

Отчёт по лабораторной работе №5

Дисциплина: Имитационное моделирование

Ганина Таисия Сергеевна, НФИбд-01-22

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	9
4.1	Повторим пример из лабораторной работы	9
4.1.1	Реализация модели в xcos	9
4.1.2	Реализация модели с помощью блока Modelica в xcos . . .	15
4.2	Упражнение. Реализация модели SIR в OpenModelica	19
4.3	Задание для самостоятельной работы	20
4.3.1	Реализация модели в xcos	21
4.3.2	Реализация модели с помощью блока Modelica в xcos (2 варианта)	24
4.3.3	Реализация модели в OpenModelica	29
4.3.4	Графики эпидемического порога при различных значениях параметров модели	31
5	Выводы	36
	Список литературы	37

Список иллюстраций

4.1	Установить контекст для учебного примера в Xcos	9
4.2	Модель SIR в xcos	10
4.3	Задать начальные значения в верхнем блоке интегрирования . .	11
4.4	Задать начальные значения в среднем блоке интегрирования . .	11
4.5	Задать начальные значения в нижнем блоке интегрирования . . .	12
4.6	Задать конечное время интегрирования в xcos = 30	12
4.7	Параметры MUX	13
4.8	Редактирование параметров блока Scope	13
4.9	Задание параметров суммы	14
4.10	Задаю параметр $\beta = -1$	14
4.11	Задаю параметр $\nu = 0.3$	15
4.12	Эпидемический порог модели SIR, график построен в Xcos	15
4.13	Параметры блока Modelica для модели. Ввод значений	16
4.14	Параметры блока Modelica для модели. Ввод значений - функция .	17
4.15	CONST_m β	17
4.16	CONST_m ν	18
4.17	Модель с помощью блока Modelica в xcos	18
4.18	Эпидемический порог модели SIR, график построен с помощью блока Modelica в xcos	19
4.19	Код для модели SIR в OpenModelica	19
4.20	Установки симуляции (конечное время = 40)	20
4.21	Эпидемический порог модели SIR, график построен в OpenModelica	20
4.22	Задание параметров суммы (одной из сумм)	21
4.23	Задать конечное время интегрирования в xcos = 30	22
4.24	Установить контекст для самостоятельного примера в Xcos	22
4.25	Параметры MUX	23
4.26	Модель SIR (с параметром μ) в xcos	23
4.27	Эпидемический порог модели SIR (с параметром μ), график по- строен в Xcos	24
4.28	Модель с помощью блока Modelica (с параметром μ) в xcos	25
4.29	Параметры блока Modelica для модели (с параметром μ). Ввод зна- чений	25
4.30	Параметры блока Modelica для модели (с параметром μ). Ввод зна- чений - функция	26
4.31	Установить контекст в Xcos (с параметром μ)	26
4.32	Эпидемический порог модели SIR (с параметром μ), график по- строен с помощью блока Modelica в xcos	27

4.33	Модель с помощью блока Modelica (с параметром μ и N) в xcos . .	27
4.34	Установить контекст в Xcos (с параметром μ и N)	27
4.35	Параметры блока Modelica для модели (с параметром μ и N). Ввод значений	28
4.36	Параметры блока Modelica для модели (с параметром μ и N). Ввод значений - функция	28
4.37	Эпидемический порог модели SIR (с параметром μ и N), график построен с помощью блока Modelica в xcos	28
4.38	Код для модели SIR в OpenModelica (с параметром μ)	29
4.39	Установки симуляции (конечное время = 30)	29
4.40	Эпидемический порог модели SIR (с параметром μ), график построен в OpenModelica	30
4.41	Код для модели SIR в OpenModelica (с параметром μ и N)	30
4.42	Эпидемический порог модели SIR (с параметром μ и N), график построен в OpenModelica	31
4.43	$\beta = 1, \nu = 0.3, \mu = 0.1$	31
4.44	$\beta = 1, \nu = 0.3, \mu = 0.5$	32
4.45	$\beta = 1, \nu = 0.3, \mu = 1$	32
4.46	$\beta = 1, \nu = 0.3, \mu = 0.3$	32
4.47	$\beta = 3, \nu = 0.6, \mu = 0.1$	33
4.48	$\beta = 16, \nu = 0.9, \mu = 0.8$	33

Список таблиц

1 Цель работы

Выполнить задания и получить практические навыки работы со средствами моделирования xcos, Modelica и OpenModelica. Рассмотреть модель эпидемии (SIR).

2 Задание

1. Реализовать имитационную модель эпидемии в xcos;
2. Реализовать имитационную модель эпидемии в Modelica;
3. Реализовать имитационную модель эпидемии в OpenModelica (упражнение);
4. Выполнить задание для самостоятельной работы.

3 Теоретическое введение

Модель SIR предложена в 1927 г. (W. O. Kermack, A. G. McKendrick). Предполагается, что особи популяции размера N могут находиться в трёх различных состояниях:

- S (susceptible, уязвимые) — здоровые особи, которые находятся в группе риска и могут подхватить инфекцию;
- I (infective, заражённые, распространяющие заболевание) — заразившиеся переносчики болезни;
- R (recovered/removed, вылечившиеся) — те, кто выздоровел и перестал распространять болезнь (в эту категорию относят, например, приобретших иммунитет или умерших).

Внутри каждой из выделенных групп особи считаются неразличимыми по свойствам. Типичная эволюция особи популяции описывается следующей диаграммой:

$S \rightarrow I \rightarrow R$

Считаем, что система замкнута, т.е. $N=S+I+R$. [1].

4 Выполнение лабораторной работы

4.1 Повторим пример из лабораторной работы

4.1.1 Реализация модели в xcos

Зайдя в среду моделирования Xcos начала выполнять учебный пример. В начале во вкладке “Моделирование” открыла “Установить контекст” и задала переменные $\beta = 1, \nu = 0.3$ (рис. 4.1).

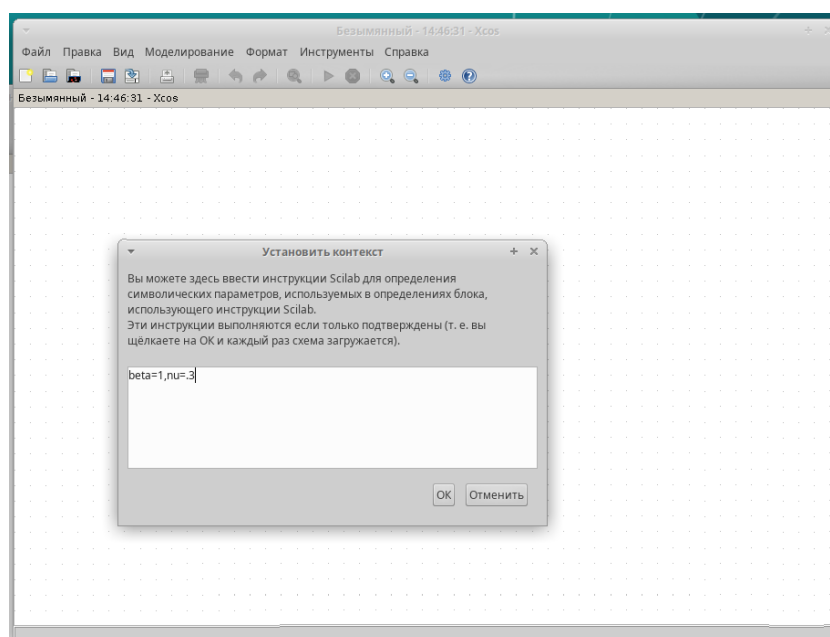


Рис. 4.1: Установить контекст для учебного примера в Xcos

Далее я реализовала модель при помощи следующих блоков xcos(рис. 4.2):

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

- 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 β и ν ;
- SUMMATION — блок суммирования;
- PROD_f — поэлементное произведение двух векторов на входе блока.

