

# **Лабораторная работа № 5**

**Модель эпидемии (SIR)**

Дворкина Ева Владимировна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Реализация модели в xcos . . . . .	8
4.2	Реализация модели с помощью блока Modelica в xcos . . . . .	14
4.3	Реализация модели в OpenModelica . . . . .	16
<b>5</b>	<b>Задание для самостоятельного выполнения</b>	<b>18</b>
5.1	Модель SIR с учетом демографии . . . . .	18
5.2	Реализация модели в xcos . . . . .	19
5.3	Реализация модели с помощью блока Modelica в xcos . . . . .	21
5.4	Реализация модели в OpenModelica . . . . .	24
5.5	Анализ графиков при разных параметрах модели . . . . .	25
<b>6</b>	<b>Выводы</b>	<b>29</b>
	<b>Список литературы</b>	<b>30</b>

# Список иллюстраций

4.1	Задать контекст в xcos . . . . .	8
4.2	Задать количество входов в мультиплексор . . . . .	9
4.3	Ввод параметров блока суммирования . . . . .	10
4.4	Модель SIR в xcos . . . . .	11
4.5	Задать начальное значение в блоке интегрирования . . . . .	11
4.6	Задать начальное значение в блоке интегрирования . . . . .	12
4.7	Задать конечное время интегрирования в xcos . . . . .	12
4.8	Задать значения отображения графиков в регистрирующем устройстве . . . . .	13
4.9	График решения модели SIR при $\beta = 1, \nu = 0.3$ . . . . .	13
4.10	Настройка параметров блока Modelica . . . . .	14
4.11	Настройка параметров блока Modelica . . . . .	15
4.12	Модель SIR в xcos с применением блока Modelica . . . . .	15
4.13	График решения модели SIR при $\beta = 1, \nu = 0.3$ . . . . .	16
4.14	Модель в OpenModelica . . . . .	16
4.15	График решения модели SIR при $\beta = 1, \nu = 0.3$ . . . . .	17
5.1	Задать переменные окружения в xcos . . . . .	19
5.2	Модель SIR с учетом демографии в xcos . . . . .	20
5.3	График решения модели SIR с учетом демографии при $\beta = 1, \nu = 0.3, \mu = 0.1$ . . . . .	21
5.4	Настройка параметров блока Modelica . . . . .	22
5.5	Настройка параметров блока Modelica . . . . .	23
5.6	Модель SIR в xcos с применением блока Modelica . . . . .	23
5.7	График решения модели SIR с учетом демографии при $\beta = 1, \nu = 0.3, \mu = 0.1$ . . . . .	24
5.8	Модель SIR с учетом демографии в OpenModelica . . . . .	24
5.9	Параметры моделирования в OpenModelica . . . . .	25
5.10	График решения модели SIR с учетом демографии при $\beta = 1, \nu = 0.3, \mu = 0.1$ . . . . .	25
5.11	График решения модели SIR с учетом демографии при $\beta = 1, \nu = 0.3, \mu = 0.3$ . OpenModelica . . . . .	26
5.12	График решения модели SIR с учетом демографии при $\beta = 1, \nu = 0.3, \mu = 0.5$ . OpenModelica . . . . .	26
5.13	График решения модели SIR с учетом демографии при $\beta = 1, \nu = 0.3, \mu = 0.05$ . OpenModelica . . . . .	27

5.14 График решения модели SIR с учетом демографии при $\beta = 1.5$ , $\nu = 0.2$ , $\mu = 0.2$ . OpenModelica . . . . .	27
5.15 График решения модели SIR с учетом демографии при $\beta = 1$ , $\nu = 0.5$ , $\mu = 0.2$ . OpenModelica . . . . .	28

# 1 Цель работы

Исследование модели эпидемии (SIR) с помощью xcos и OpenModelica.

## 2 Задание

- Реализовать классическую модель SIR с помощью xcos(в том числе с помощью блока Modelica) и OpenModelica.
- Реализовать модель SIR с учетом демографических признаков с помощью xcos(в том числе с помощью блока Modelica) и OpenModelica.
- Исследовать модель SIR с учетом демографических признаков, изменяя параметры.

### 3 Теоретическое введение

Модель SIR предложена в 1927 г. (W. O. Kermack, A. G. McKendrick).

$$\begin{cases} \frac{dS}{dt} = -\frac{\beta IS}{N}, \\ \frac{dI}{dt} = \frac{\beta IS}{N} - \nu I, \\ \frac{dR}{dt} = \nu I, \end{cases}$$

где  $S$  – численность восприимчивой популяции,  $I$  – численность инфицированных,  $R$  – численность удаленной популяции (в результате смерти или выздоровления), и  $N$  – это сумма этих трёх, а  $\beta$  и  $\nu$  – это коэффициенты заболеваемости и выздоровления соответственно [1].

## 4 Выполнение лабораторной работы

### 4.1 Реализация модели в xcoss

Зафиксируем начальные параметры в меню Моделирование, Задать переменные окружения, а затем построим модель при помощи блоков моделирования (рис. 4.1).

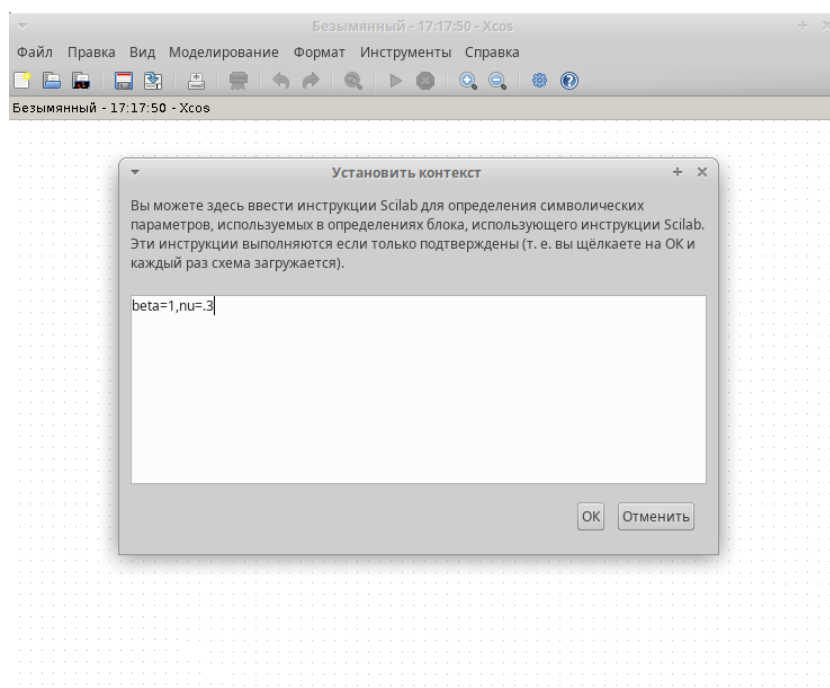


Рис. 4.1: Задать контекст в xcoss

Для реализации модели потребовались следующие блоки xcoss [2]:

- CLOCK\_c – запуск часов модельного времени;



- CSCOPE – регистрирующее устройство для построения графика;
- TEXT\_f – задаёт текст примечаний;
- MUX – мультиплексер, позволяющий в данном случае вывести на графике сразу несколько кривых;
- INTEGRAL\_m – блок интегрирования;
- GAINBLK\_f – в данном случае позволяет задать значения коэффициентов  $\beta$  и  $\gamma$ ;
- SUMMATION – блок суммирования;
- PROD\_f – поэлементное произведение двух векторов на входе блока.

