

Отчет по лабораторной работе № 6

По дисциплине Математическое Моделирование

Максимов Алексей Александрович

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.0.1	на Julia	9
4.0.2	на OpenModelica	12
5	Выводы	15

Список иллюстраций

2.1	image	6
4.1	image	9
4.2	image	10
4.3	image	11
4.4	image	12
4.5	image	13
4.6	image	14

Список таблиц

1 Цель работы

Ознакомиться с языком программирования Julia и OpenModelica и решить задачу об эпидемии.

2 Задание

Вариант 32

На одном острове вспыхнула эпидемия. Известно, что из всех проживающих на острове ($N=1000$) в момент начала эпидемии ($t=0$) число заболевших людей (являющихся распространителями инфекции) $I(0)=290$, А число здоровых людей с иммунитетом к болезни $R(0)=52$. Таким образом, число людей восприимчивых к болезни, но пока здоровых, в начальный момент времени $S(0)=N-I(0)-R(0)$.

Постройте графики изменения числа особей в каждой из трех групп. Рассмотрите, как будет протекать эпидемия в случае:

- 1) если $I(0) \leq I^*$
- 2) если $I(0) > I^*$

Рис. 2.1: image

3 Теоретическое введение

Рассмотрим простейшую модель эпидемии. Предположим, что некая популяция, состоящая из N особей, (считаем, что популяция изолирована) подразделяется на три группы. Первая группа - это восприимчивые к болезни, но пока здоровые особи, обозначим их через $S(t)$. Вторая группа – это число инфицированных особей, которые также при этом являются распространителями инфекции, обозначим их $I(t)$. А третья группа, обозначающаяся через $R(t)$ – это здоровые особи с иммунитетом к болезни.

4 Выполнение лабораторной работы

4.0.1 на Julia

```
lab6.jl
13 S0 = N - I0 - R0
14
15 t0 = 0
16 tmax = 200
17
18
19 t = collect(LinRange(t0, tmax, 4000))
20
21 function syst(dx,x,p,t)
22     dx[1] = 0
23     dx[2] = -b * x[2]
24     dx[3] = b * x[2]
25 end
26
27 function syst2(dx, x, p, t)
28     dx[1] = -a*x[1]
29     dx[2] = -a*x[1] - b*x[2]
30     dx[3] = b*x[2]
31 end
32
33 t0 = 0
34
35 x0 = [S0 ; I0; R0]
36
37 y = ODEProblem(syst, x0, (t0, tmax))
38
39 #y = ODEProblem( функция, нач, диапазон)
40
41 sol = solve(y)
42
43 plot(sol)
44
45 savefig("C:\\Users\\maksi\\OneDrive\\Рабочий стол\\unik2.0\\Математическое
Моделирование\\julia\\juliaab4jl06.png")
46
47 y2 = ODEProblem(syst, x0, (t0, tmax))
48
49 #y = ODEProblem( функция, нач, диапазон)
50
51 sol = solve(y2)
52 plot(sol)
53
54
55 savefig("C:\\Users\\maksi\\OneDrive\\Рабочий стол\\unik2.0\\Математическое
Моделирование\\julia\\juliaab4jl06.2.png")
```