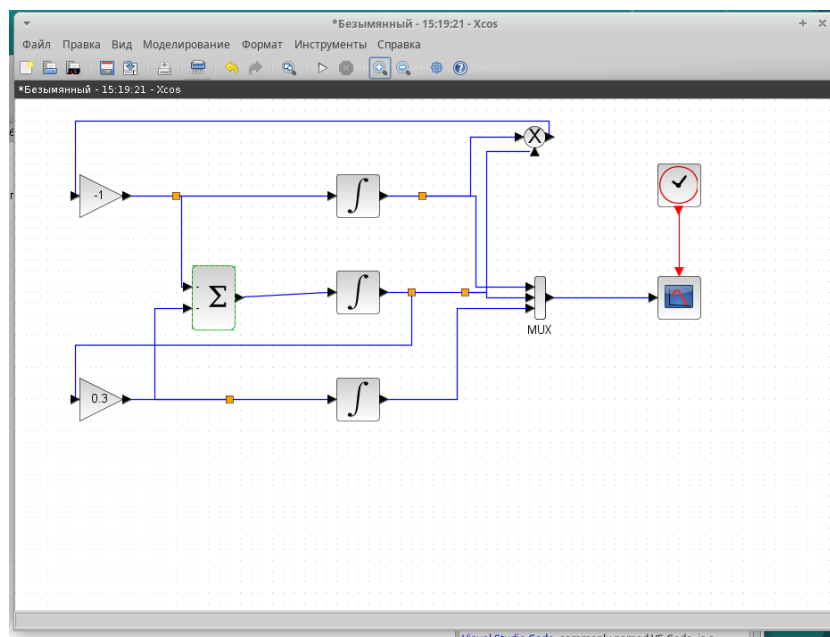
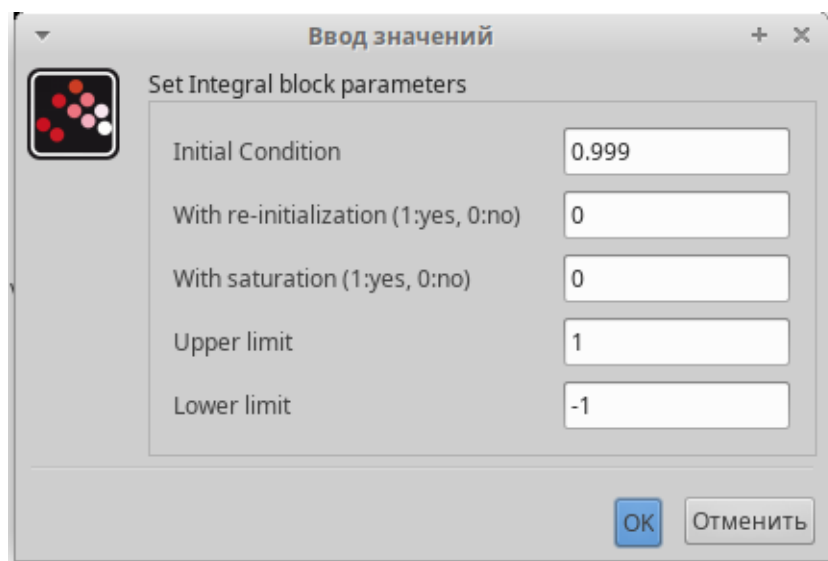


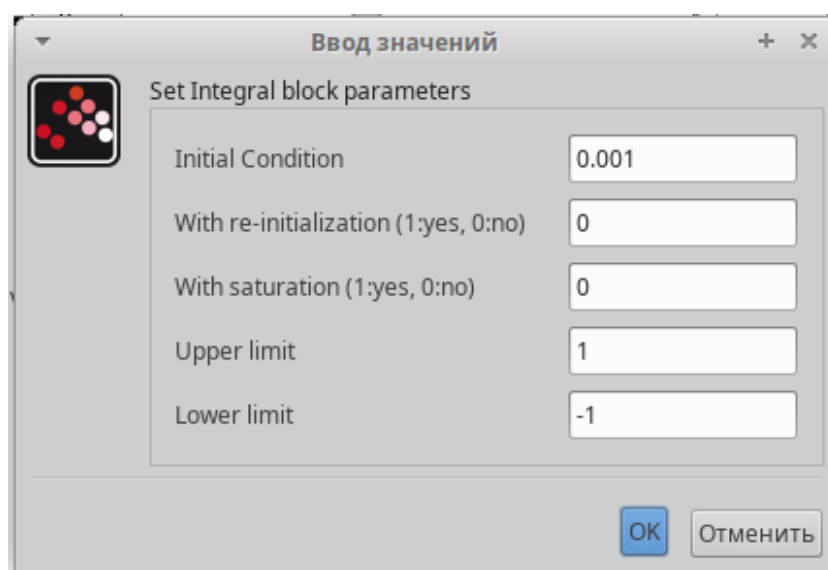
Рис. 4.2: Модель SIR в xcos

Более подробно про вводимые мной значения (рис. 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12)



Set Integral block parameters	
Initial Condition	0.999
With re-initialization (1:yes, 0:no)	0
With saturation (1:yes, 0:no)	0
Upper limit	1
Lower limit	-1

Рис. 4.3: Задать начальные значения в верхнем блоке интегрирования



Set Integral block parameters	
Initial Condition	0.001
With re-initialization (1:yes, 0:no)	0
With saturation (1:yes, 0:no)	0
Upper limit	1
Lower limit	-1