Настраиваю количество входов в блок мультиплексер (рис. 4.2).

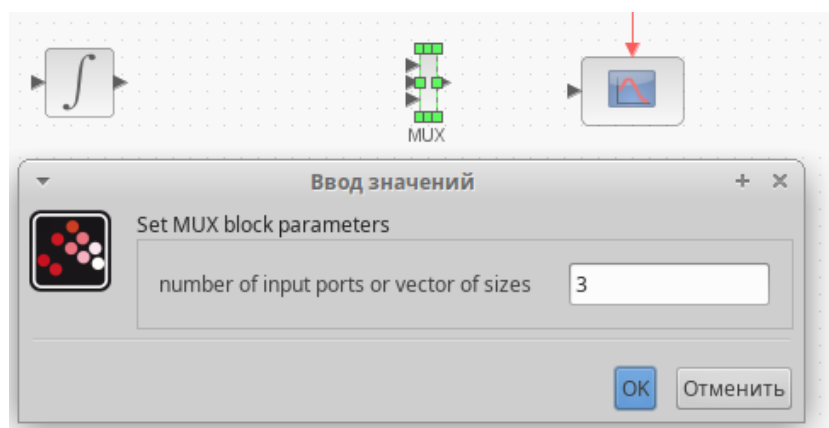


Рис. 4.2: Задать количество входов в мультиплексор

Настраиваю параметры блока суммирования, чтобы оба слагаемых в сумме на входе в интегратор были со знаком минус (рис. 4.3).

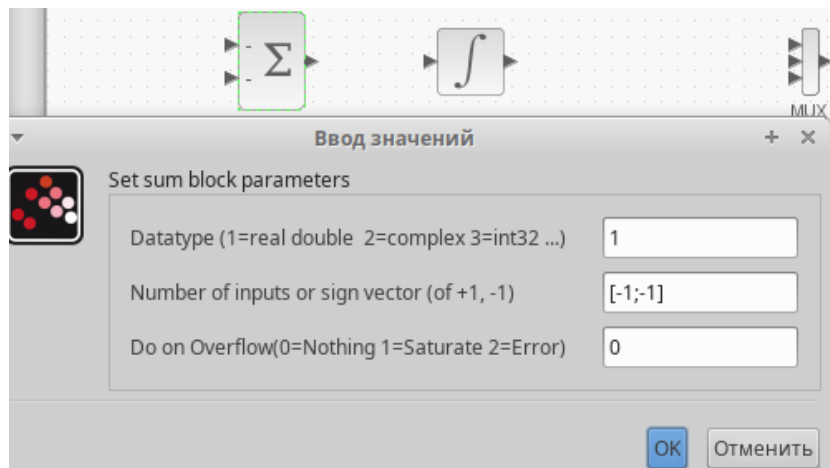


Рис. 4.3: Ввод параметров блока суммирования

Первое уравнение модели задано верхним блоком интегрирования, блоком произведения и блоком задания коэффициента  $\beta$ . Блок произведения соединен с выходами верхнего и среднего блоков интегрирования и блоком коэффициента  $\beta$ , что реализует математическую конструкцию  $\beta s(t)i(t)$ . Третье уравнение модели задано нижним блоком интегрирования и блоком задания коэффициента  $\nu$ . Для реализации математической конструкции  $\nu i(t)$  соединяем выход среднего блока интегрирования и вход блока задания коэффициента  $\nu$ , а результат передаём на вход нижнего блока интегрирования.

Средний блок интегрирования и блок суммирования определяют второе уравнение модели, которое по сути является суммой правых частей первого и третьего уравнений. Для реализации соединяем входы верхнего и нижнего блоков интегрирования с входами блока суммирования, меняя при этом в его параметрах оба знака на минус. Выход блока суммирования соединяем с входом среднего блока интегрирования (рис. 4.4).

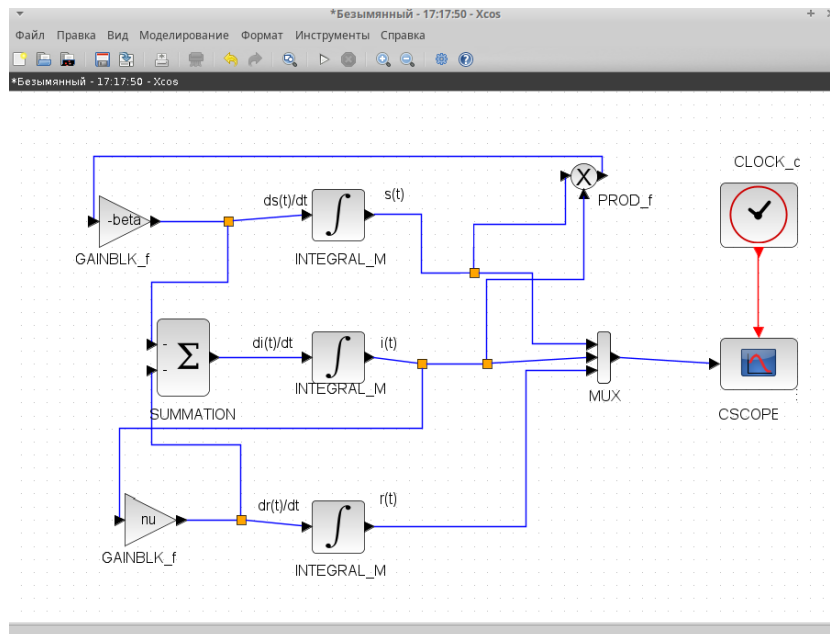


Рис. 4.4: Модель SIR в xcos

Зафиксируем начальные значения в блоках интегрирования (рис. 4.5, 4.6).