Рис. 4.1: image

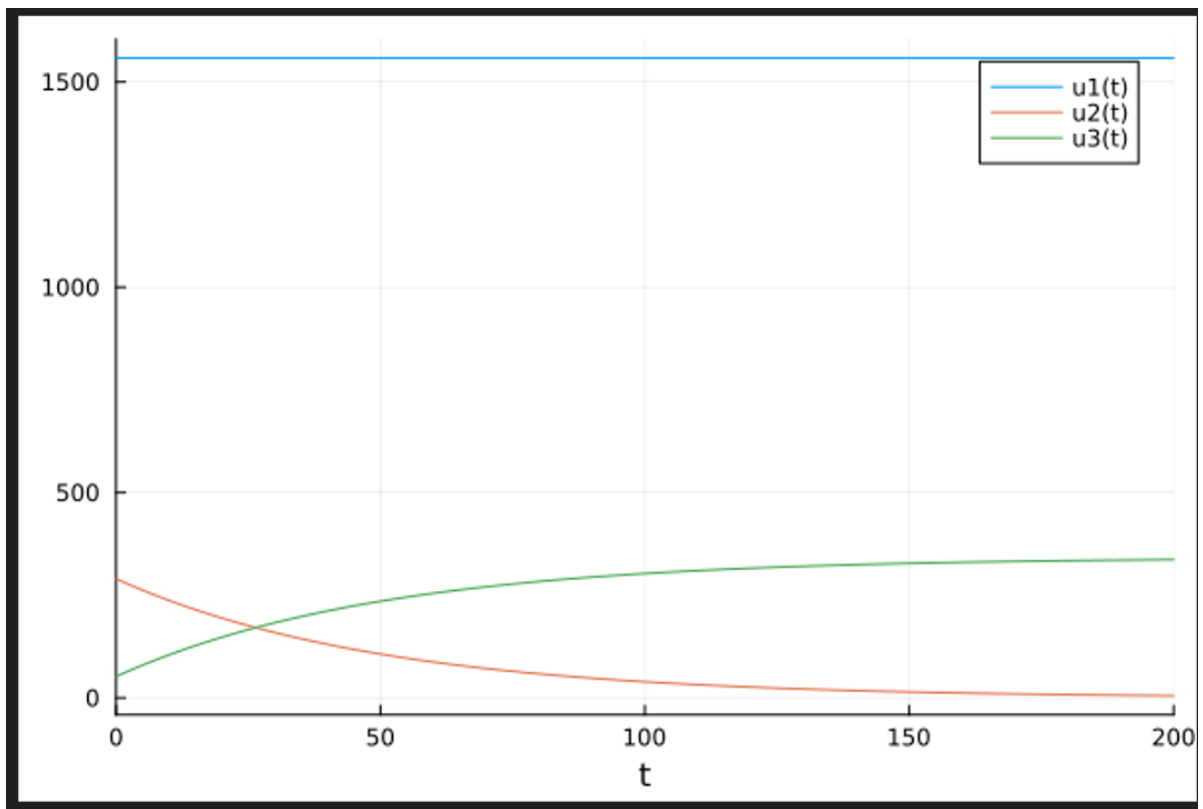


Рис. 4.2: image

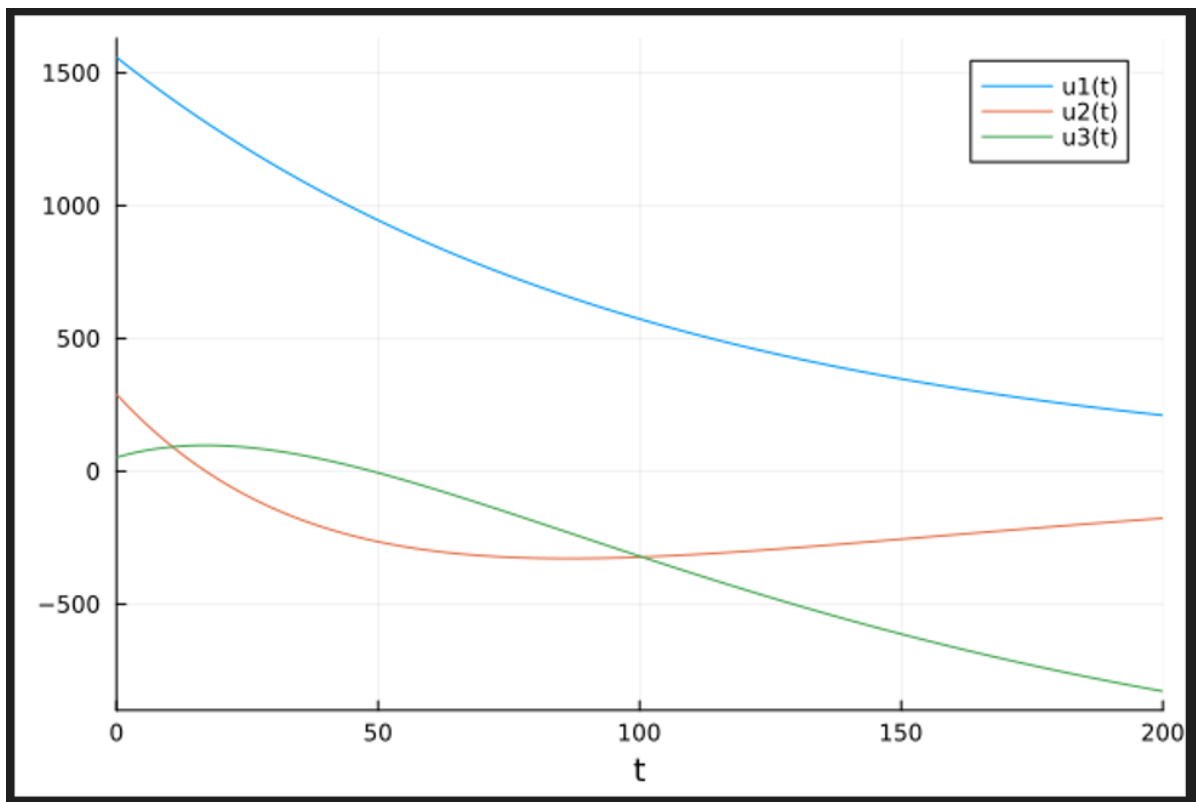


Рис. 4.3: image

4.0.2 на OpenModelica

```
1 model lab6
2   parameter Real a=0.01;
3   parameter Real b=0.02;
4   parameter Real N=1900;
5   parameter Real I0=290;
6   parameter Real R0=52;
7   parameter Real S0=N-I0-R0;
8   Real S(start=S0) "количество восприимчивых к болезни";
9   Real I(start=I0) "количество инфицированных";
10  Real R(start=R0) "количество особей с иммунитетом";
11
12  Real S2(start=S0) "количество восприимчивых к болезни 2";
13  Real I2(start=I0) "количество инфицированных 2";
14  Real R2(start=R0) "количество особей с иммунитетом 2";
15  equation
16    der(S) = 0;
17    der(I) = -b*I;
18    der(R) = b*I;
19  equation
20    der(S2) = -a*S2;
21    der(I2) = -a*S2-b*I2;
22    der(R2) = b*I2;
23 end lab6;
```