Рис. 4.4: Задать начальные значения в среднем блоке интегрирования

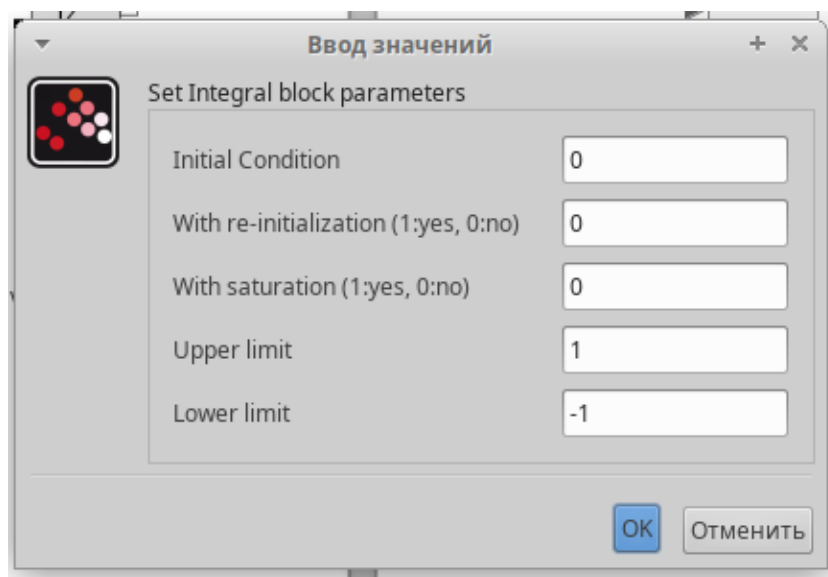


Рис. 4.5: Задать начальные значения в нижнем блоке интегрирования

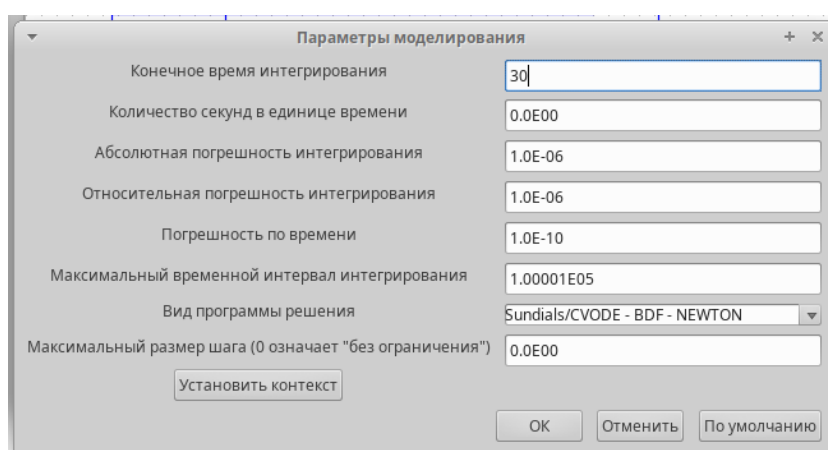


Рис. 4.6: Задать конечное время интегрирования в $x_{cos} = 30$

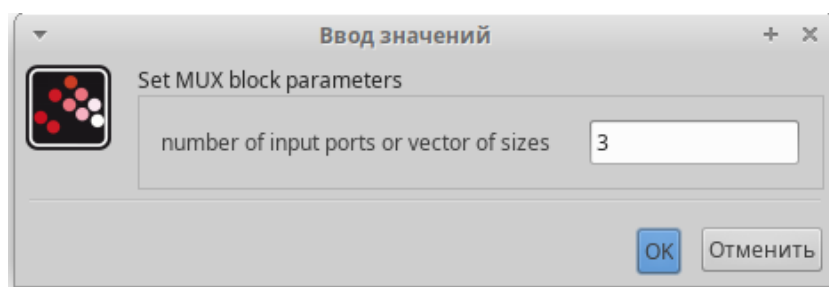


Рис. 4.7: Параметры MUX

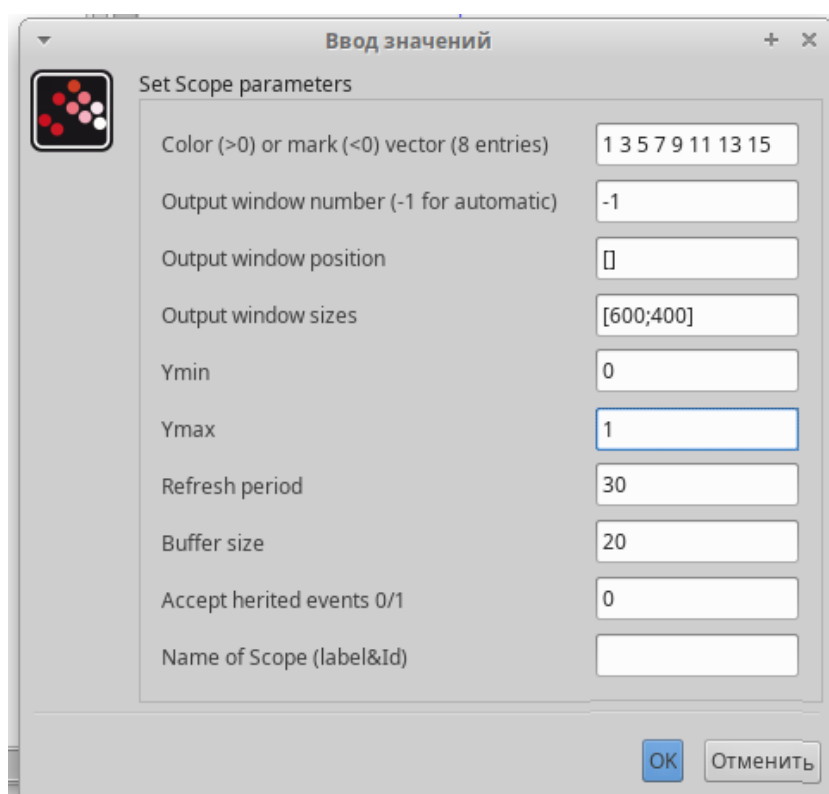


Рис. 4.8: Редактирование параметров блока Scope

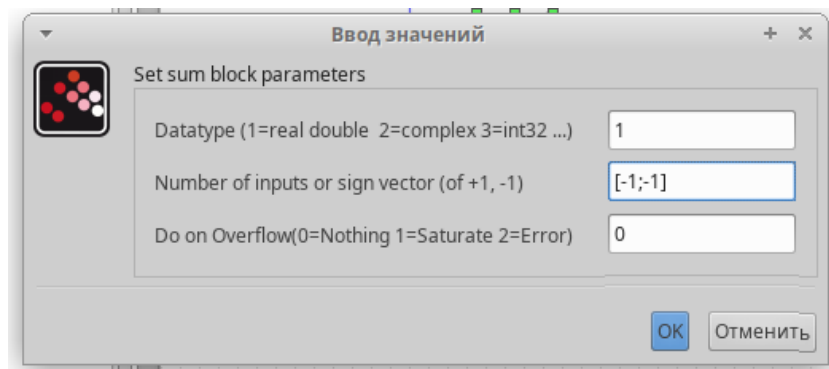


Рис. 4.9: Задание параметров суммы

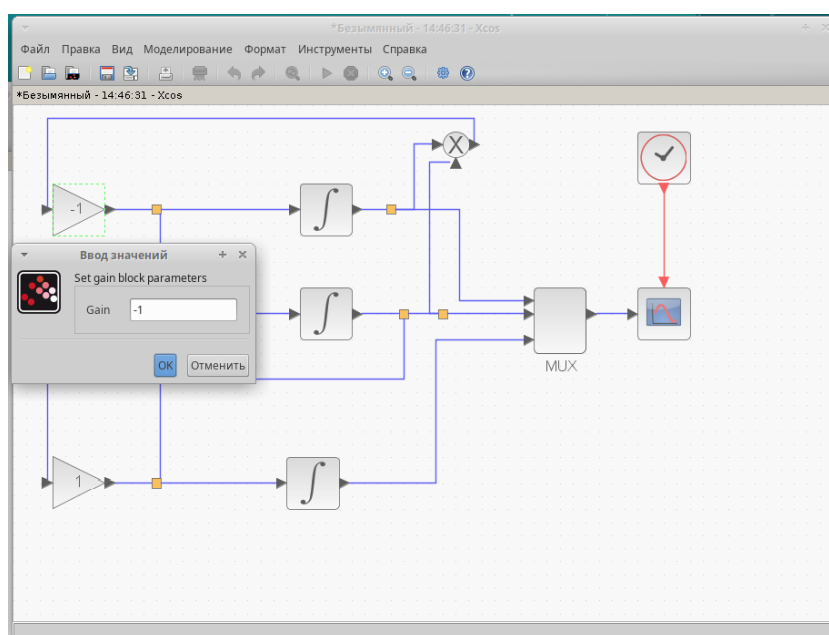


Рис. 4.10: Задаю параметр $\beta = -1$

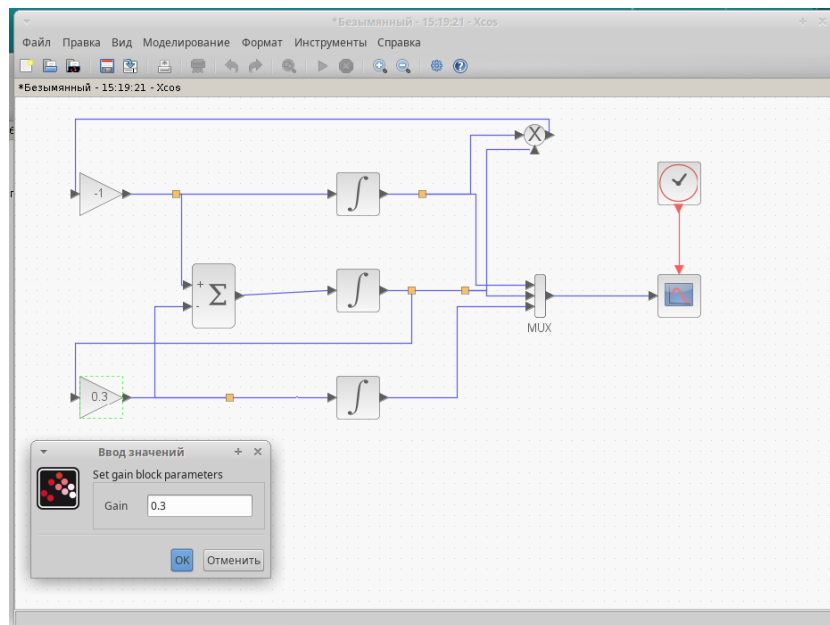


Рис. 4.11: Задаю параметр $\nu = 0.3$

