

Лабораторная работа №5

Имитационное моделирование

Екатерина Канева, НФИбд-02-22

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	18

Список иллюстраций

4.1	Модель 1 в xcoss.	8
4.2	Параметры верхнего блока интегрирования.	9
4.3	Параметры среднего блока интегрирования.	9
4.4	Длительность моделирования.	10
4.5	График модели 1 в xcoss.	10
4.6	Модель 1 в xcoss (с блоком Modelica).	11
4.7	Параметры блока Modelica.	11
4.8	Код для блока Modelica.	12
4.9	Модель 1 в OpenModelica.	13
4.10	Модель 2 в xcoss.	14
4.11	График модели 2 в xcoss с $\mu = 0.1$	14
4.12	График модели 2 в xcoss с $\mu = 0.05$	15
4.13	Модель 2 в xcoss (с блоком Modelica).	15
4.14	График модели 2 в xcoss с $\mu = 0.1$	17
4.15	График модели 2 в xcoss с $\mu = 0.05$	17

Список таблиц

1 Цель работы

Построить модель SIR в xcos и OpenModelica.

2 Задание

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.

3 Теоретическое введение

Задача о распространении эпидемии описывается системой дифференциальных уравнений:

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

где β – скорость заражения, ν – скорость выздоровления.

Если брать в расчёт демографические процессы (рождение и гибель), то система выглядит так:

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

где μ – константа, которая равна коэффициенту смертности и рождаемости.

4 Выполнение лабораторной работы

Для начала я построила модель из блоков в xcos (рис. 4.1).

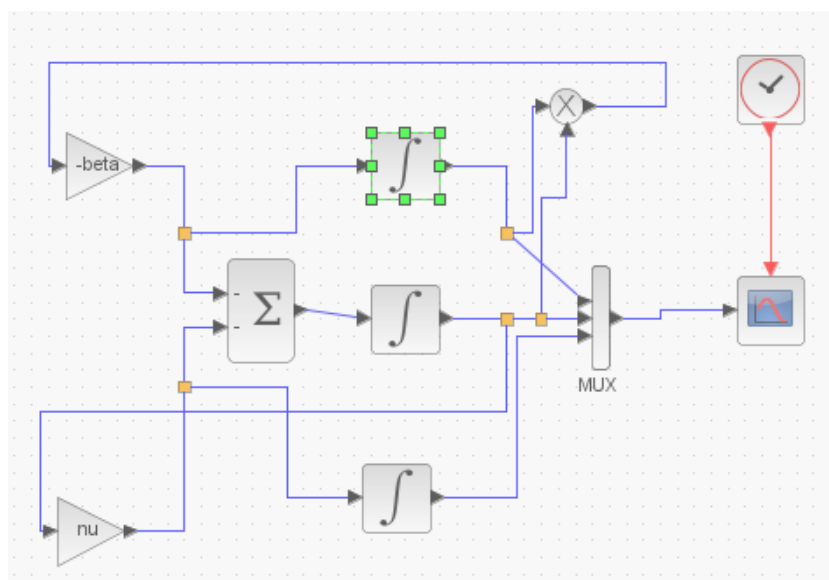


Рис. 4.1: Модель 1 в xcos.

Задала следующие параметры для блоков интегрирования для верхнего рис. 4.2, для среднего рис. 4.3:

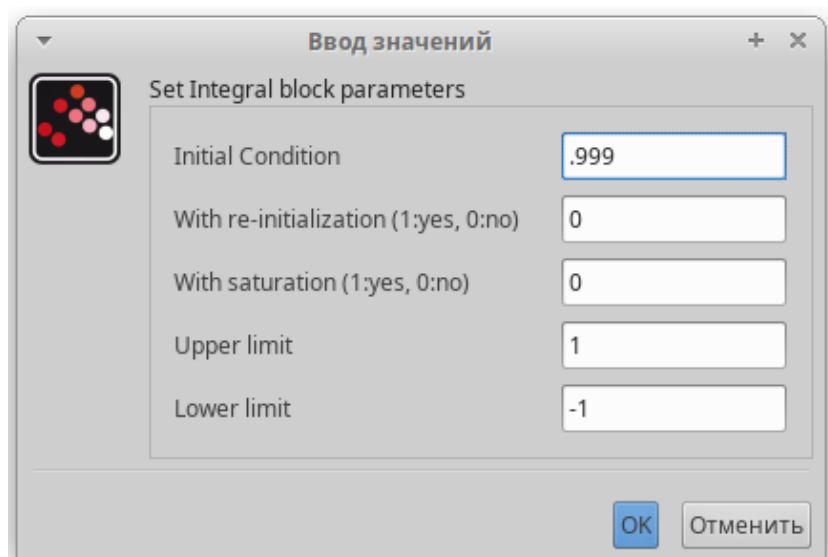


Рис. 4.2: Параметры верхнего блока интегрирования.