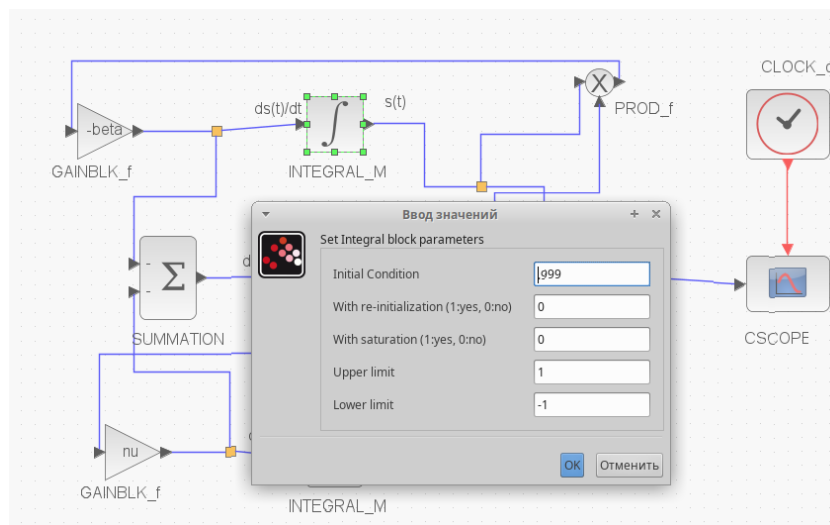


Рис. 4.5: Задать начальное значение в блоке интегрирования

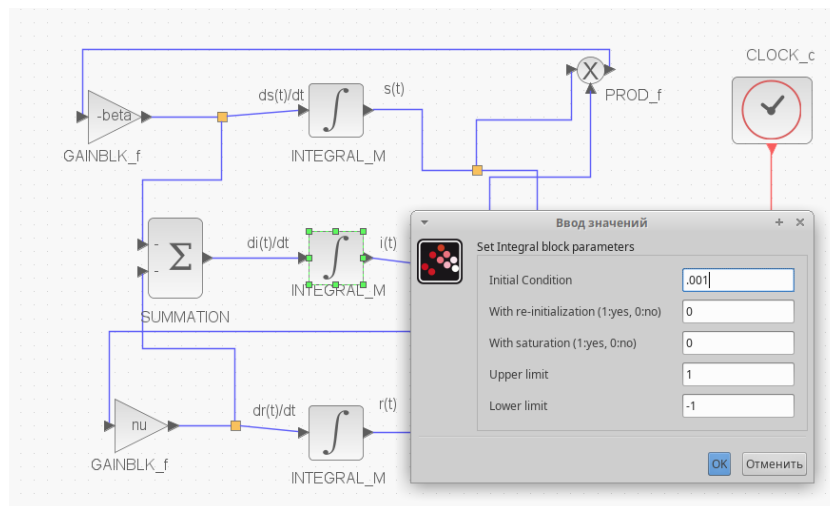


Рис. 4.6: Задать начальное значение в блоке интегрирования

Также зададим время интегрирования равное 30 единиц модельного времени (рис. 4.7).

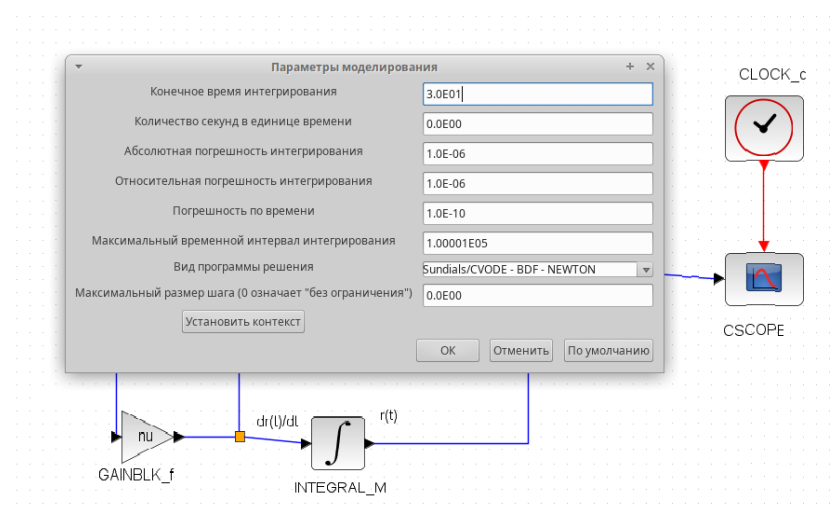


Рис. 4.7: Задать конечное время интегрирования в хcos

Настроим параметры регистрирующего устройства для отображения графика (рис. 4.8).

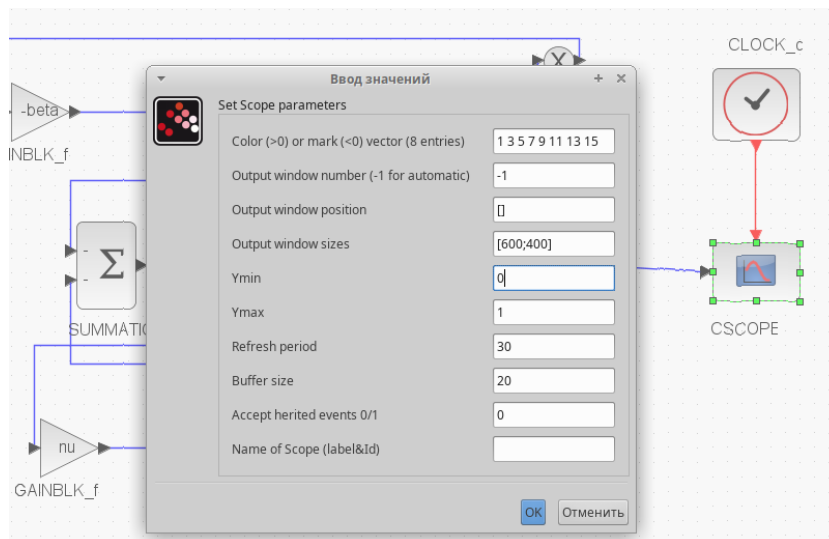


Рис. 4.8: Задать значения отображения графиков в регистрирующем устройстве

Решение модели SIR выглядит следующим образом (рис. 4.9).

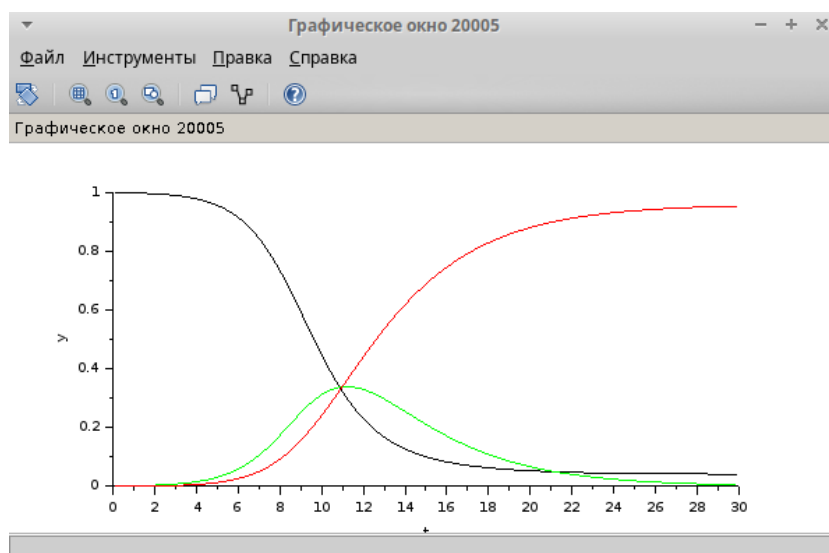


Рис. 4.9: График решения модели SIR при  $\beta = 1$ ,  $\nu = 0.3$

Видим, что точка пересечения всех функций - порог эпидемии, после которого количество заболевших уменьшается. Также на конец моделирование у нас остается некоторое количество уязвимых, которые не успели переболеть и больше не смогут заразиться, а все заболевшие выздоровели.

## 4.2 Реализация модели с помощью блока Modelica в Xcos

Для реализации модели с помощью языка Modelica помимо блоков CLOCK\_c, CSCOPE, TEXT\_f и MUX требуются блоки [2] :

- CONST\_m – задаёт константу;
- MBLOCK(Modelica generic) – блок реализации кода на языке Modelica.

Задаём значения переменных  $\beta$  и  $\nu$ . Параметры блока Modelica: переменные на входе ("beta", "nu") и выходе ("s", "i", "r") блока заданы как внешние ("E"). Затем прописываем дифференциальное уравнение (рис. 4.10, 4.11).

