

Лабораторная работа 5

Модель ЭПИДЕМИИ SIR

Извекова Мария Петровна

11 апрель 2025

Российский университет дружбы народов, Москва, Россия

- Извекова Мария Петровна
- студентка 3-го курса
- Российский университет дружбы народов
- 1132226460@pfur.ru



Построить модель SIR в xcos и OpenModelica.

1. Реализовать модель SIR в в xcos;
2. Реализовать модель SIR с помощью блока Modelica в в xcos;
3. Реализовать модель SIR в OpenModelica;
4. Реализовать модель SIR с учётом процесса рождения / гибели особей в xcos (в том числе и с использованием блока Modelica), а также в OpenModelica;
5. Построить графики эпидемического порога при различных значениях параметров модели (в частности изменяя параметр μ);
6. Сделать анализ полученных графиков в зависимости от выбранных значений параметров модели.

$$\begin{cases} \dot{s}(t) = -\beta s(t)i(t), \\ \dot{i}(t) = \beta s(t)i(t) - \nu i(t), \\ \dot{r}(t) = \nu i(t) \end{cases}$$

где β - скорость распространения, ν - скорость выздоровления

Зафиксируем начальные данные: $\beta = 1$ $\nu = .3$ $s(0)=0.999$, $r(0)=0$, $i(0)=0.001$ В меню Моделирование, Установить контекст зададим значения переменных β и ν (рис. [-@fig:001]).

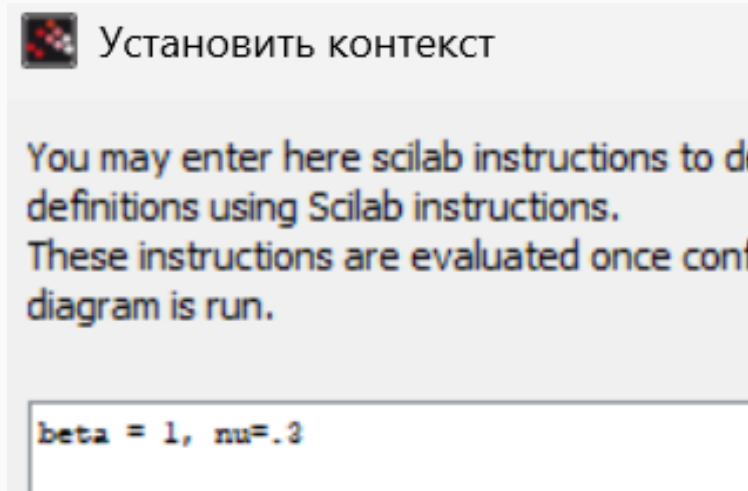


Рис. 1: Фиксирование переменных

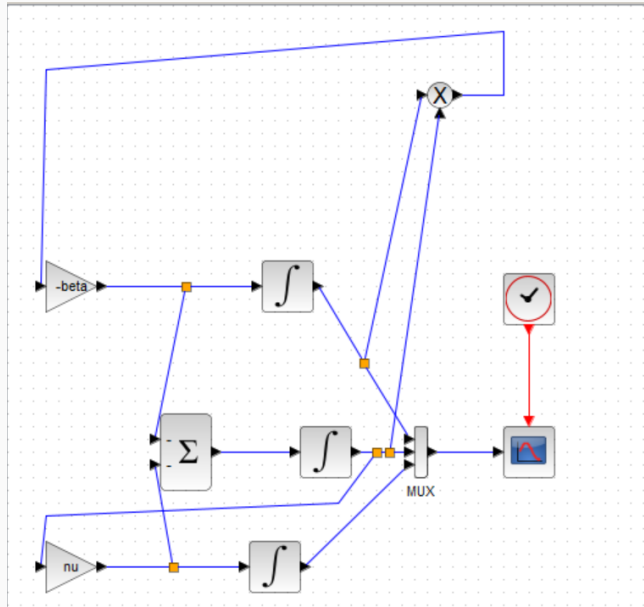


Рис. 2: Готовая модель

Ввод значений

Установите параметры блока INTEGRAL_m

| | |
|--------------------------------------|-----------------------------------|
| Initial Condition | <input type="text" value=".999"/> |
| With re-initialization (1:yes, 0:no) | <input type="text" value="0"/> |
| With saturation (1:yes, 0:no) | <input type="text" value="0"/> |
| Upper limit | <input type="text" value="1"/> |
| Lower limit | <input type="text" value="-1"/> |