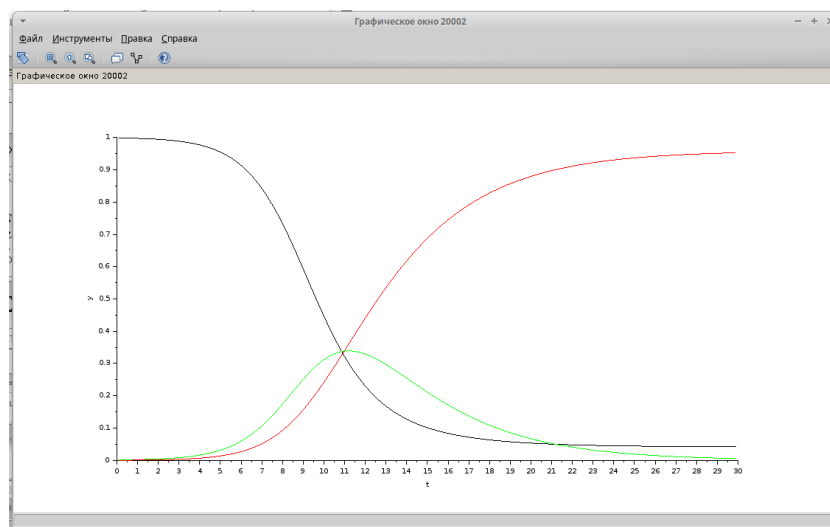


Рис. 4.12: Эпидемический порог модели SIR, график построен в Xcos

4.1.2 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:

- CONST_m — задаёт константу;
- MBLOCK (Modelica generic) — блок реализации кода на языке Modelica.

В начале задаем необходимые параметры такие же, как и в прошлый раз (рис. 4.1, 4.6).

После этого я перешла к новым параметрам (рис. 4.13, 4.14):

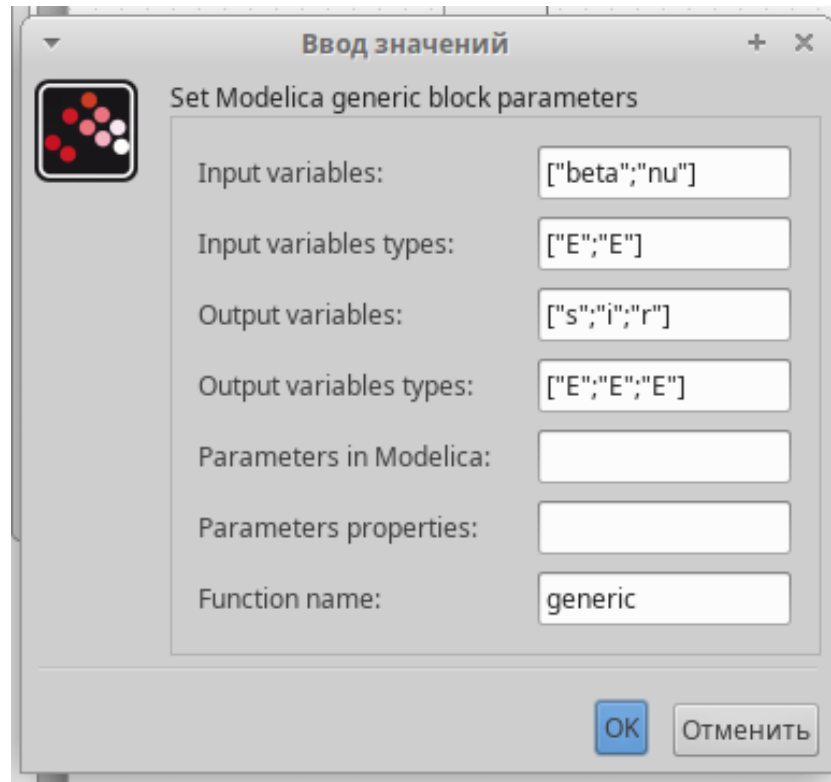


Рис. 4.13: Параметры блока Modelica для модели. Ввод значений

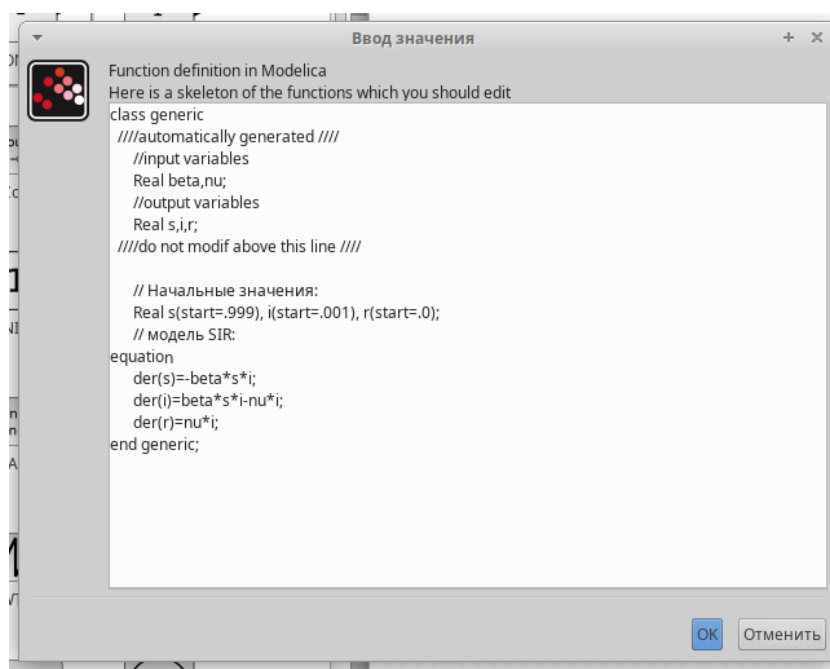


Рис. 4.14: Параметры блока Modelica для модели. Ввод значений - функция

Пишу значения в блоки CONST_m (рис. 4.15, 4.16):