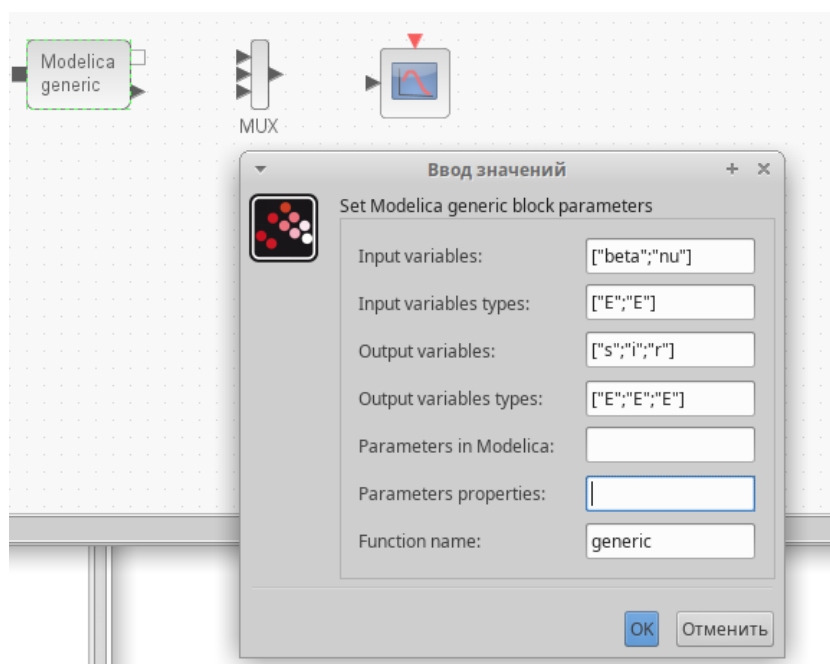


Рис. 4.10: Настройка параметров блока Modelica

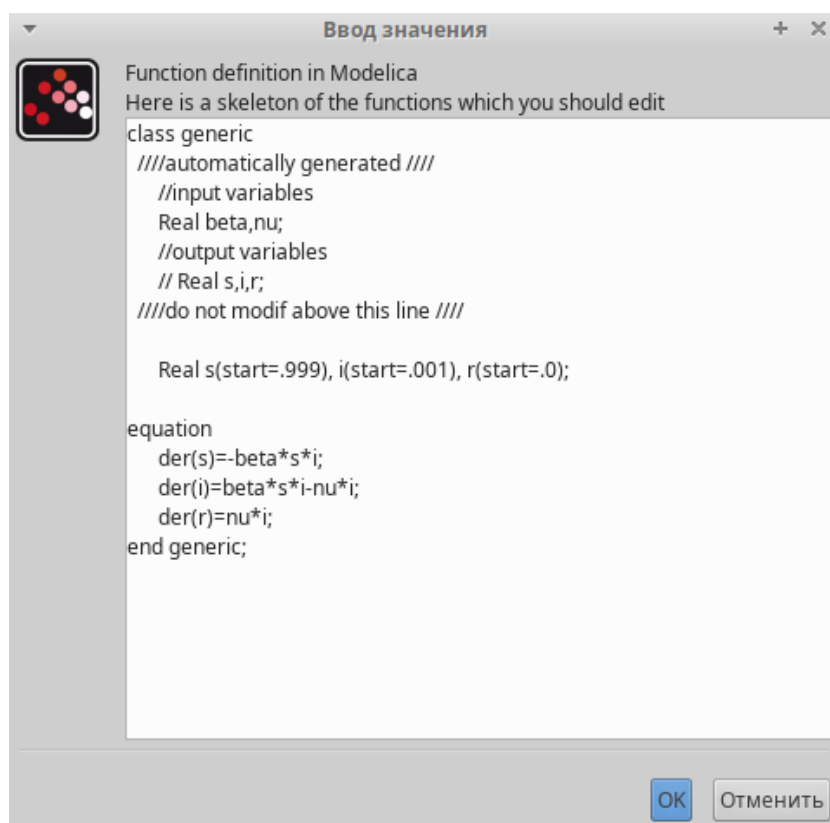


Рис. 4.11: Настройка параметров блока Modelica

Соединив блоки, получим следующую модель (рис. 4.12).

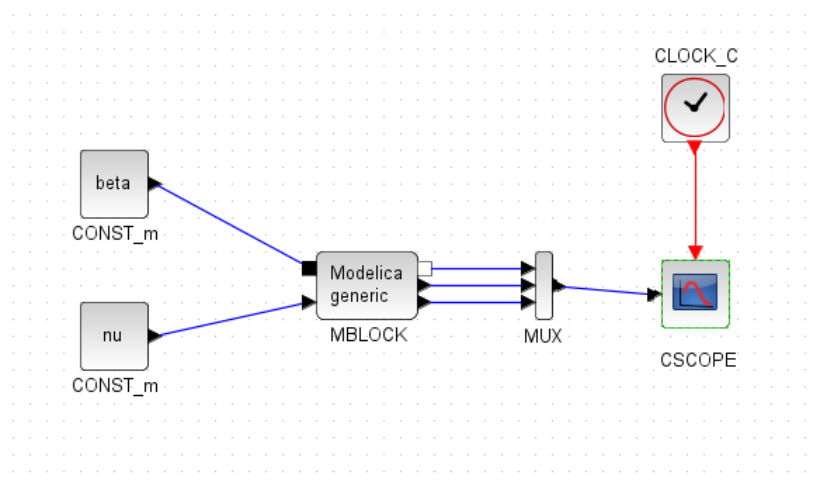


Рис. 4.12: Модель SIR в xcos с применением блока Modelica

В результате получим аналогичное предыдущему решение (рис. 4.13).

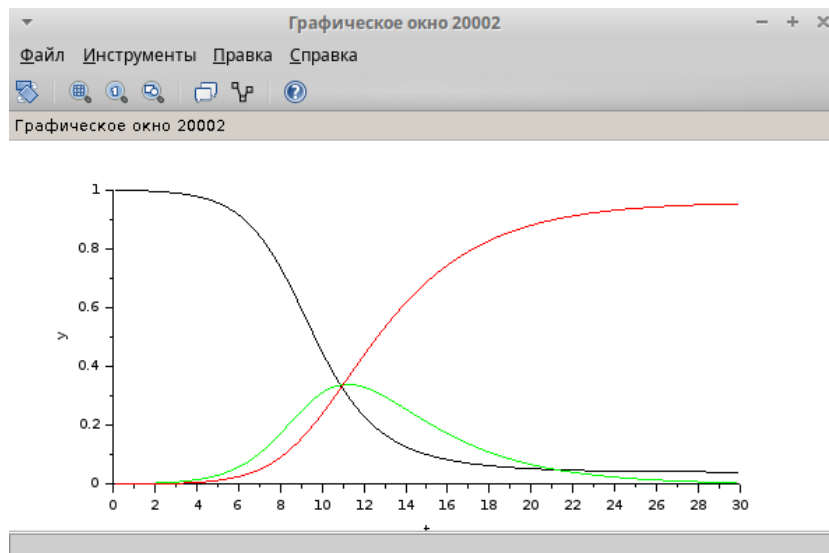


Рис. 4.13: График решения модели SIR при  $\beta = 1$ ,  $\nu = 0.3$

## 4.3 Реализация модели в OpenModelica

Реализуем модель в OpenModelica. Для этого создадим файл модели, пропишем там параметры и начальные условие, а также дифференциальное уравнение (рис. 4.14).

```

1  model lab5_1
2
3      parameter Real beta=1;
4      parameter Real nu=0.3;
5      parameter Real S_0=0.999;
6      parameter Real I_0=0.001;
7      parameter Real R_0=0;
8
9      Real s(start=S_0);
10     Real i(start=I_0);
11     Real r(start=R_0);
12
13     equation
14         der(s) = -beta*s*i;
15         der(i) = beta*s*i - nu*i;
16         der(r) = nu*i;
17
18     end lab5_1;

```

Рис. 4.14: Модель в OpenModelica

Затем укажем параметры моделирования, время так же поставим равным 30 единиц модельного времени



В результате получим график аналогичный графикам в xcos (рис. 4.15).

