

Лабораторная работа № 3

Математическое моделирование

Королёв И.А.

Российский университет дружбы народов, Москва, Россия

Цель работы

Рассмотреть простейшие модели боевых действий и смоделировать их на языках программирования Julia и Modelica.

Задание

Необходимо было рассмотреть 3 модели боя.

1. Модель боевых действий между регулярными войсками
2. Модель ведение боевых действий с участием регулярных войск и партизанских отрядов
3. Модель боевых действий между партизанскими отрядами

Задание

Выполнить самостоятельное задание. Вариант 13.

Между страной X и страной Y идет война. Численность состава войск исчисляется от начала войны, и являются временными функциями $x(t)$ и $y(t)$. В начальный момент времени страна X имеет армию численностью 40 000 человек, а в распоряжении страны Y армия численностью в 69 000 человек. Для упрощения модели считаем, что коэффициенты a, b, c, h постоянны. Также считаем $P(t)$ и $Q(t)$ непрерывные функции. Постройте графики изменения численности войск армии X и армии Y для следующих случаев:

1. Модель боевых действий между регулярными войсками

$$\begin{cases} \frac{dx}{dt} = -0.445x(t) - 0.806y(t) + \sin(t + 7) + 1 \\ \frac{dy}{dt} = -0.419x(t) - 0.703y(t) + \cos(t + 4) + 1 \end{cases}$$

Теоретическое введение

Рассмотрим некоторые простейшие модели боевых действий – модели Ланчестера. В противоборстве могут принимать участие как регулярные войска, так и партизанские отряды. В общем случае главной характеристикой соперников являются численности сторон. Если в какой-то момент времени одна из численностей обращается в нуль, то данная сторона считается проигравшей (при условии, что численность другой стороны в данный момент положительна). Рассмотрим три случая ведения боевых действий: 1. Боевые действия между регулярными войсками 2. Боевые действия с участием регулярных войск и партизанских отрядов 3. Боевые действия между партизанскими отрядами В первом случае численность регулярных войск определяется тремя факторами: * скорость уменьшения численности войск из-за причин, не связанных с боевыми действиями (болезни, травмы, дезертирство); * скорость потерь, обусловленных боевыми действиями противоборствующих сторон (что связано с качеством стратегии, уровнем вооружения, профессионализмом солдат и т.п.); * скорость поступления подкрепления (задаётся некоторой функцией от времени).

Выполнение лабораторной работы

Построение моделей на языке программирования Julia

Моделирование боевых действий между регулярными войсками. Подключение библиотек для ОДУ и построения графиков. Функция для построения модели, начальные параметры модели (численность армий, коэффициенты модели, временной интервал моделирования). Решение системы ОДУ и задачи численным методом.

Модель боевых действий между регулярными войсками

```
[1]: using DifferentialEquations, Plots # библиотека для ОДУ и построения графиков

[3]: function model_one(arm, coef, t)
      x, y = arm # числ армии
      a, b, c, h = coef # параметры модели
      dx = -a*x - b*y + sin(t*7) + 1 # изменение численности первой армии с учетом потерь и внешних факторов
      dy = -c*x - h*y + cos(t*4) + 1 # изменение численности второй армии с учетом потерь и внешних факторов
      return [dx, dy] # производные
    end

[3]: model_one (generic function with 1 method)

[5]: arm = [50000, 39000] # нач числ армии
      coef = [0.445, 0.006, 0.419, 0.703] # параметры модели
      t = (0,1) # временной интервал моделирования

[5]: (0, 1)

[17]: prob = ODEProblem(model_one, arm, t, coef)

[17]: ODEProblem with uType Vector{Int64} and tType Int64. In-place: false
      Non-trivial mass matrix: false
      timespan: (0, 1)
      u0: 2-element Vector{Int64}:
           50000
           39000

[19]: solution = solve(prob, Tsit5())

retcode: Success ***
```

Вывод результата модели боевых действий между регулярными войсками.