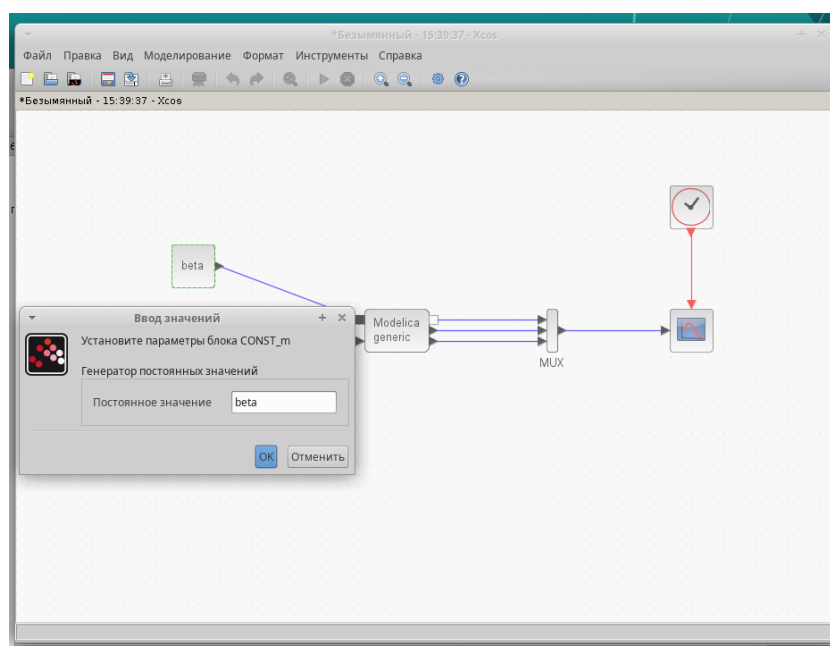


Рис. 4.15: CONST_m β

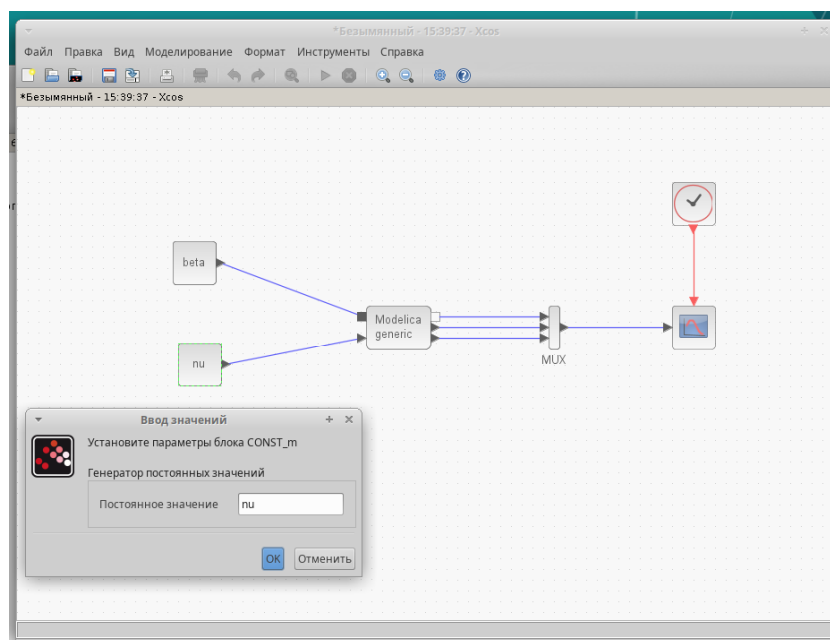


Рис. 4.16: CONST_m ν

Итоговая модель выглядит так (рис. 4.17, 4.18):

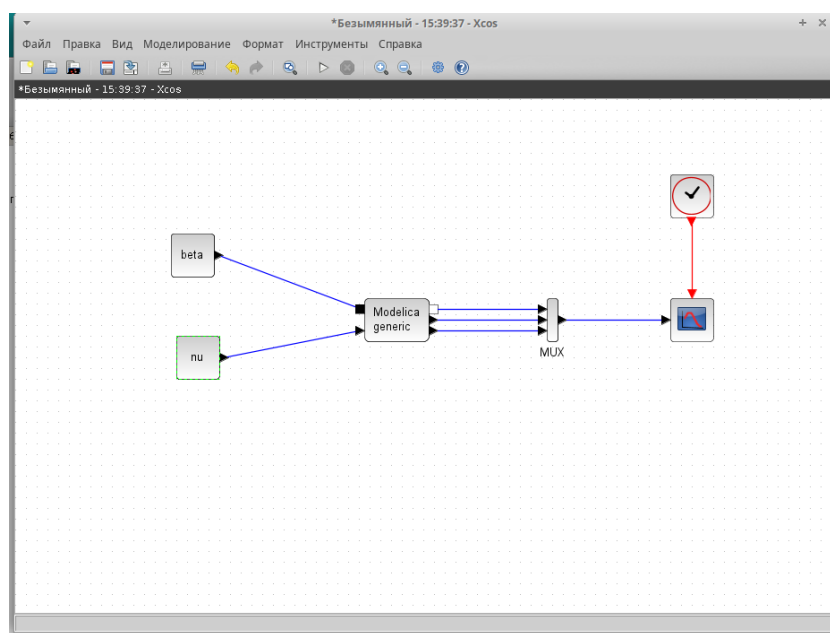


Рис. 4.17: Модель с помощью блока Modelica в xcos

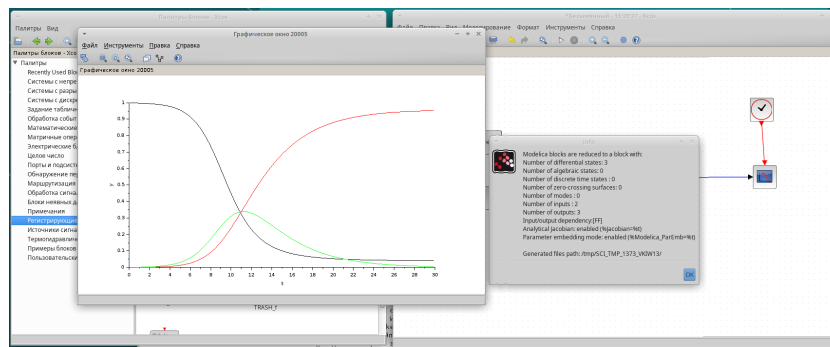


Рис. 4.18: Эпидемический порог модели SIR, график построен с помощью блока Modelica в xcos

4.2 Упражнение. Реализация модели SIR в OpenModelica

Для начала я создала класс, а после написала код (рис. 4.19, 4.20).

The screenshot shows the OpenModelica IDE with the 'sir_model' class defined in the 'sir_model' package. The code is as follows:

```

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

```

A confirmation dialog box titled 'OMEdit - Проверить Модель' is displayed, showing the following messages:

- 1 Check of sir_model completed successfully.
- 2 Class sir model has 3 equation(s) and 3 variable(s).
- 3 0 of these are trivial equation(s).

