

Этап 3

Программная реализация проекта хищник-жертва

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

3 июня 2023

Российский университет дружбы народов, Москва, Россия

Информация

Студенты группы НКНбд-01-21

- Беличева Дарья Михайловна
- Демидова Екатерина Алексеевна
- Самигуллин Эмиль Артурович
- Смирнов-Мальцев Егор Дмитриевич

Вводная часть

- Программная реализация проекта хищник-жертва

- Описать функции для решения ОДУ в Octave
- Построить график зависимости числа хищников от числа жертв
- Построить графики зависимости числа видов от времени
- Найти стационарное состояние системы

Программная реализация

- `ode23(@f, interval, X0, options)` – метод Богацки-Шампина
- `ode45(@f, interval, X0, options)` – метод Дормана-Принса

Опции:

- `RelTol` – относительная точность решения
- `AbsTol` – абсолютная точность решения
- `InitialStep` – начальное значение шага
- `MaxStep` – максимальное значение шага


```
function dx=f(t, x)
    a = 0.2; % коэффициент естественной смертности хищников
    b = 0.05; % коэффициент естественного прироста жертв
    c = 0.5; % коэффициент увеличения числа хищников
    d = 0.02; % коэффициент смертности жертв
    dx(1) = -a*x(1) + b*x(1)*x(2);
    dx(2) = c*x(2) - d*x(1)*x(2);
endfunction
```

```
A(:,1) = 21:0.5:25;  
A(:,2) = 1:0.5:5;  
  
for i = 1:size(A(:,1))  
    [T M] = ode23 (@f, [0 50], A(i,:));  
end
```

Построим стационарную точку системы.

```
[T M] = ode23 (@f, [0 30], [25 4]);
```

Построим зависимости численности популяций хищников и жертв.

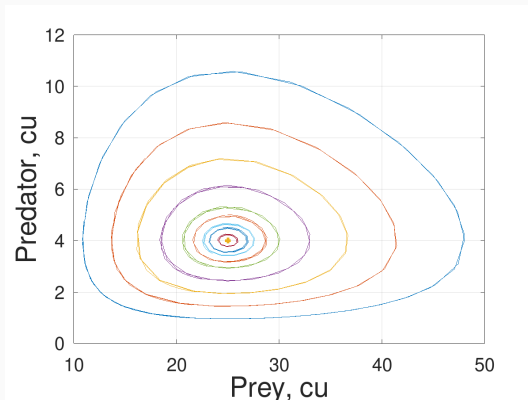


Рис. 1: Фазовый портрет

Реализация решения методом Богацки-Шампина с максимальным шагом 0.1.

```
A(:,1) = 21:0.5:25;  
A(:,2) = 1:0.5:5;  
opt = odeset ("MaxStep", 0.1);  
for i = 1:size(A(:,1))  
[T M] = ode23 (@f, [0 50], A(i,:),opt);
```

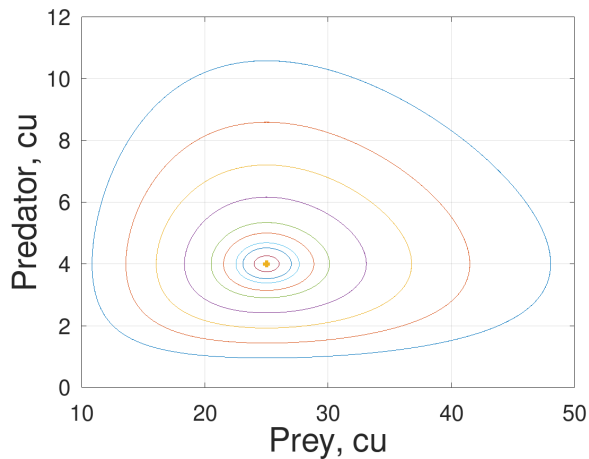


Рис. 2: Фазовый портрет с максимальным шагом 0.1

Построим зависимость видов от времени.

```
A(:,1) = 21:0.5:25;
```

```
A(:,2) = 1:0.5:5;
```

```
[T M] = ode23 (@f, [0 100], [21 1]);
```