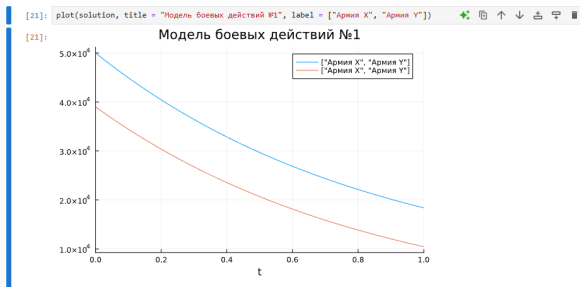


Рис. 2: Модель боевых действий № 1

Моделирование ведения боевых действий с участием регулярных войск и партизанских отрядов.

Модель ведение боевых действий с участием регулярных войск и партизанских отрядов

```
[65]: function model_one(arm, coef, t)
      x, y = arm # числ армии
      a, b, c, h = coef # параметры модели
      dx = -a*x - b*y + sin(2*pi*t) # изменение численности первой армии с учетом потерь и внешних факторов
      dy = -c*x*y - h*y + 2*cos(t) # изменение численности второй армии с учетом потерь и внешних факторов
      return [dx, dy] # производные
    end

[65]: model_one (generic function with 1 method)

[87]: arm = [50000, 30000] # нач числ армии
      coef = [0.203, 0.795, 0.203, 0.801] # параметры модели
      t = (0, 0.001) # временной интервал моделирования

[87]: (0, 0.001)

[89]: prob = ODEProblem(model_one, arm, t, coef)

[89]: ODEProblem with uType Vector{Int64} and tType Float64. In-place: false
      Non-trivial mass matrix: false
      timespan: (0.0, 0.001)
      u0: 2-element Vector{Int64}:
        50000
        30000

[91]: solution = solve(prob, Tsit5())

      retcode: Success ***
```

Рис. 3: Реализация

Построение моделей на языке программирования Julia

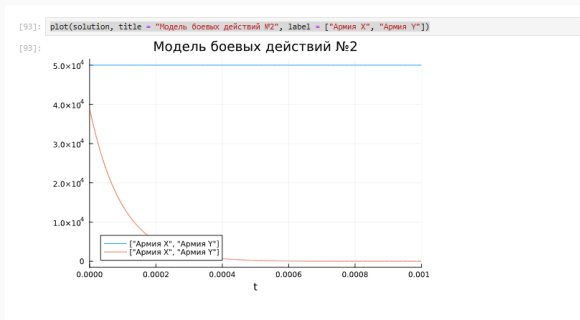


Рис. 4: Модель боевых действий № 2

Построение моделей на языке программирования Julia

Дополнительно, этого пункта нет в самостоятельном задании, но я его тоже смоделировал для полноценного изучения всех вариаций моделей.

Дополнительно. Третья модель из примера. Модель боевых действий между партизанскими отрядами.

```
[43]: function model_one(arm, coef, t)
      x, y = arm # числ армии
      a, b, c, h = coef # параметры модели
      dx = -a*x - b*x*y + sin(2*t) # изменение численности первой армии с учетом потерь и влияния факторов
      dy = -h*x - c*x*y + 2*cos(t) # изменение численности второй армии с учетом потерь и влияния факторов
      return [dx, dy] # производные
    end

[43]: model_one (generic function with 1 method)

[97]: arm = [50000, 39000] # нач числ армии
      coef = [0.205, 0.705, 0.205, 0.801] # параметры модели
      t = (0, 0.001) # временной интервал моделирования

[97]: (0, 0.001)

[99]: prob = ODEProblem(model_one, arm, t, coef)

[99]: ODEProblem with uType Vector{Int64} and tType Float64. In-place: false
      Non-trivial mass matrix: false
      timespan: (0.0, 0.001)
      u0: 2-element Vector{Int64}:
        50000
        39000

[101]: solution = solve(prob, Tsit5())

      retcode: Success ***
```

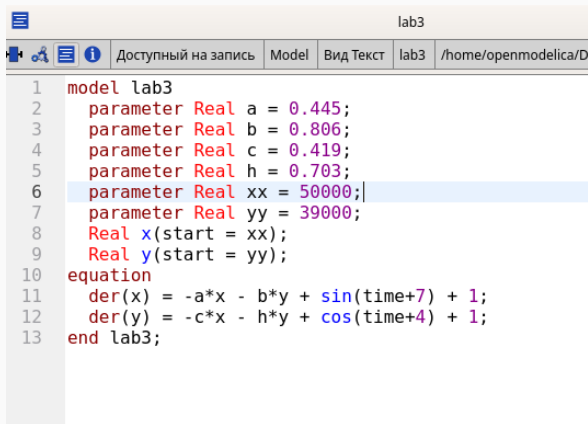
Рис. 5: Реализация



Рис. 6: Модель боевых действий № 3

Построение моделей на языке программирования Modelica

Моделирование боевых действий между регулярными войсками. Реализация почти аналогична с Julia, только небольшая особенность задания переменных и решения.