The 'OK' button is visible at the bottom right of the dialog.

Рис. 4.19: Код для модели SIR в OpenModelica

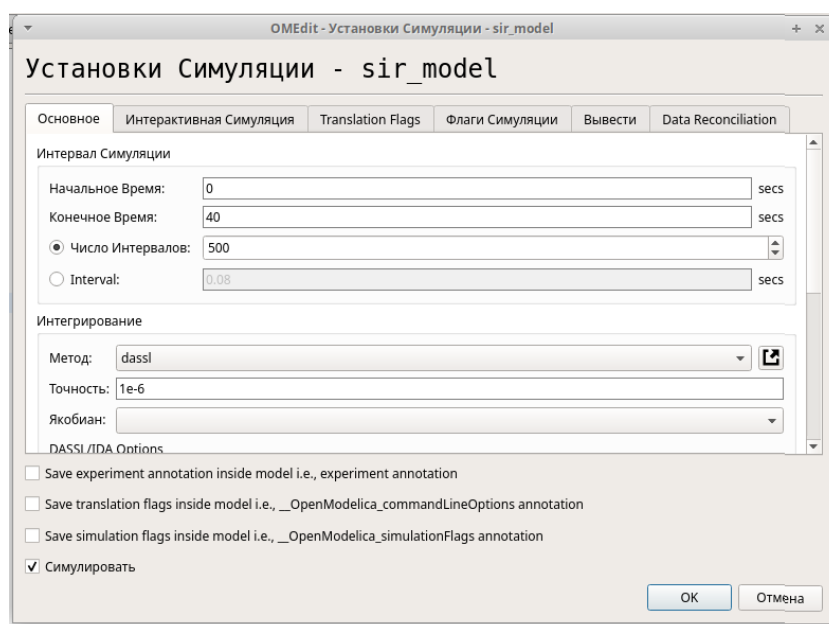


Рис. 4.20: Установки симуляции (конечное время = 40)

Результат моделирования выглядит вот так (абсолютно такой же, что и предыдущие – значит, код верный) (рис. 4.21)

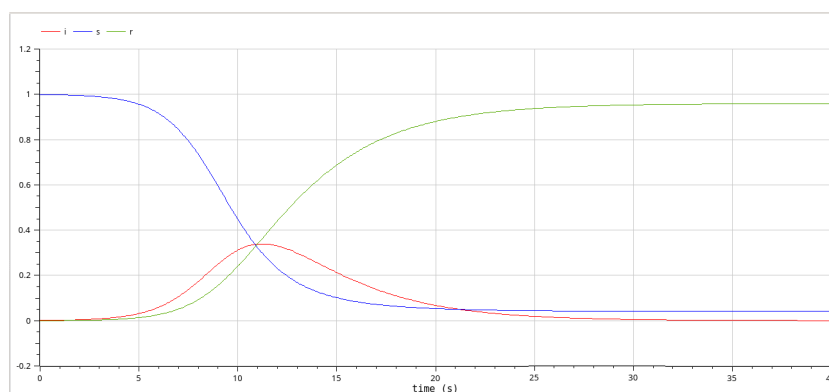


Рис. 4.21: Эпидемический порог модели SIR, график построен в OpenModelica

4.3 Задание для самостоятельной работы

Предположим, что в модели SIR 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:

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

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

4.3.1 Реализация модели в хcos

По примеру уже выполненного задания строим модель (рис. 4.22, 4.23, 4.24, 4.25, 4.26, 4.27).

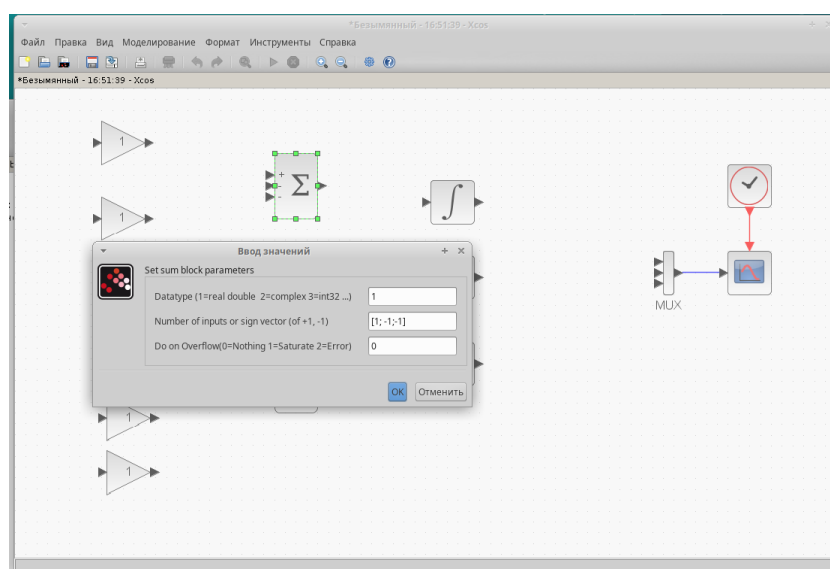


Рис. 4.22: Задание параметров суммы (одной из сумм)

