfrac{dM}{dt} = (-b + d \cdot N)M, \end{cases}$$

где $c, d > 0$ – коэффициенты, обозначающие скорость изменения популяции в зависимости от второй популяции.

Таким образом, построили модель, являющуюся системой обыкновенных дифференциальных уравнений, которая описывает взаимодействие двух популяций модели «Хищник-жертва». Для получения единственного решения добавим начальные условия:

$$\begin{cases} N(0) = N_0, \\ M(0) = M_0, \end{cases}$$

3. Анализ модели

Исследуем систему на устойчивость.

$$\begin{cases} \dfrac{dN}{dt} = (a - c \cdot M)N = 0, \\ \dfrac{dM}{dt} = (-b + d \cdot N)M = 0; \end{cases} \Rightarrow \begin{cases} a - c \cdot M = 0, \\ -b + d \cdot N = 0; \end{cases} \Rightarrow \begin{cases} M_e = \dfrac{a}{c}, \\ N_e = \dfrac{b}{d}. \end{cases}$$

Также есть тривиальное решение $(0, 0)$.

Применим метод первого приближения. Найдём матрицу Якоби.

$$J = \begin{pmatrix} a - c \cdot M & -c \cdot N \\ d \cdot M & -b + d \cdot N \end{pmatrix}.$$

Подставим точки равновесия в матрицу и найдём собственные значения.

$$J\Big|_{(0,0)} = \begin{pmatrix} a & 0 \\ 0 & -b \end{pmatrix} \Rightarrow \lambda_1 = -b, \lambda_2 = a.$$

Нулевая точка является седловой точкой.

$$J\Big|_{(N_e,M_e)} = \begin{pmatrix} 0 & -\dfrac{c \cdot b}{d} \\ \dfrac{d \cdot a}{c} & 0 \end{pmatrix} \Rightarrow \lambda_{1,2} = \pm i\sqrt{ab}.$$

Точка (N_e, M_e) является «Центром» – окружность с центром в этой точке или эллипс.

4. Вычислительные эксперименты

4.1. Алгоритм

Для компьютерного вычисления будем использовать метод Рунге-Кутты, с помощью которого получим численное решение системы дифференциальных уравнений с заданными параметрами. После чего построим графики решений на координатной и фазовой плоскостях.

4.2. Программа

Для расчётов и визуализации был использован язык Python с библиотеками numpy и matplotlib.

```
import numpy as np
import scipy as sci
import matplotlib.pyplot as plt
import math


def runge_kutta(function, y0: float, a: float, b: float, h: float):
    num = math.ceil((b - a) / h)
    x_a = np.linspace(a, b, num=num, endpoint=False)
    y_a = [y0] * num

    for i in range(num - 1):
        k0 = function(x_a[i], y_a[i])
        k1 = function(x_a[i] + h / 2, y_a[i] + h * k0 / 2)
        k2 = function(x_a[i] + h / 2, y_a[i] + h * k1 / 2)
        k3 = function(x_a[i] + h, y_a[i] + h * k2)
        y_a[i + 1] = y_a[i] + h / 6 * (k0 + 2 * k1 + 2 * k2 + k3)

    return x_a, np.array(y_a)


def count_time(a, b, c, d, y0):
    plt.figure("C(t)")
```

```
    leg = []
    for i in range(len(y0)):
        x, y = model(a, b, c, d, np.array(y0[i]))

        plt.plot(x, y[0])
        plt.plot(x, y[1])
        leg.extend([f"N{i} - жертвы", f"M{i} - хищники"])

    plt.xlabel('t - Время')
    plt.ylabel('C(t) - Количество')
    plt.legend(leg)

    plt.xlim(left=0)
    plt.ylim(bottom=0)


def phase(a, b, c, d, init):
    plt.figure("Phase")
    for i in init:
        x, y = model(a, b, c, d, np.array(i))
        plt.plot(y[0], y[1], marker='o', markevery=[0])

    plt.plot(b/d, a/c, 'ro')
    plt.legend(init)
    plt.xlabel('N - жертвы')
    plt.ylabel('M - хищники')

    plt.xlim(left=0)
    plt.ylim(bottom=0)

def model(a, b, c, d, y0):
    def diff(t, NM):
        return np.array([
            (a - c * NM[1]) * NM[0],
            (-b + d * NM[0]) * NM[1]
        ])


    x_ = np.linspace(t0, tn, n)
    x_, y_ = runge_kutta(diff, y0, t0, tn, (tn-t0)/n)
```

```
    y_ = y_.T

    return x_, y_



    # plt.plot(y[1], y[0])

t0, tn = 0, 10
n = 1000

# a, c = 2, 0.5
# b, d = 1, 0.5

a, c = 2, 2
b, d = 1, 4

ravno = [b/d, a/c]

init_val = [[4,4], [2,6], [3,2], [5, 3], [2,3], [2,1], [1/2, 1/2]]

# count_time(a,b,c,d, [[4,4], [2,6], ravno])

phase(a,b,c,d, init_val)

plt.savefig("./sem6-matmodelling/population.pdf")
plt.title(f"{a=}, {c=}; {b=}, {d=}")
print(f"{a = }, {c = }, {b = }, {d = }")
plt.show()
```