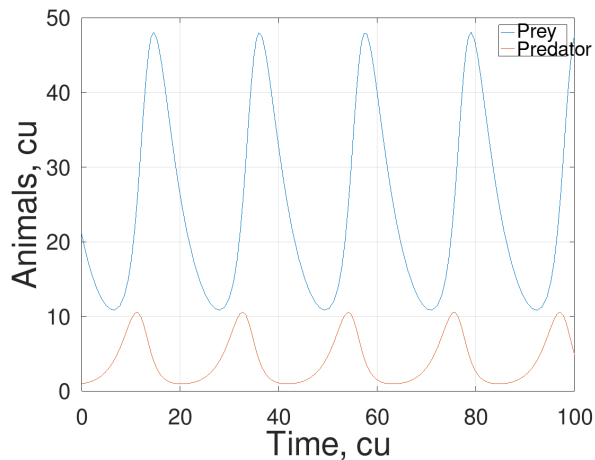


Рис. 3: Зависимость видов от времени


```
A(:,1) = 21:0.5:25;  
A(:,2) = 1:0.5:5;  
for i = 1:size(A(:,1))  
    [T M] = ode45 (@f, [0 50], A(i,:));  
end
```

Построим стационарную точку системы.

```
[T M] = ode45 (@f, [0 50], [25 4]);
```

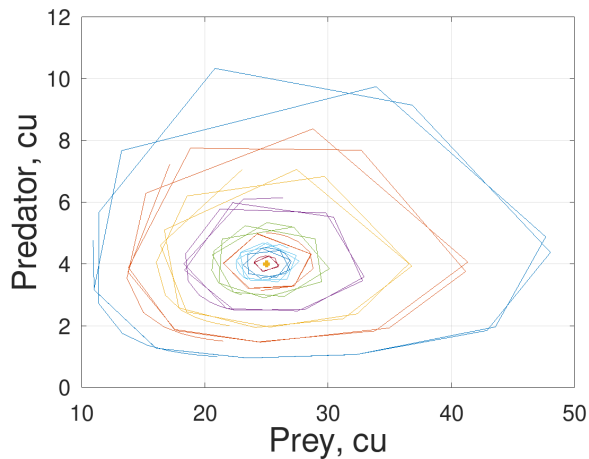


Рис. 4: Фазовый портрет

Реализация решения методом Дормана-Принса с максимальным шагом 0.1.

```
A(:,1) = 21:0.5:25;  
A(:,2) = 1:0.5:5;  
opt = odeset ("MaxStep", 0.1);  
for i = 1:size(A(:,1))  
    [T M] = ode45 (@f, [0 50], A(i,:),opt);  
end
```

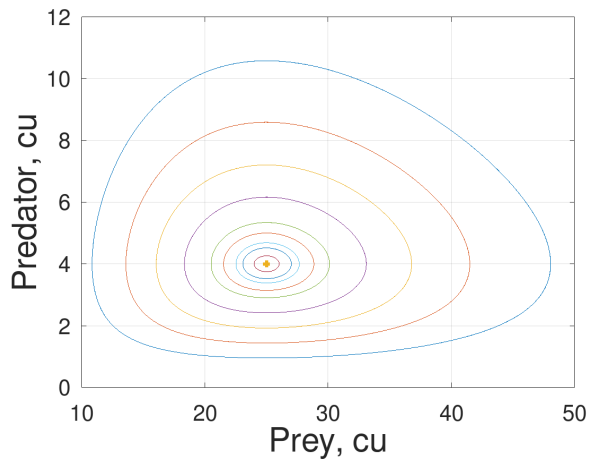


Рис. 5: Фазовый портрет с максимальным шагом 0.1

Построение зависимости видов от времени.

```
A(:,1) = 21:0.5:25;
```

```
A(:,2) = 1:0.5:5;
```

```
par = odeset("MaxStep", 0.1)
```

```
[T M] = ode45 (@f, [0 100], [21 1], par);
```

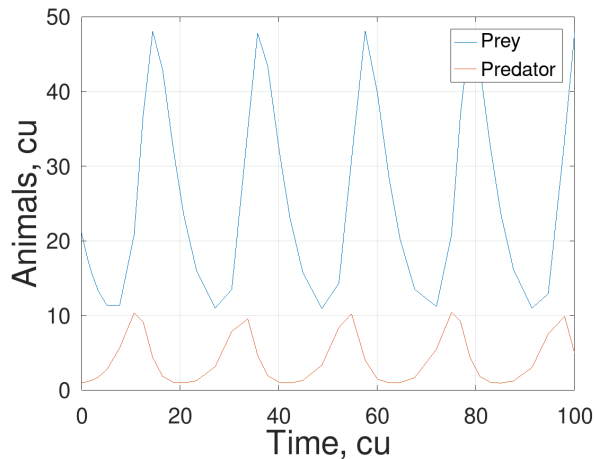


Рис. 6: Зависимость видов от времени

Заключение

В процессе выполнения этого этапа проекта мы выполнили программную реализацию проекта хищник-жертва.

1. GNU Octave Documentation [Электронный ресурс]. Free Software Foundation, 2023. URL: https://docs.octave.org/v4.2.0/Matlab_002dcompatible-solvers.html.