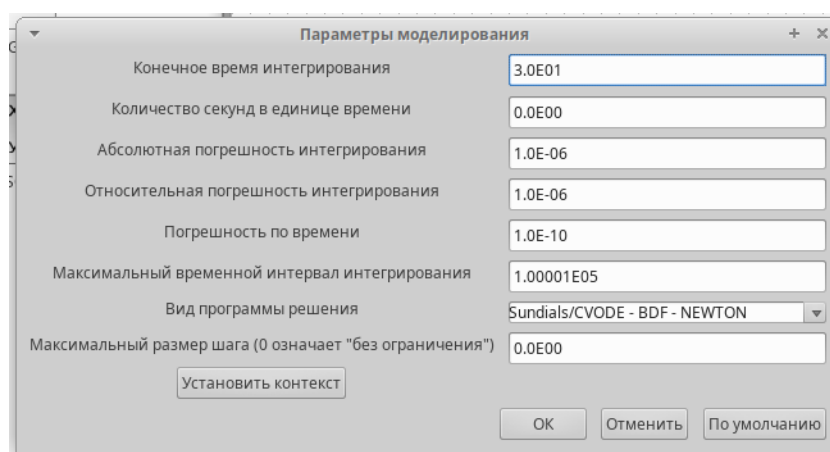


Рис. 4.23: Задать конечное время интегрирования в $x_{cos} = 30$

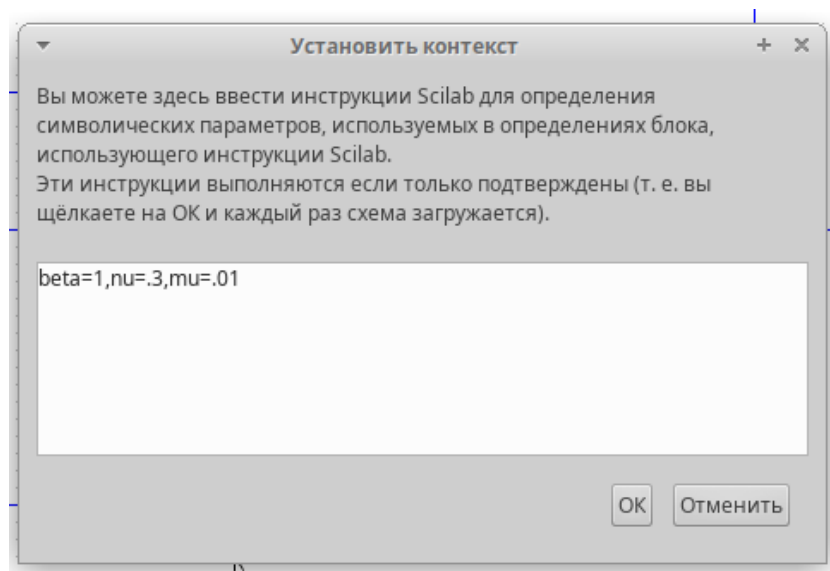


Рис. 4.24: Установить контекст для самостоятельного примера в Xcos

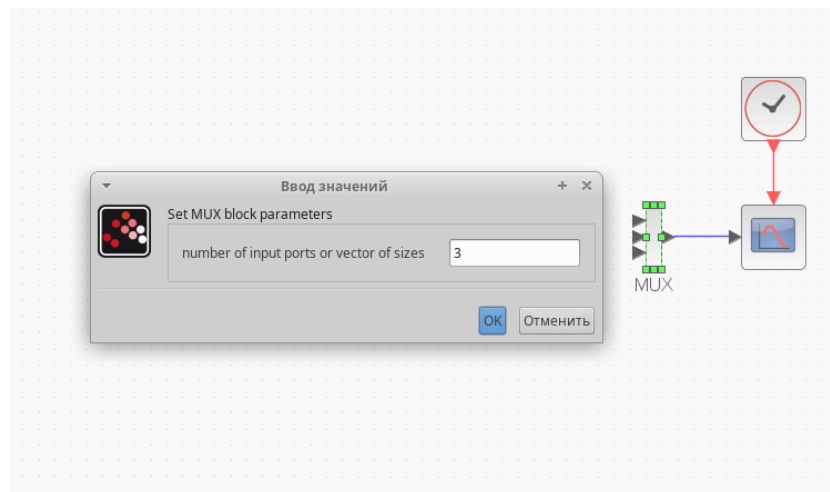


Рис. 4.25: Параметры MUX

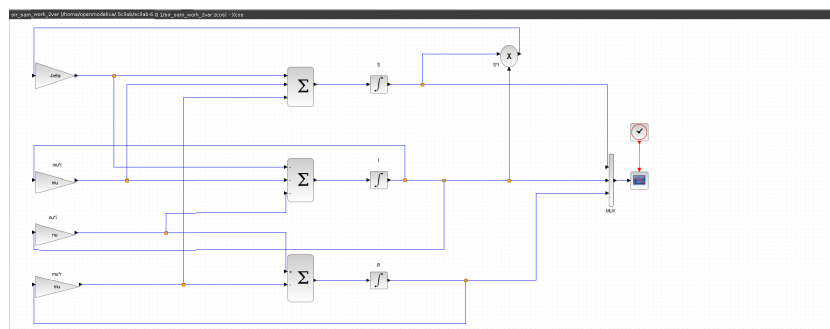


Рис. 4.26: Модель SIR (с параметром μ) в xcos

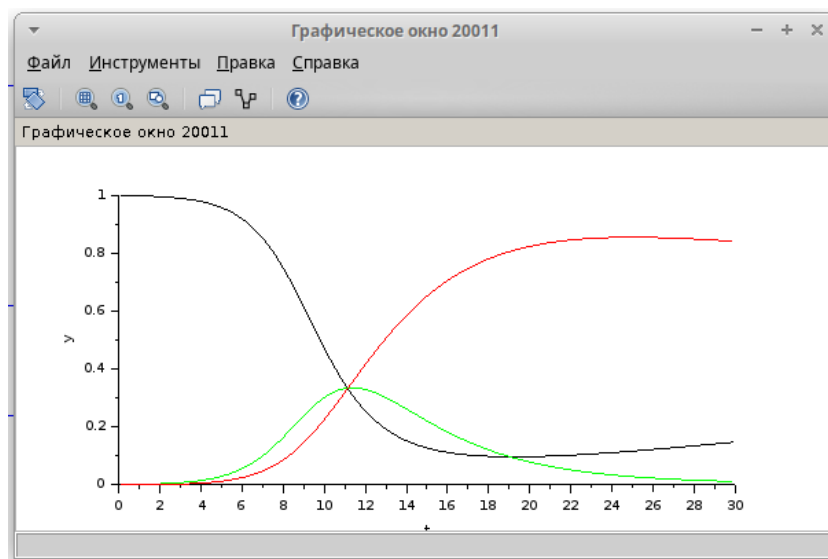


Рис. 4.27: Эпидемический порог модели SIR (с параметром μ), график построен в Xcos

4.3.2 Реализация модели с помощью блока Modelica в xcos (2 варианта)

В задании формула выглядит следующим образом:

$$-\beta s(t)i(t) + \mu(N - s(t))$$

Но в начале мы говорили, что $N = S + I + R$, значит, можно вывести следующее:

$$N - S = I + R$$

Используем это для построения модели (рис. 4.28, 4.29, 4.30, 4.31, 4.32):