4.3. Графики на координатной плоскости

Построим несколько решений системы с одинаковыми параметрами, но разными начальными условиями, в том числе и точку равновесия.

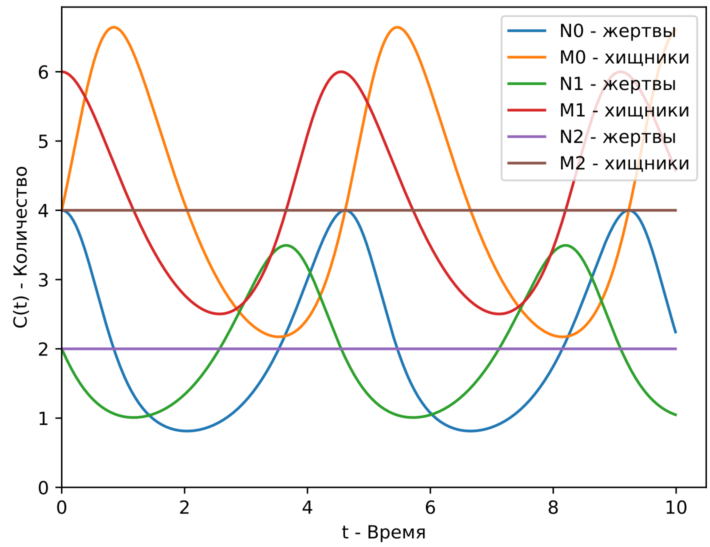

Рис. 1: Графики при $a = 2, c = 0.5, b = 1, d = 0.5$, с начальными условиями

$$x_0 = (4, 4),\ x_1 = (2, 6),\ x_2 = (N_e, M_e)$$

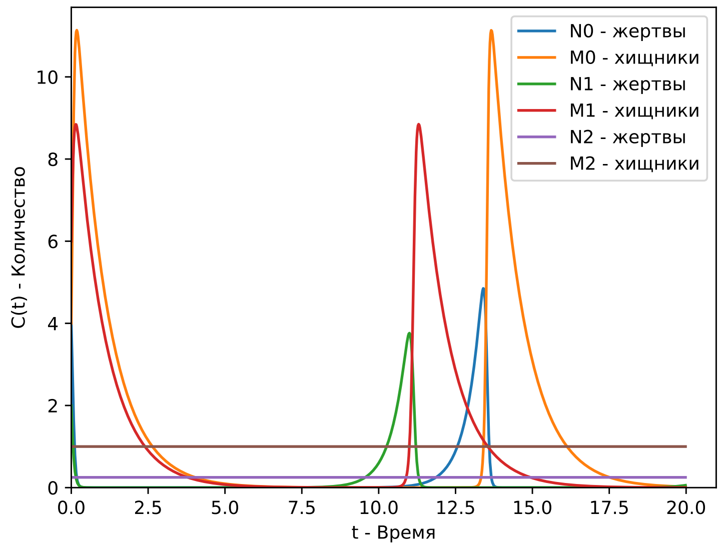

Рис. 2: Графики при $a = 2, c = 2, b = 1, d = 4$, с начальными условиями

$$x_0 = (4, 4),\ x_1 = (2, 6),\ x_2 = (N_e, M_e)$$

4.4. Графики на фазовой плоскости