OK Отменить

Ввод значений

Установите параметры блока INTEGRAL_m

| | |
|--------------------------------------|-----------------------------------|
| Initial Condition | <input type="text" value=".001"/> |
| With re-initialization (1:yes, 0:no) | <input type="text" value="0"/> |
| With saturation (1:yes, 0:no) | <input type="text" value="0"/> |
| Upper limit | <input type="text" value="1"/> |
| Lower limit | <input type="text" value="-1"/> |

OK Отменить

Ввод значений

Установите параметры блока INTEGRAL_m

| | |
|--------------------------------------|---------------------------------|
| Initial Condition | <input type="text" value="0"/> |
| With re-initialization (1:yes, 0:no) | <input type="text" value="0"/> |
| With saturation (1:yes, 0:no) | <input type="text" value="0"/> |
| Upper limit | <input type="text" value="1"/> |
| Lower limit | <input type="text" value="-1"/> |

Ввод значений

Установите параметры блока SUMMATION

| | |
|---|--------------------------------------|
| Datatype (1=real double 2=complex 3=int32 ..) | <input type="text" value="1"/> |
| Number of inputs or sign vector (of +1, -1) | <input type="text" value="[-1;-1]"/> |
| Do on Overflow (0=Nothing 1=Saturate 2=Error) | <input type="text" value="0"/> |

параметры моделирования

| | |
|---|-------------------------------|
| Конечное время интегрирования | 3.0E01 |
| Количество секунд в единице времени | 0.0E00 |
| Абсолютная погрешность интегрирования | 1.0E-06 |
| Относительная погрешность интегрирования | 1.0E-06 |
| Погрешность по времени | 1.0E-10 |
| Максимальный временной интервал интегрирования | 1.00001E05 |
| Вид программы решения | Sundials/CVODE - BDF - NEWTON |
| Максимальный размер шага (0 означает "без ограничения") | 0.0E00 |

Рис. 3: Конечное время интегрирования

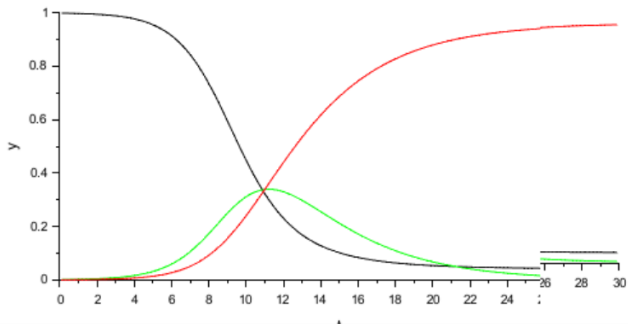


Рис. 4: Результат моделирования

Реализация модели с помощью блока Modelica в xcos

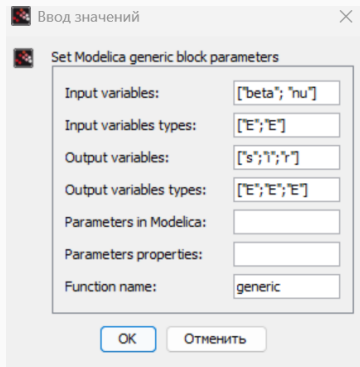


Рис. 5: Фиксированные переменные

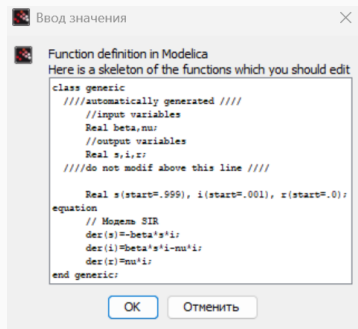


Рис. 6: Функция generic

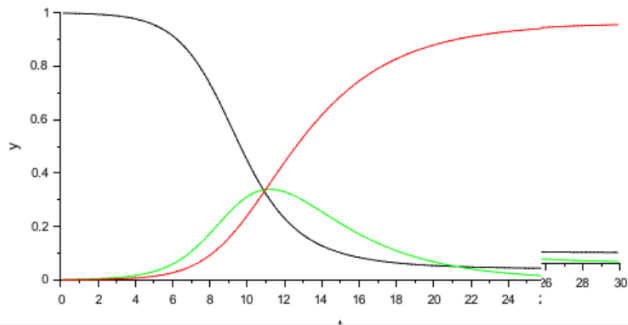


Рис. 7: Результат моделирования

```
modelica parameter Real I_0 = 0.001; parameter Real R_0 = 0; parameter Real S_0 = 0.999;  
parameter Real beta = 1; parameter Real nu = 0.3; Real s(start=S_0); Real i(start=I_0); Real  
r(start=R_0); equation der(s) = -beta * s * i; der(i) = beta * s * i - nu * i; der(r) = nu * i;
```

| | | | |
|-------|-----------------------------|------------------|------------|
| Общее | Интерактивное моделирование | Флаги трансляции | Флаги моде |
|-------|-----------------------------|------------------|------------|