The screenshot shows the OpenModelica IDE interface. The title bar indicates the file is named 'lab3'. The menu bar includes options like 'Доступный на запись', 'Model', 'Вид Текст', 'lab3', and the file path '/home/openmodelica/D...'. The main editor area contains the following Modelica code:

```
1 model lab3
2   parameter Real a = 0.445;
3   parameter Real b = 0.806;
4   parameter Real c = 0.419;
5   parameter Real h = 0.703;
6   parameter Real xx = 50000;
7   parameter Real yy = 39000;
8   Real x(start = xx);
9   Real y(start = yy);
10  equation
11    der(x) = -a*x - b*y + sin(time+7) + 1;
12    der(y) = -c*x - h*y + cos(time+4) + 1;
13  end lab3;
```

Рис. 7: Реализация

Вывод результата модели боевых действий между регулярными войсками.

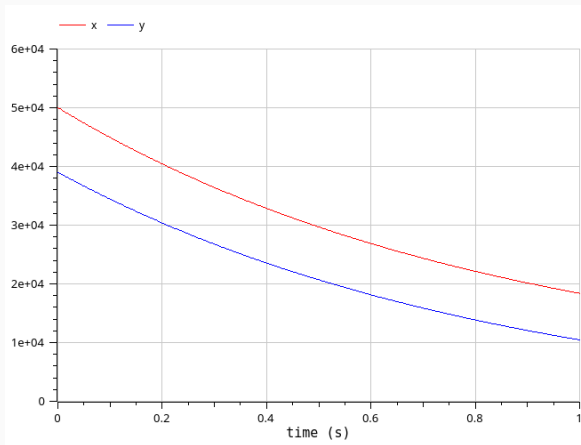


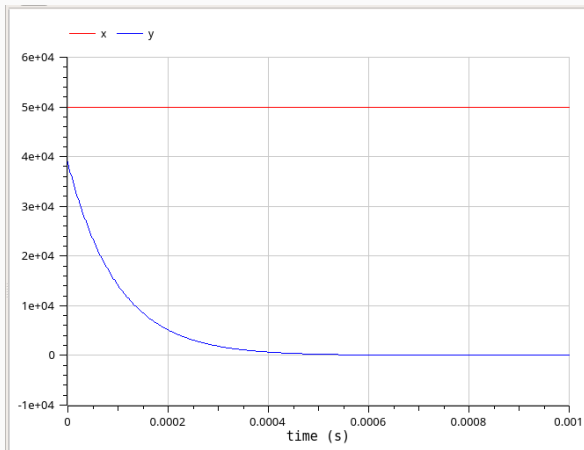
Рис. 8: Модель боевых действий № 1

Моделирование ведения боевых действий с участием регулярных войск и партизанских отрядов.

```
1 model lab3_2
2   parameter Real a = 0.203;
3   parameter Real b = 0.705;
4   parameter Real c = 0.203;
5   parameter Real h = 0.801;
6   parameter Real xx = 50000;
7   parameter Real yy = 39000;
8   Real x(start = xx);
9   Real y(start = yy);
10  equation
11    der(x) = -a*x - b*y + sin(2*time);
12    der(y) = -c*x*y - h*y + 2*cos(time);
13 end lab3_2;
```

Рис. 9: Реализация

Вывод результата ведения боевых действий с участием регулярных войск и партизанских отрядов. Для более наглядного вывода, я сократил временной интервал.



Сравнивая графики, полученные в Julia и OpenModelica, разницы особой незаметно. Если сильно вглядываться, можно заметить, что в OpenModelica график чуть более плавный и точный.

Выводы

Рассмотрели простейшие модели боевых действий и смоделировать их на языках программирования Julia и Modelica.

Список литературы