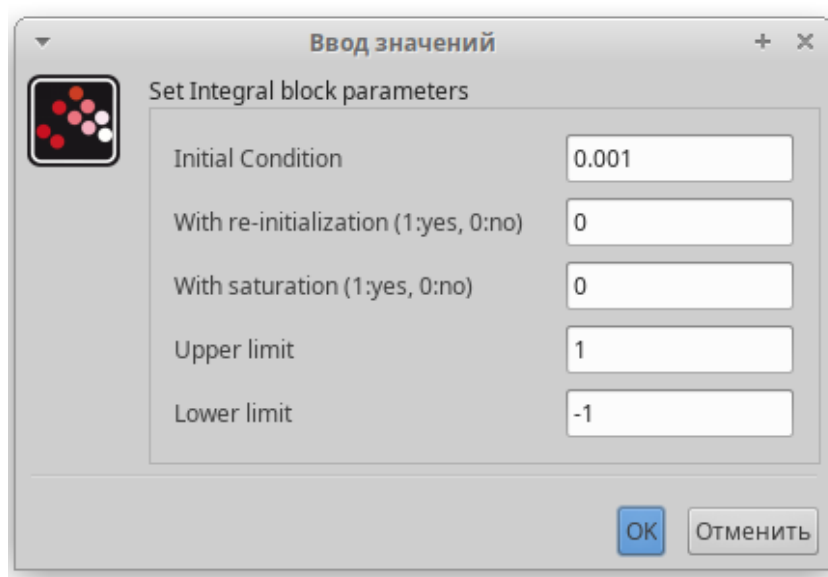


Рис. 4.3: Параметры среднего блока интегрирования.

Задала длительность моделирования (рис. 4.4):

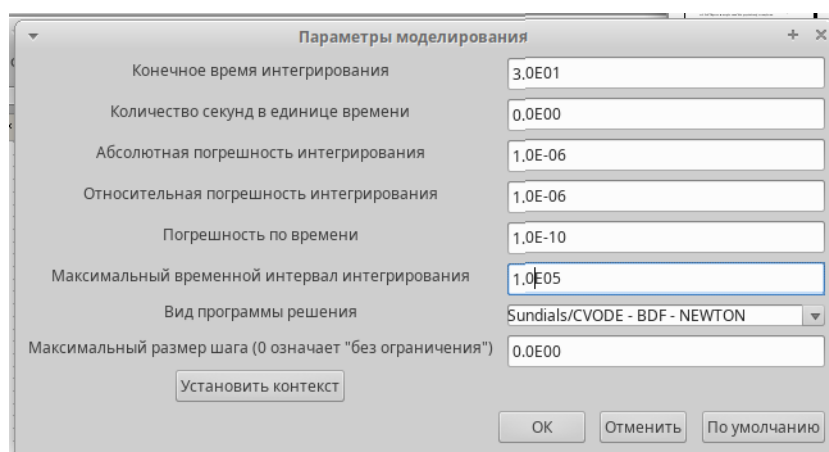


Рис. 4.4: Длительность моделирования.

Запустила моделирование, получила следующий график (рис. 4.5):