Интервал моделирования

Время начала:

0

Время завершения:

30

☒ Число интервалов:

500

☐ Интервал:

0.002

Интегрирование

Метод:

class

Рис. 8: Время симмуляции

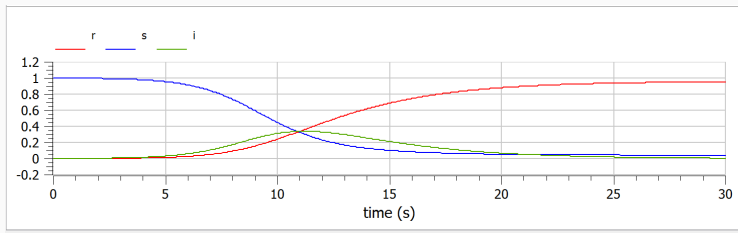


Рис. 9: Результат моделирования

$$\begin{cases} \dot{s}(t) = -\beta s(t)i(t) + \mu(N - s(t)), \\ \dot{i}(t) = \beta s(t)i(t) - \nu i(t) - \mu i(t), \\ \dot{r}(t) = \nu i(t) - \mu r(t) \end{cases}$$

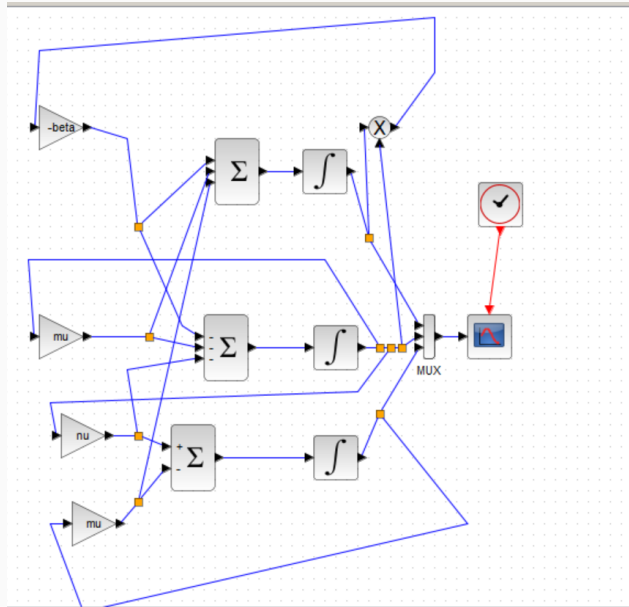


Рис. 10: Готовая модель



Установить контекст

You may enter here scilab instructions to define symbolic parameter definitions using Scilab instructions.

These instructions are evaluated once confirmed (i.e. you click diagram is run).

```
beta = 1, nu=.3, mu=.1
```