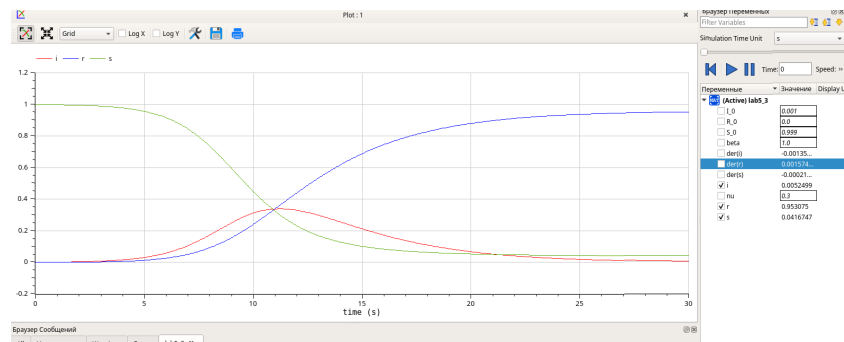


Рис. 4.15: График решения модели SIR при  $\beta = 1$ ,  $\nu = 0.3$

## 5 Задание для самостоятельного выполнения

### 5.1 Модель SIR с учетом демографии

В дополнение к предположениям, которые были сделаны для модели SIR, предположим, что учитываются демографические процессы, в частности, что смертность в популяции полностью уравнивает рождаемость, а все рожденные индивидуумы появляются на свет абсолютно здоровыми. Тогда получим следующую систему уравнений:

$$\begin{cases} \frac{dS}{dt} = -\beta IS + \mu(N - S), \\ \frac{dI}{dt} = \beta IS - \nu I - \mu I, \\ \frac{dR}{dt} = \nu I - \mu R, \end{cases}$$

где  $\mu$  – константа, которая равна коэффициенту смертности и рождаемости [1]. Требуется:

- реализовать модель SIR с учётом процесса рождения гибели особей в `xcos` (в том числе и с использованием блока `Modelica`), а также в `OpenModelica`;
- построить графики эпидемического порога при различных значениях параметров модели (в частности изменяя параметр  $\mu$ );
- сделать анализ полученных графиков в зависимости от выбранных значений параметров модели

## 5.2 Реализация модели в xcos

Для реализации этой модели добавим в переменные окружения  $\mu$  (рис. 5.1).

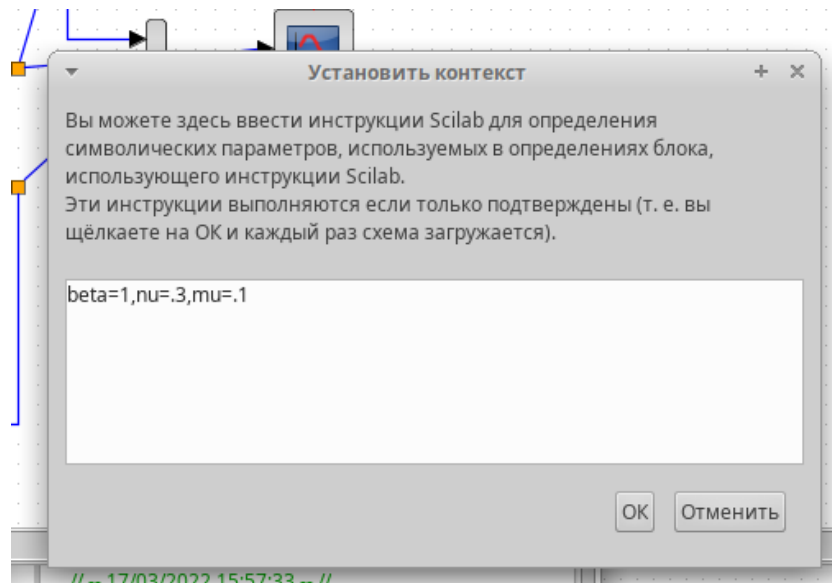


Рис. 5.1: Задать переменные окружения в xcos

Необходимые блоки такие же (рис. 5.2).

Первое уравнение модели задано верхним блоком интегрирования, блоком произведения, блоком задания коэффициента  $-\beta$  и сумматором. Блок произведения соединён с выходами верхнего и среднего блоков интегрирования и блоком коэффициента  $-\beta$ , что реализует математическую конструкцию  $-\beta s(t)i(t)$ , которая передается в блок суммирования. Ниже заданы математические конструкции  $\mu i$  и  $\mu r$ , которые со знаком плюс передаются в сумматор перед первым блоком интегрирования.

Третье уравнение модели задано нижним блоком интегрирования и блоком задания коэффициента  $\nu$ . Для реализации математической конструкции  $\nu i(t)$  соединяем выход среднего блока интегрирования и вход блока задания коэффициента  $\nu$ . Перед блоком интегрирования размещаем сумматор, в которой передаем математические конструкции  $\mu r$  со знаком минус и  $\nu i(t)$ . Результат суммирования передаём на вход нижнего блока интегрирования.

Средний блок интегрирования и блок суммирования определяют второе уравнение модели, которое по сути является суммой правых частей первого и третьего уравнений со знаком минус. Для реализации соединяем входы верхнего и нижнего блоков интегрирования с входами блока суммирования, меняя при этом в его параметрах оба знака на минус. Выход блока суммирования соединяем с входом среднего блока интегрирования

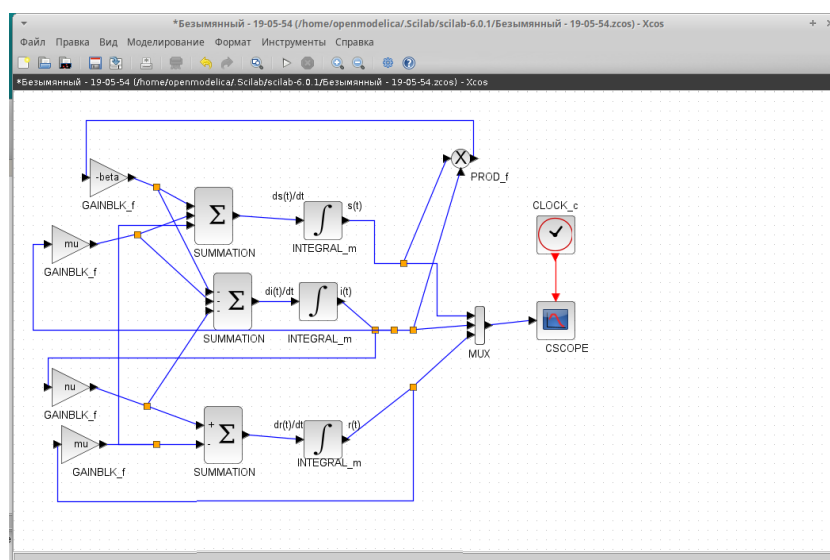


Рис. 5.2: Модель SIR с учетом демографии в xcos

В результате получим график решения (рис. 5.3).