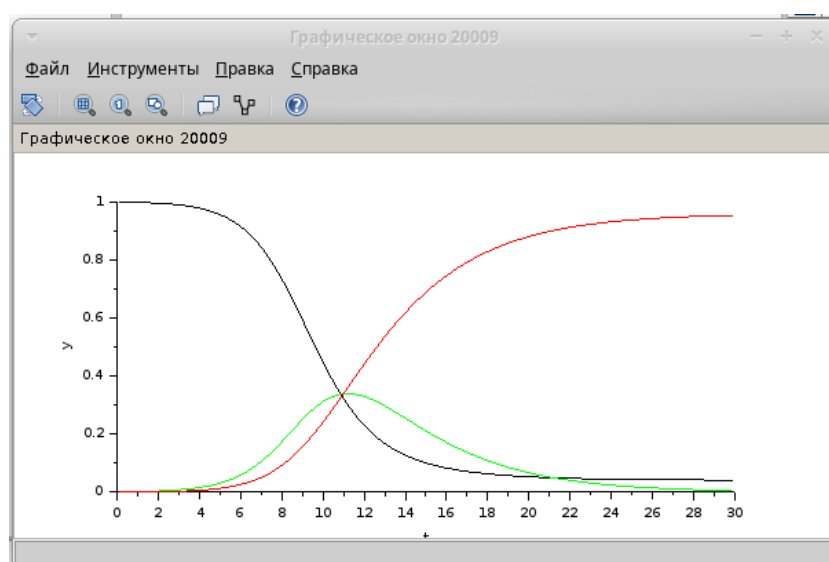


Рис. 4.5: График модели 1 в xcos.

Далее построила то же самое, но с блоком Modelica (рис. 4.6):

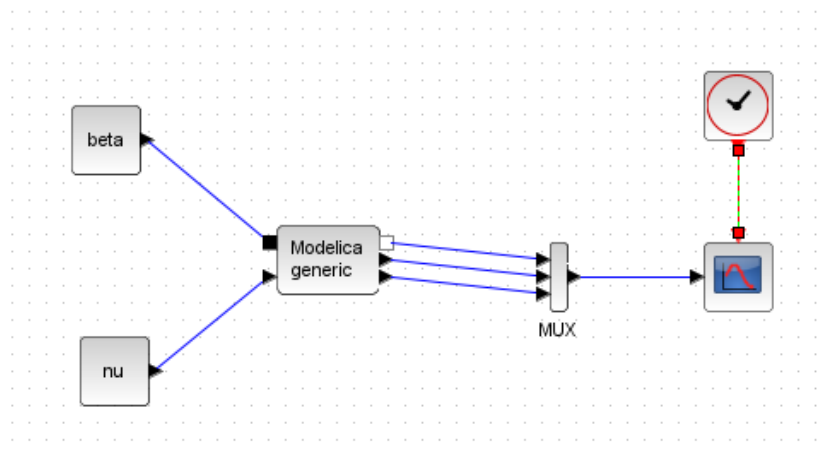


Рис. 4.6: Модель 1 в xcos (с блоком Modelica).

Задала параметры для блока Modelica (рис. 4.7) и написала код для этого блока (рис. 4.8):

The screenshot shows a dialog box titled 'Ввод значений' (Input values) with a subtitle 'Set Modelica generic block parameters'. The dialog contains several input fields for configuring the block:

- Input variables: ["beta","nu"]
- Input variables types: ["E","E"]
- Output variables: ["s","i","r"]
- Output variables types: ["E","E","E"]
- Parameters in Modelica: (empty field)
- Parameters properties: (empty field)
- Function name: generic

At the bottom right, there are 'OK' and 'Отменить' (Cancel) buttons. A status bar at the bottom shows a scale from 4 to 28.

Рис. 4.7: Параметры блока Modelica.