Рис. 11: Задаем параметры

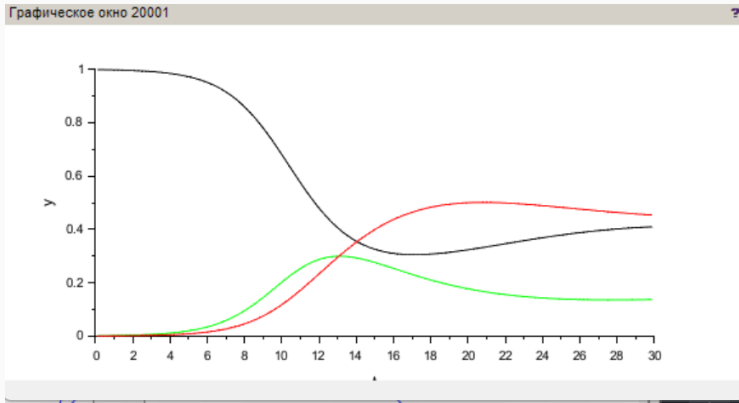


Рис. 12: Результат моделирования

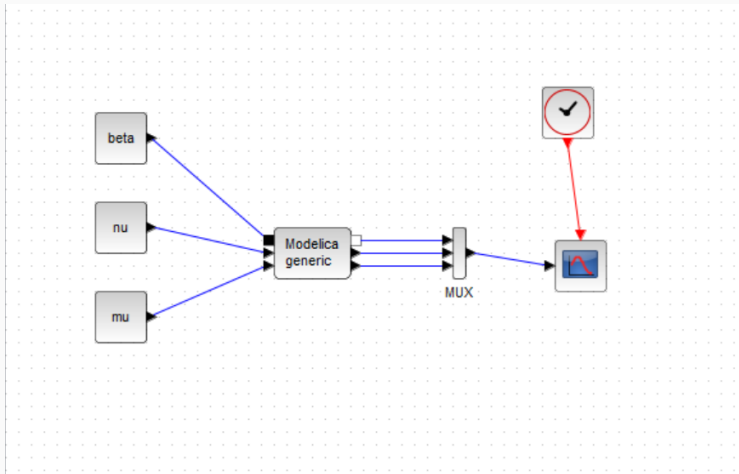


Рис. 13: Модель с блоком

Ввод значений

Set Modelica generic block parameters

Input variables: ["beta"; "mu"; "nu"]

Input variables types: ["E"; "E"; "E"]

Output variables: ["s"; "i"; "r"]

Output variables types: ["E"; "E"; "E"]

Parameters in Modelica:

Parameters properties:

Function name: generic

OK Отменить

Ввод значения

Function definition in Modelica

Here is a skeleton of the functions which you should edit

```
class generic
  ///automatically generated ///
  //input variables
  Real beta,mu,nu;
  //output variables
  Real s,i,r;
  ///do not modif above this line ///

  Real s(start=.999), i(start=.001), r(start=.0);
equation
  // Modela SIR
  der(s)=-beta*s*i+mu*i+mu*r;
  der(i)=beta*s*i-nu*i-mu*i;
  der(r)=nu*i-mu*r;
end generic;
```

OK Отменить

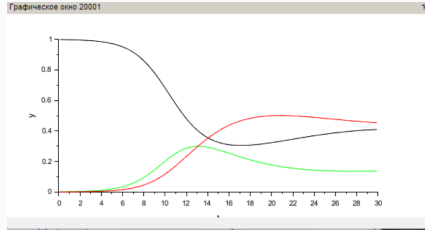


Рис. 14: Результат моделирования при $\mu=0.1$

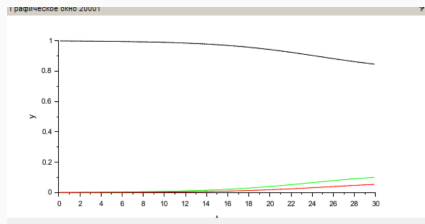


Рис. 15: Результат моделирования при $\mu=0.9$

```
modelica parameter Real I_0 = 0.001; parameter Real R_0 = 0; parameter Real S_0 = 0.999;  
parameter Real N = 1; parameter Real beta = 1; parameter Real nu = 0.3; parameter Real mu = 0.5;  
Real s(start=S_0); Real i(start=I_0); Real r(start=R_0); equation der(s) = -beta * s * i + mu * i + mu *  
r; der(i) = beta * s * i - nu * i - mu * i; der(r) = nu * i - mu * r;
```

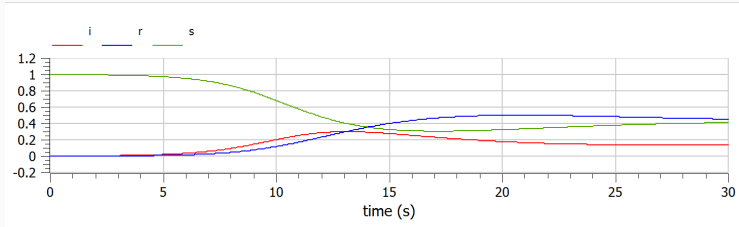



Рис. 16: Результат моделирования при $\mu=0.1$

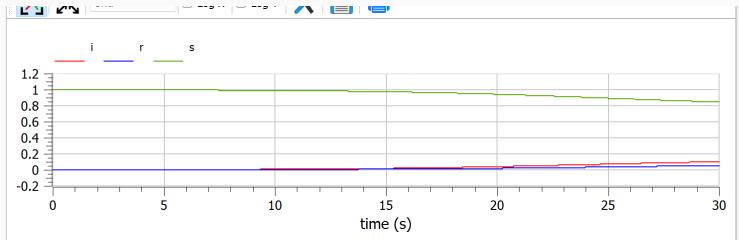


Рис. 17: Результат моделирования при $\mu=0.9$

В процессе выполнения данной лабораторной работы была построена модель SIR в xcos и OpenModelica