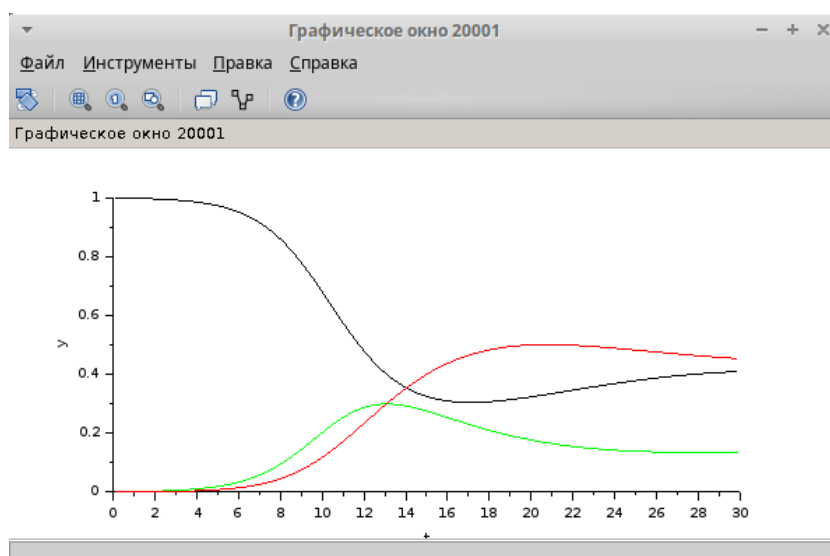


Рис. 5.3: График решения модели SIR с учетом демографии при  $\beta = 1$ ,  $\nu = 0.3$ ,  $\mu = 0.1$

Здесь так же происходит стабилизация всех функций после прохождения порога эпидемии, но, в отличие от предыдущего решения, количество заболевших стабилизируется не на уровне 0, а на уровне определенного значения, то есть из-за постоянного появления новых особей, появляются новые уязвимые, которые могут заболеть и заболевают.

## 5.3 Реализация модели с помощью блока Modelica в

### xcos

Для реализации с помощью блока Modelica добавим блок параметра  $\mu$ .

Также изменим данные блока Modelica, добавив информацию о третьем параметре и изменив дифференциальное уравнение (рис. 5.4, 5.5).

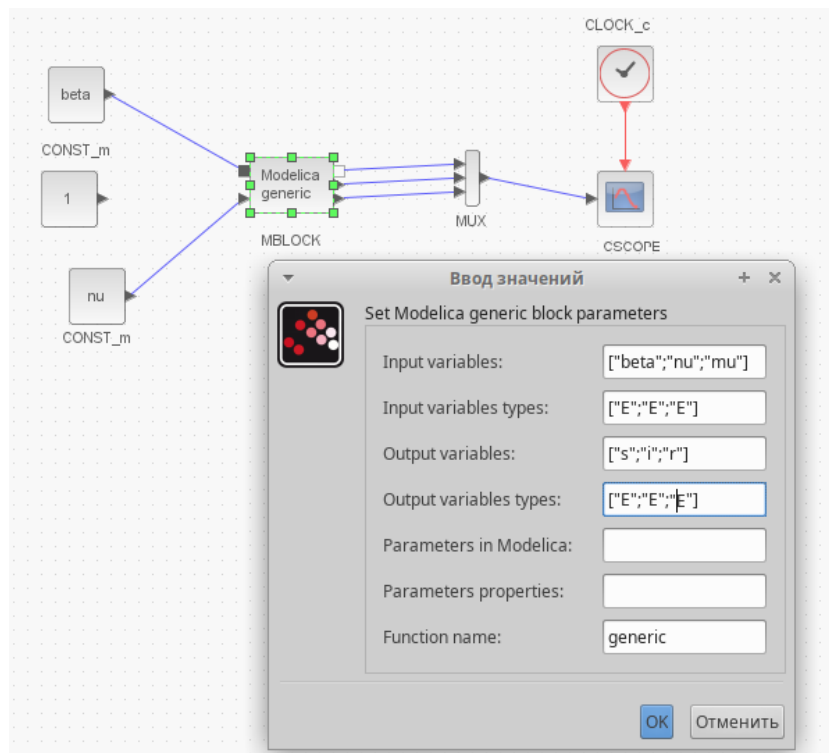


Рис. 5.4: Настройка параметров блока Modelica

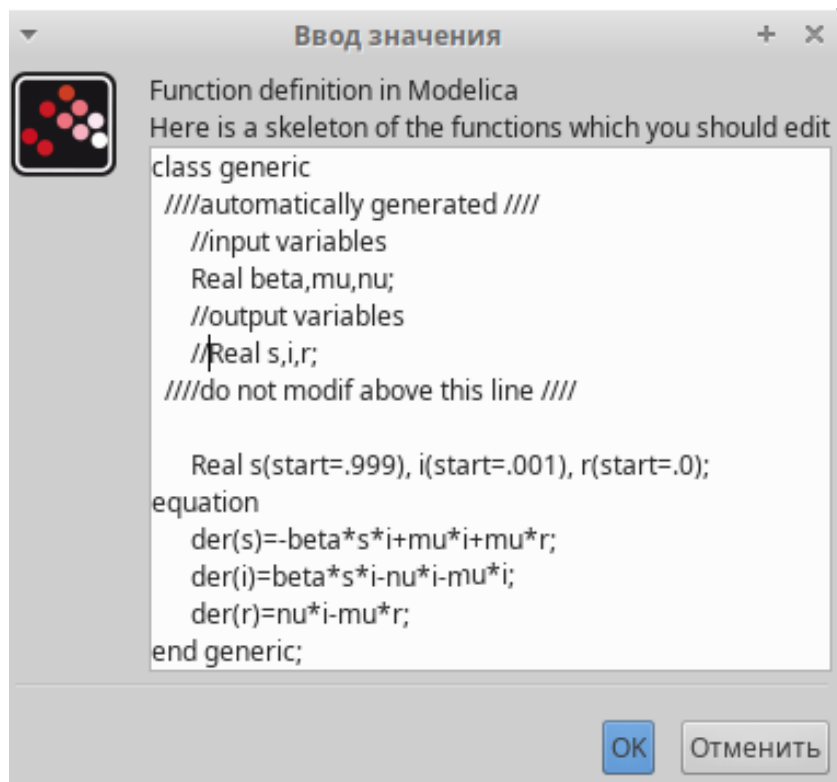


Рис. 5.5: Настройка параметров блока Modelica

Соединим блоки и получим следующую модель (рис. 5.6).