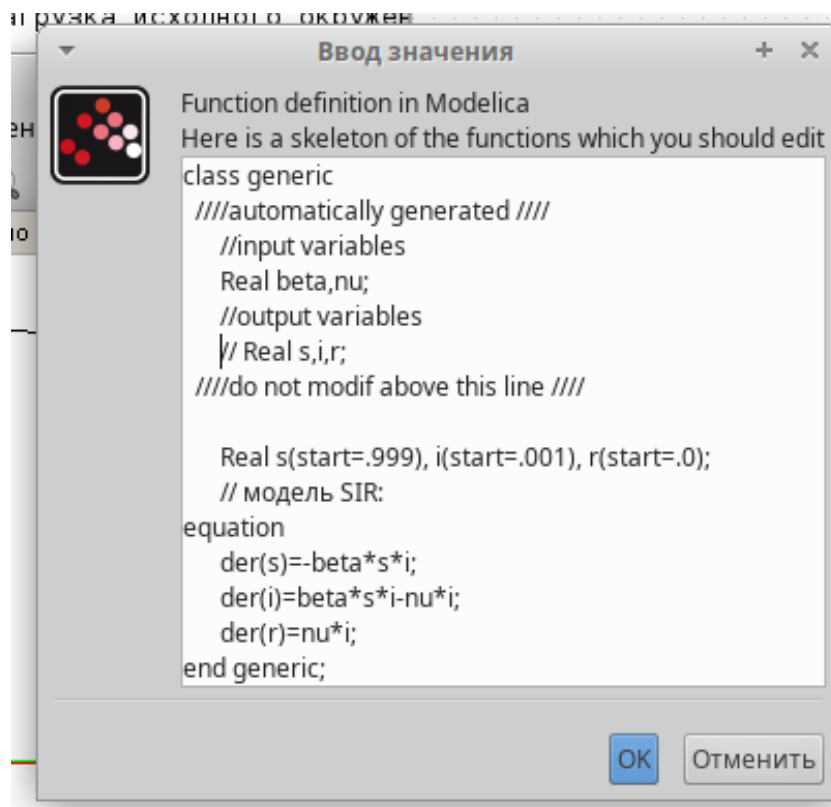


Рис. 4.8: Код для блока Modelica.

Получила такой же график, как на рис. 4.5.