Рис. 4.4: image

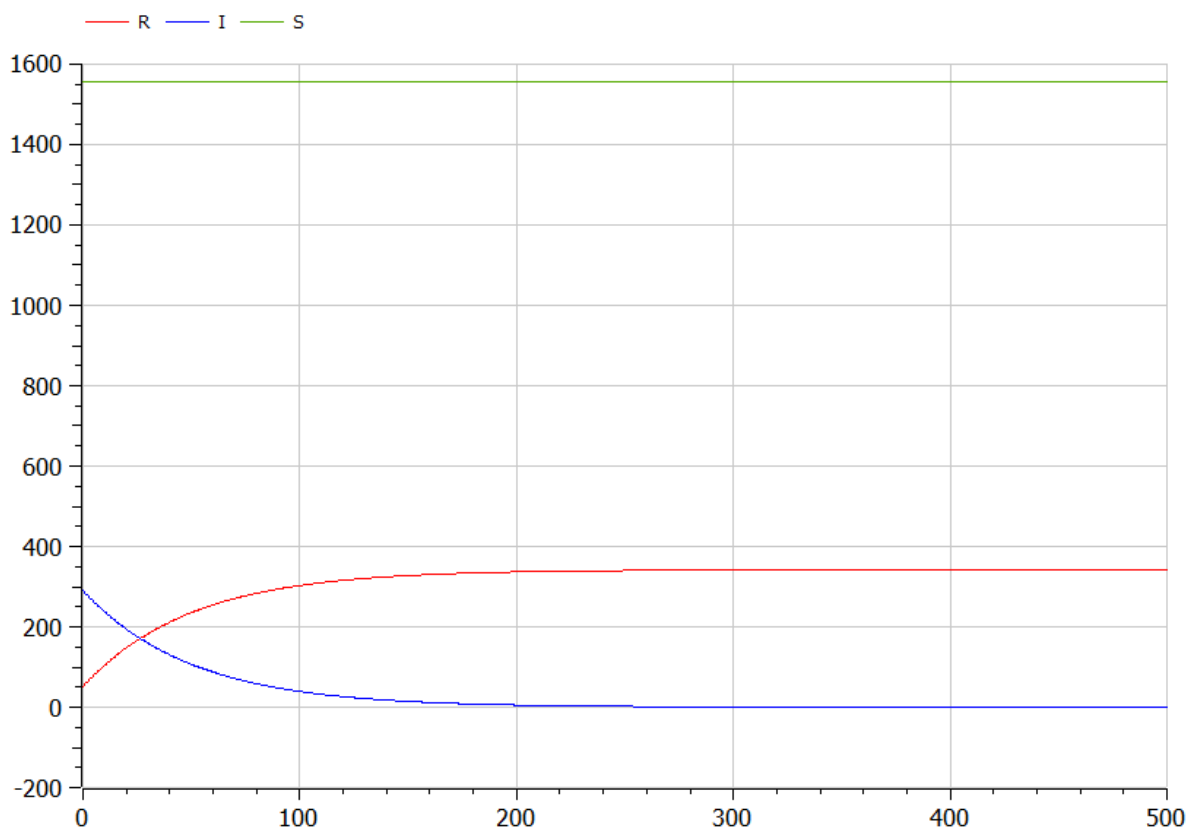


Рис. 4.5: image

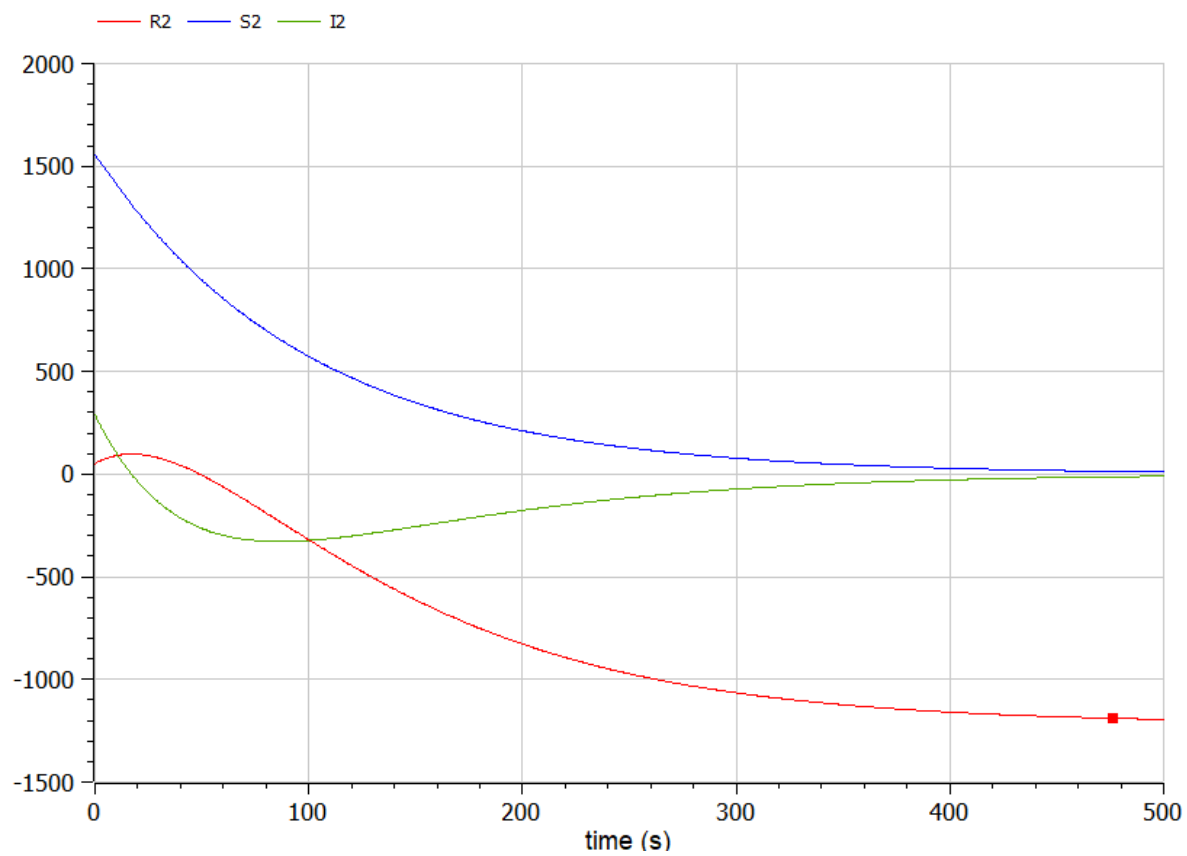


Рис. 4.6: image

5 Выводы

Решили задачу и написали программу на Julia и OpenModelica