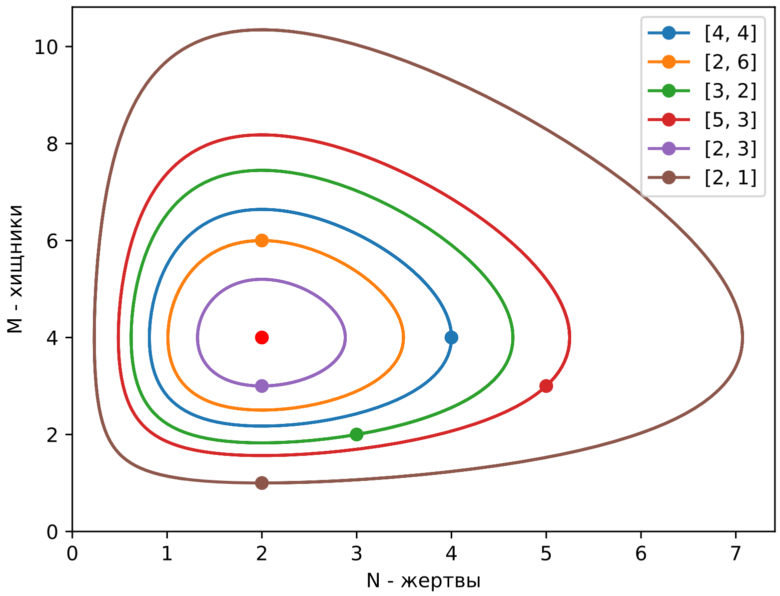

Рис. 3: Графики при $a = 2, c = 0.5, b = 1, d = 0.5$, с указанными начальными условиями на интервале времени $[0, 10]$.

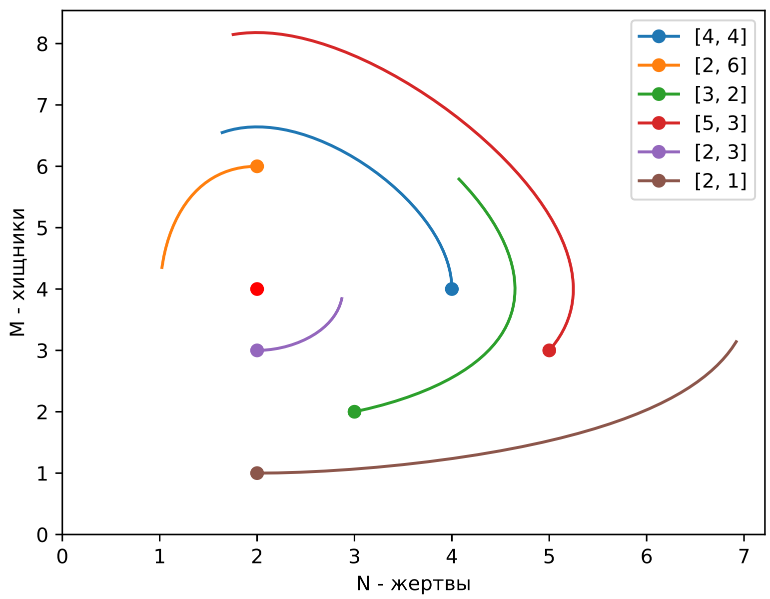

Рис. 4: Графики при $a = 2, c = 0.5, b = 1, d = 0.5$, с указанными начальными условиями на интервале времени $[0, 1]$.

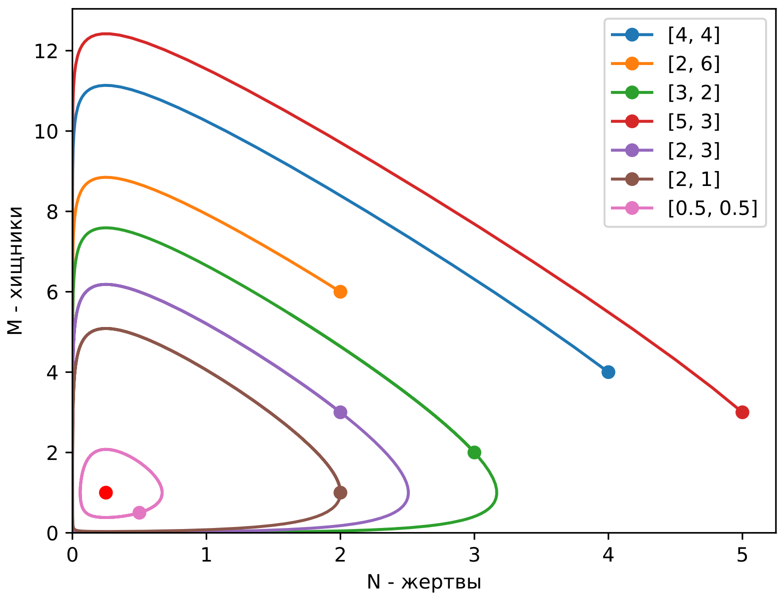

Рис. 5: Графики при $a = 2, c = 2, b = 1, d = 4$, с указанными начальными условиями на интервале времени $[0, 10]$.

5. Заключение

Таким образом, была построена математическая модель бла бла бла