Далее я выполнила то же самое в OpenModelica в качестве упражнения. Там я ввела следующий код:

```
model lab51
  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

$\text{der}(s) = -\beta \cdot s \cdot i;$

$\text{der}(i) = \beta \cdot s \cdot i - \nu \cdot i;$

$\text{der}(r) = \nu \cdot i;$

end lab51;

При запуске построился следующий график (рис. 4.9):

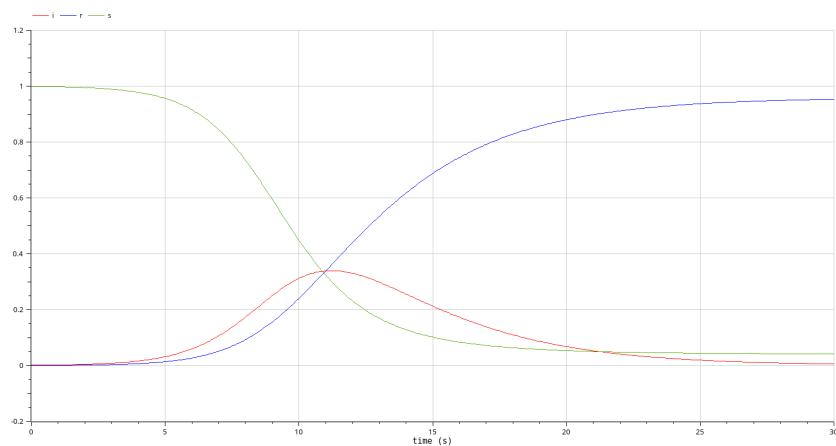


Рис. 4.9: Модель 1 в OpenModelica.

График такой же как при построении в xcos.

Потом я присупила к выполнению части с учётом демографии. Для этого я изменила схему в xcos (4.10):

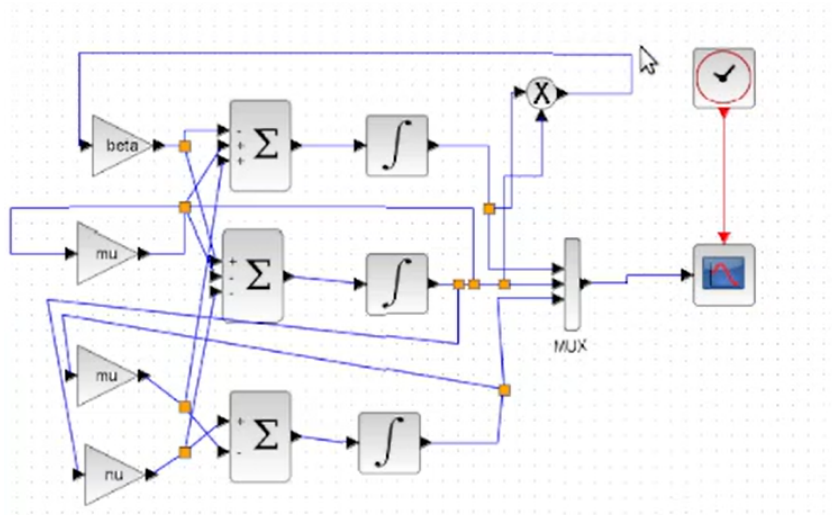


Рис. 4.10: Модель 2 в xcos.

При запуске с $\mu = 0.1$ получили график на рис. 4.11, при запуске с $\mu = 0.05$ получили график на рис. 4.12:

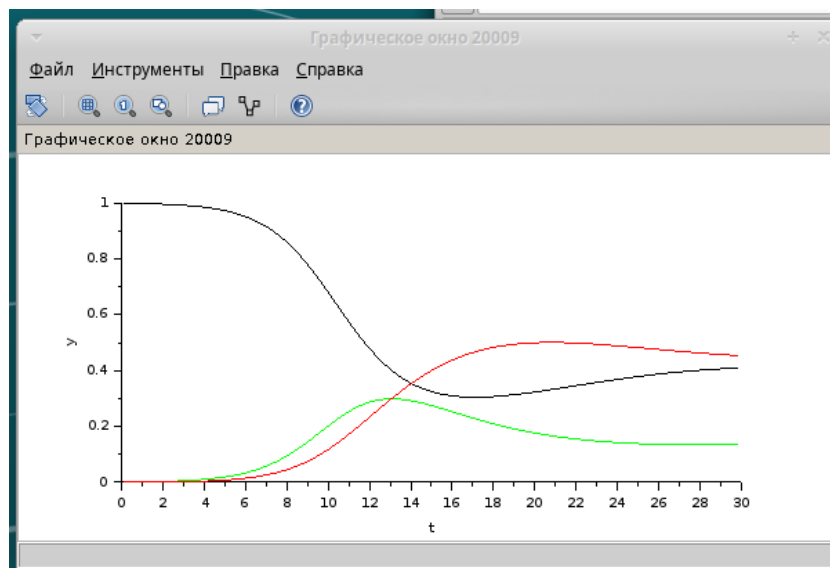


Рис. 4.11: График модели 2 в xcos с $\mu = 0.1$.

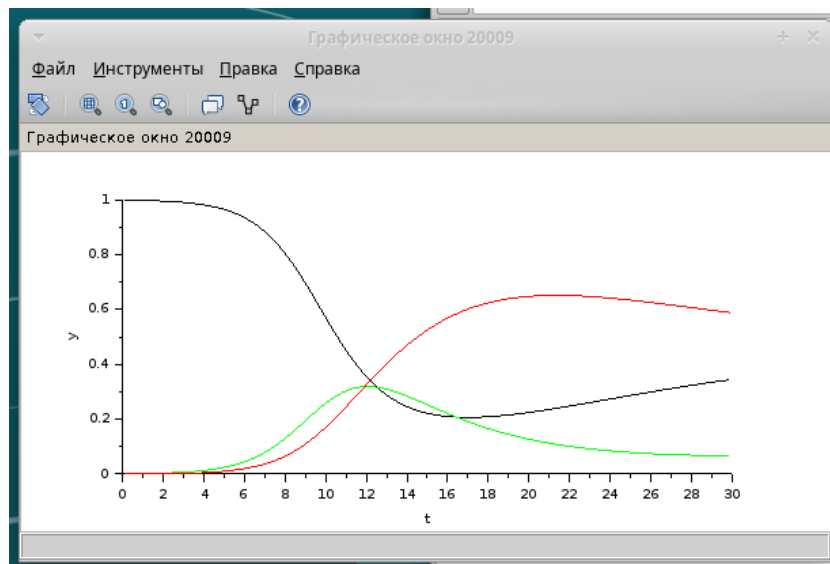


Рис. 4.12: График модели 2 в xcos с $\mu_i = 0.05$.

Далее я построила то же самое с блоком Modelica (рис. 4.8):

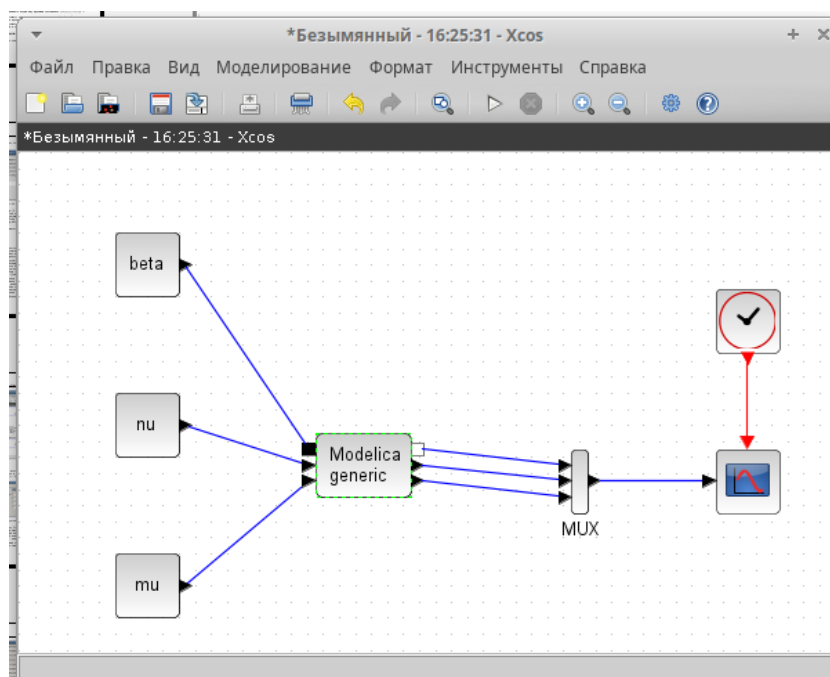


Рис. 4.13: Модель 2 в xcos (с блоком Modelica).

Изменила код и параметры блока, график получился такой же, как и без блока Modelica (рис. 4.11 и 4.12).

Далее я реализовала эту модель в OpenModelica. Для этого я отредактировала предыдущий код и получила это:

```
model lab52
  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;
  parameter Real mu = 0.1;

  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;

end lab52;
```

При запуске с $\mu = 0.1$ получился график на рис. 4.14, при запуске с $\mu = 0.05$ получили график на рис. 4.15:

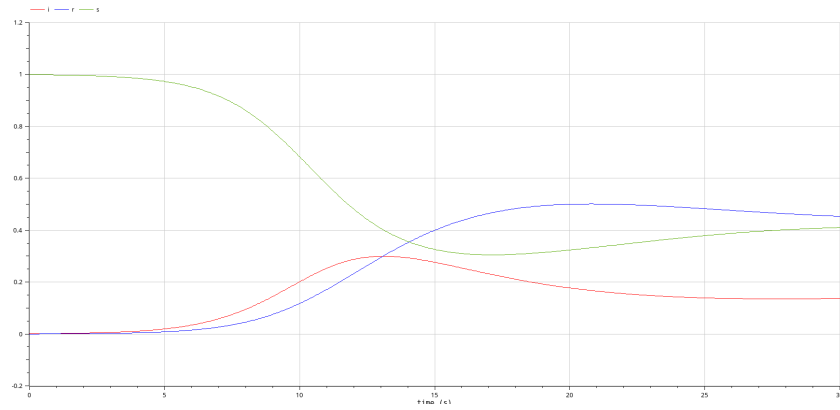


Рис. 4.14: График модели 2 в xcos с $\mu = 0.1$.

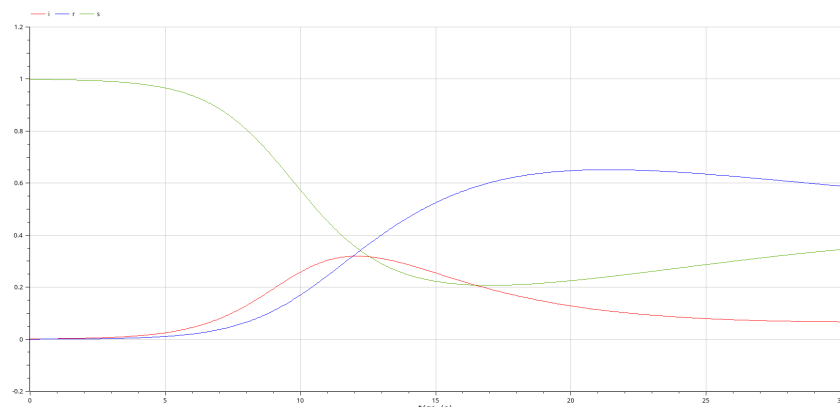


Рис. 4.15: График модели 2 в xcos с $\mu = 0.05$.

Видно, что чем больше коэффициент μ , тем сильнее график отличается от начального.

5 Выводы

Построили модель SIR в xcos и OpenModelica.