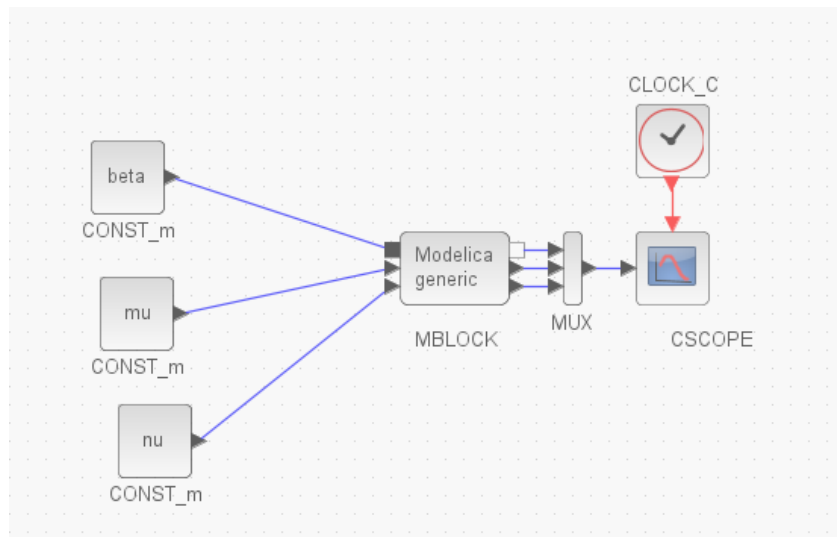


Рис. 5.6: Модель SIR в xcos с применением блока Modelica

В результате получим график решения(рис. 5.7).

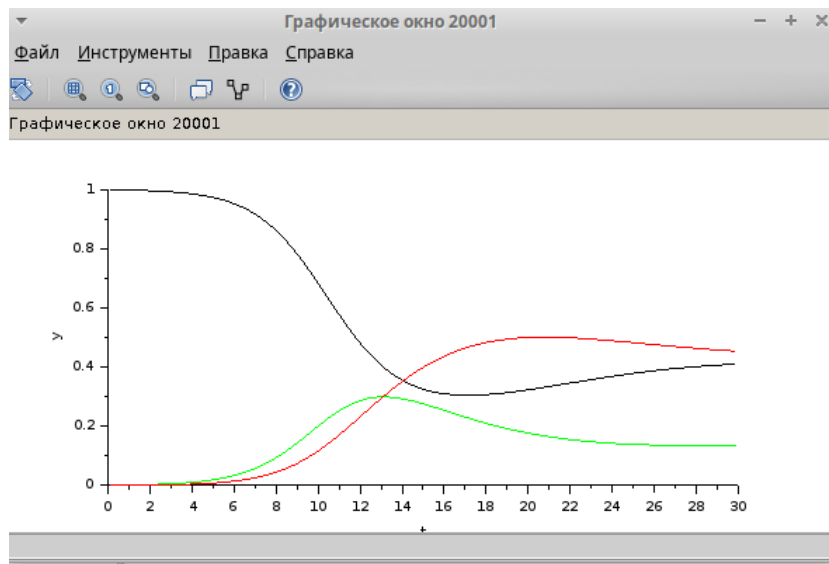


Рис. 5.7: График решения модели SIR с учетом демографии при  $\beta = 1$ ,  $\nu = 0.3$ ,  $\mu = 0.1$

## 5.4 Реализация модели в OpenModelica

Изменим данные программы в OpenModelica, добавив информацию о третьем параметре и изменив дифференциальное уравнение (рис. 5.8).

```

1  model task5
2    parameter Real beta=1;
3    parameter Real nu=0.3;
4    parameter Real mu=0.1;
5    parameter Real S_0=0.999;
6    parameter Real I_0=0.001;
7    parameter Real R_0=0;
8
9    Real s(start=S_0);
10   Real i(start=I_0);
11   Real r(start=R_0);
12
13   equation
14     der(s) = -beta*s*i + mu*(r+i);
15     der(i) = beta*s*i - nu*i - mu*i;
16     der(r) = nu*i - mu*r;
17
18   end task5;

```

Рис. 5.8: Модель SIR с учетом демографии в OpenModelica

Затем укажем параметры моделирования, время так же поставим равным 30



единиц модельного времени (рис. 5.9).

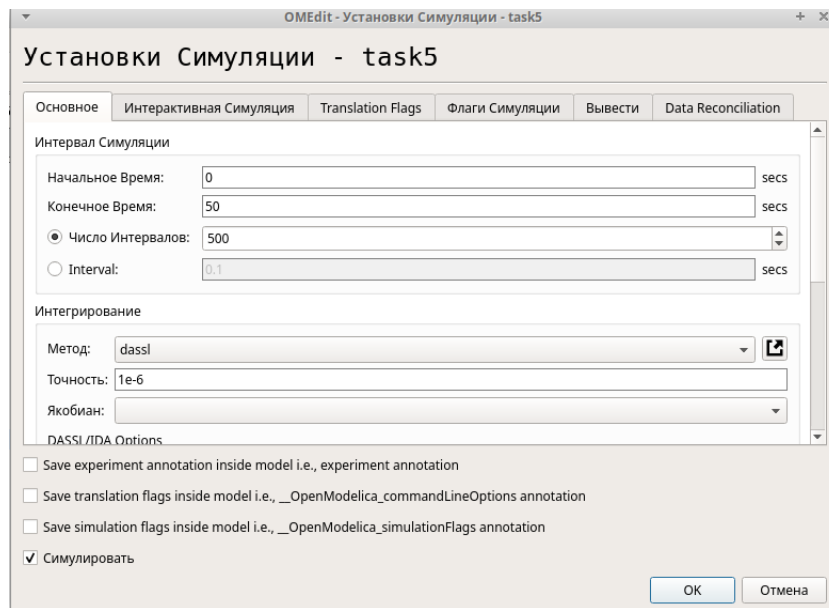


Рис. 5.9: Параметры моделирования в OpenModelica

В результате получим график аналогичный графикам в xcos (рис. 5.10).

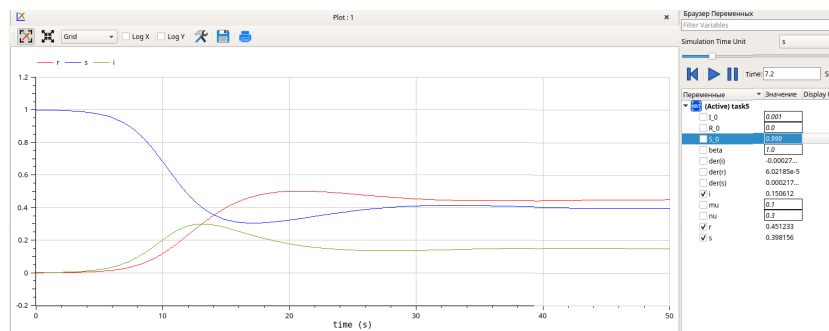


Рис. 5.10: График решения модели SIR с учетом демографии при  $\beta = 1$ ,  $\nu = 0.3$ ,  $\mu = 0.1$

## 5.5 Анализ графиков при разных параметрах модели

Можно увидеть, что чем больше значение любого параметра, тем быстрее система приходит в стационарное состояние (рис. 5.11 - 5.15).

Когда параметр  $\mu$  достигает значения 0.5 (рис. 5.12) на графике очень слабо

меняются траектории переменных. Это можно объяснить тем, что рождается и умирает почти столько же здоровых, сколько заражается. Можно сделать вывод, что чем выше показатель  $\mu$ , тем слабее эпидемия влияет на популяцию (тем слабее меняются траектории графиков)

Если  $\mu = \nu$  (рис. 5.11), то графики изменения выздоровевших/умерших и заболевших после стабилизации системы совпадут, значит рождается и заболевает столько же, сколько выздоравливает и умирает.

Чем меньше  $\mu$ , тем меньше этот параметр влияет на систему и ее решение все больше походит на систему, где этот параметр не учитывается (рис. 5.13).

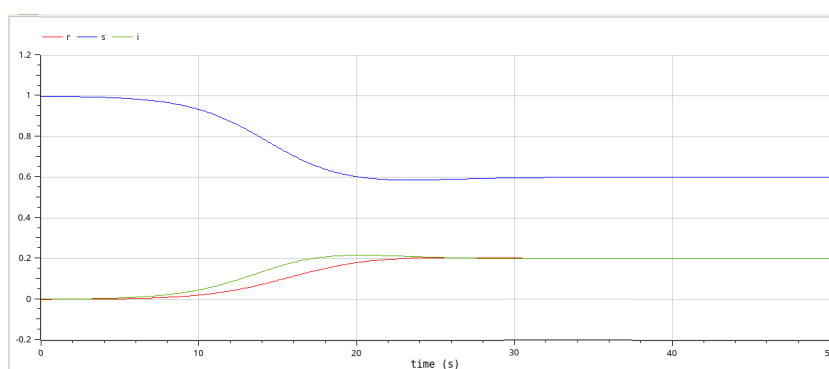


Рис. 5.11: График решения модели SIR с учетом демографии при  $\beta = 1$ ,  $\nu = 0.3$ ,  $\mu = 0.3$ . OpenModelica

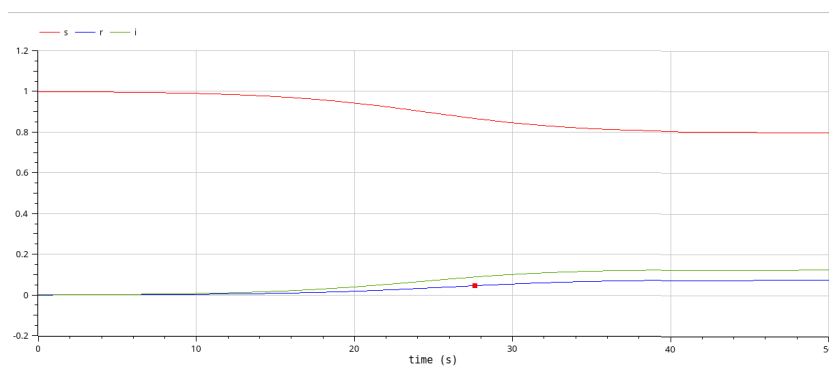


Рис. 5.12: График решения модели SIR с учетом демографии при  $\beta = 1$ ,  $\nu = 0.3$ ,  $\mu = 0.5$ . OpenModelica

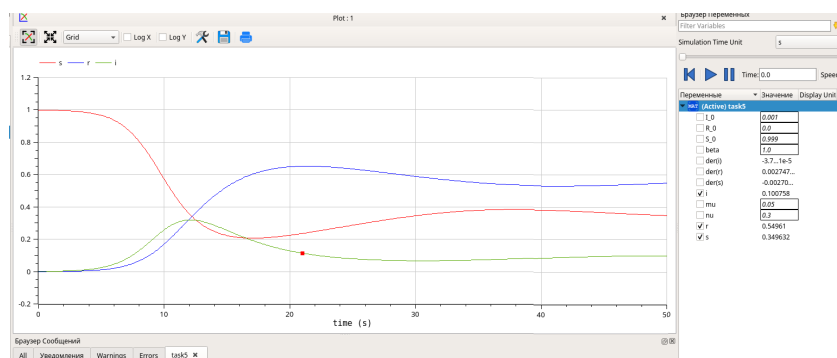


Рис. 5.13: График решения модели SIR с учетом демографии при  $\beta = 1$ ,  $\nu = 0.3$ ,  $\mu = 0.05$ . OpenModelica

При увеличении параметра  $\beta$  на графике заметим, что количество уязвимых людей снижается до минимума быстрее, ведь мы увеличили скорость заражения (рис. 5.14).

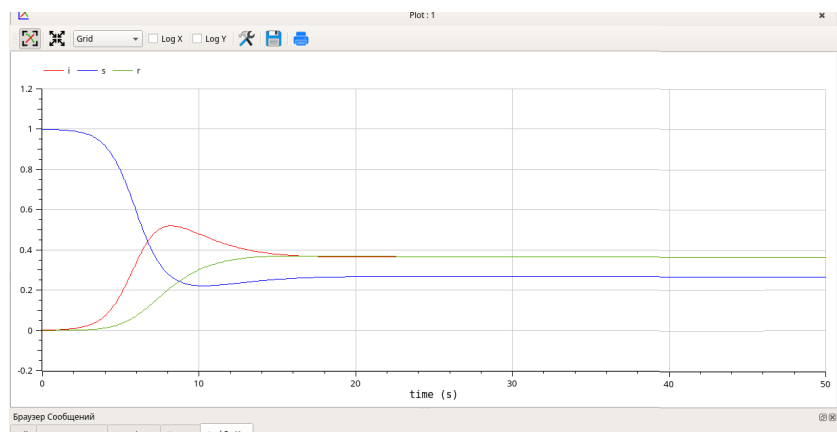


Рис. 5.14: График решения модели SIR с учетом демографии при  $\beta = 1.5$ ,  $\nu = 0.2$ ,  $\mu = 0.2$ . OpenModelica

При увеличении параметра  $\nu$ , график тех, кто приобрел иммунитет находится выше графика заболевших, и быстрее возрастает, ведь мы увеличили скорость выздоровления (рис. 5.15).

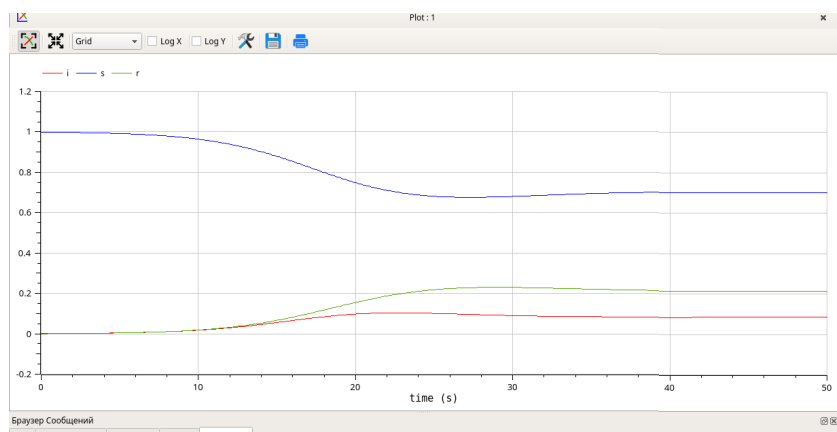


Рис. 5.15: График решения модели SIR с учетом демографии при  $\beta = 1$ ,  $\nu = 0.5$ ,  $\mu = 0.2$ . OpenModelica

## 6 Выводы

В результате выполнения работы была исследована модель SIR при помощи xcoss и OpenModelica.

## Список литературы

1. Королькова А. В. К.Д.С. Лабораторная работа 5. Модель эпидемии (SIR) [Электронный ресурс].
2. Королькова А. В. К.Д.С. Компонентное моделирование. Scilab, подсистема xcos [Электронный ресурс].