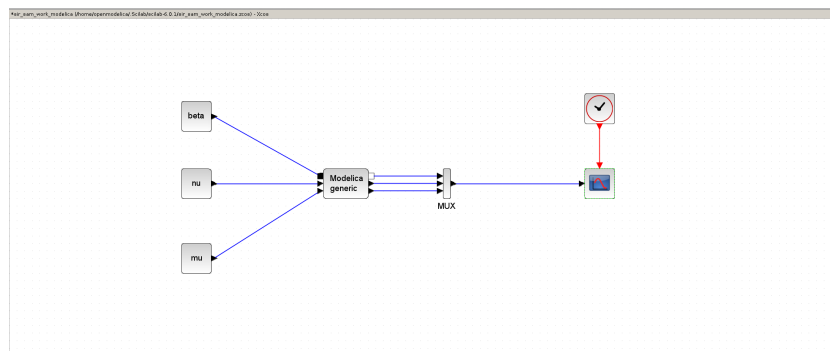


Рис. 4.28: Модель с помощью блока Modelica (с параметром μ) в xcos

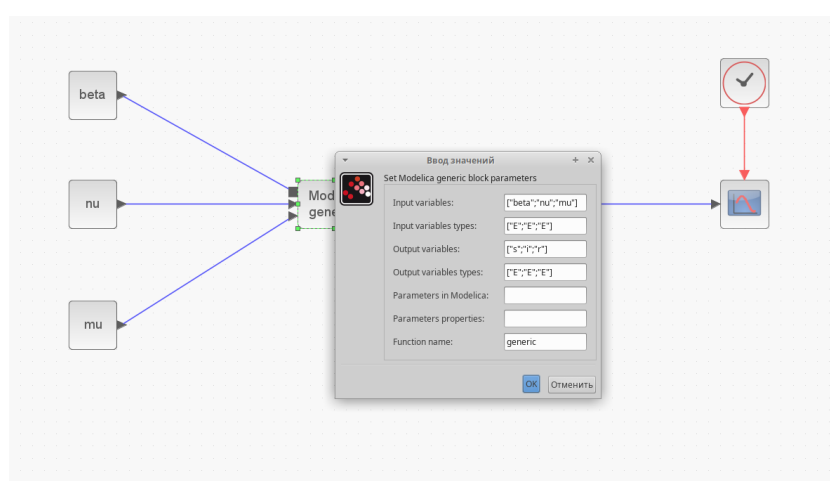


Рис. 4.29: Параметры блока Modelica для модели (с параметром μ). Ввод значений

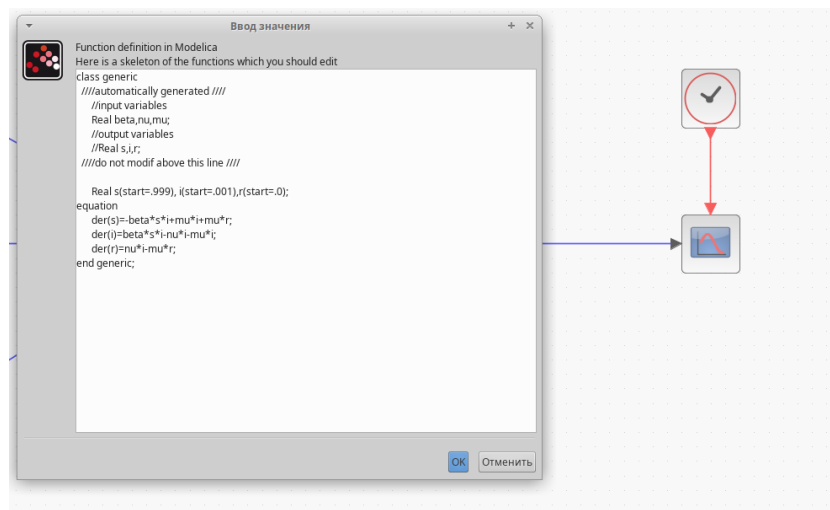


Рис. 4.30: Параметры блока Modelica для модели (с параметром μ). Ввод значений - функция

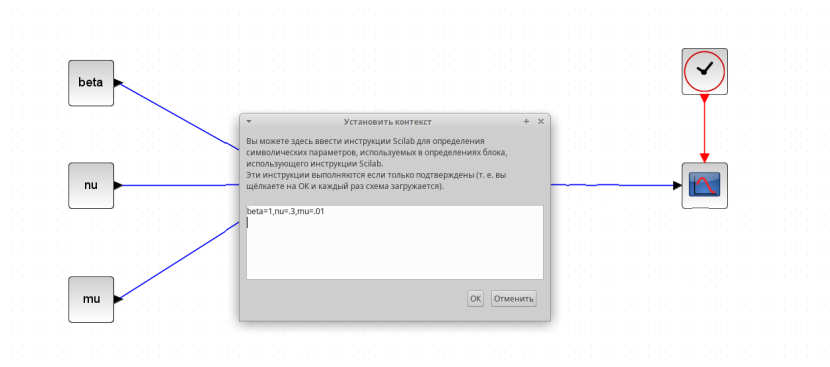


Рис. 4.31: Установить контекст в Xcos (с параметром μ)

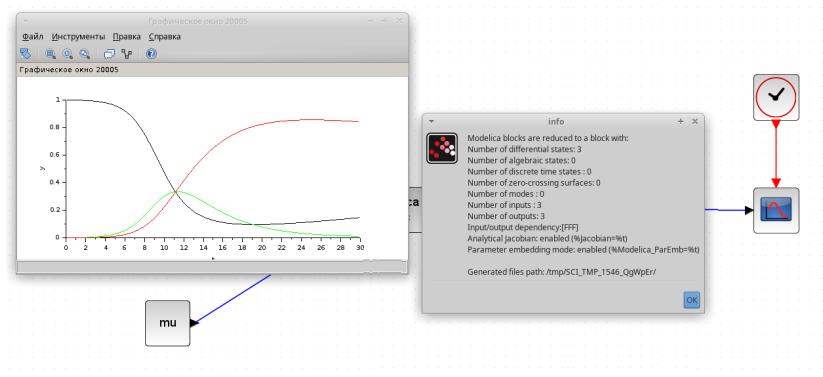


Рис. 4.32: Эпидемический порог модели SIR (с параметром μ), график построен с помощью блока Modelica в xcos

И давайте рассмотрим **второй вариант**, который реализуется без раскрытия скобок (рис. 4.33, 4.34, 4.35, 4.36, 4.37).

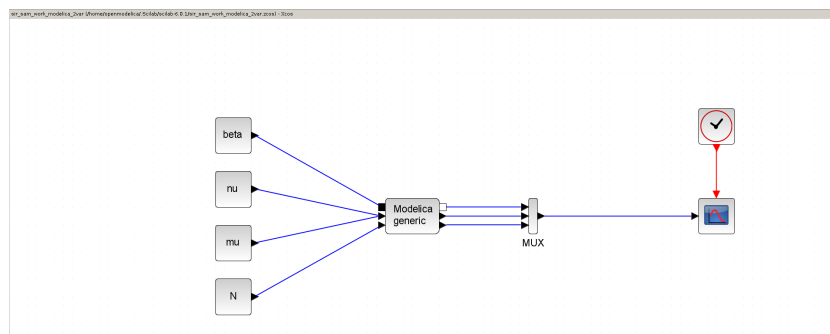


Рис. 4.33: Модель с помощью блока Modelica (с параметром μ и N) в xcos

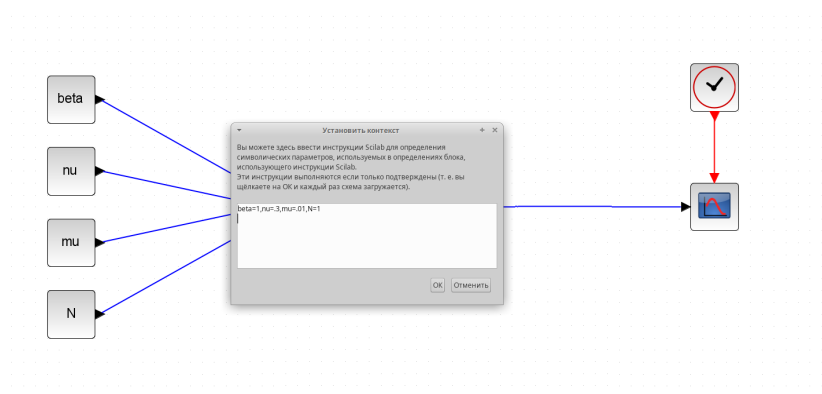


Рис. 4.34: Установить контекст в Xcos (с параметром μ и N)

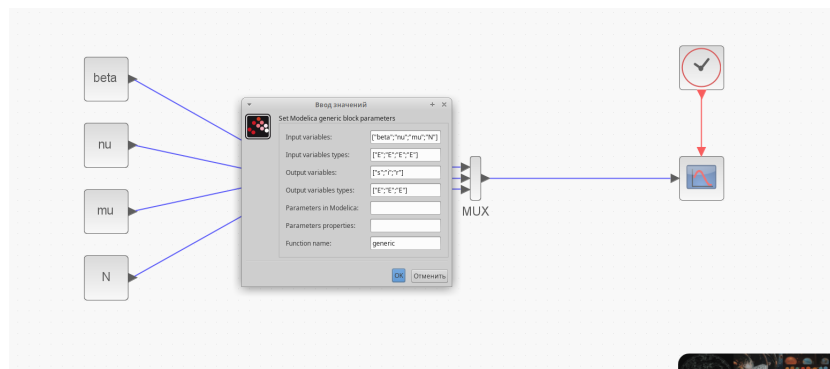


Рис. 4.35: Параметры блока Modelica для модели (с параметром μ и N). Ввод значений

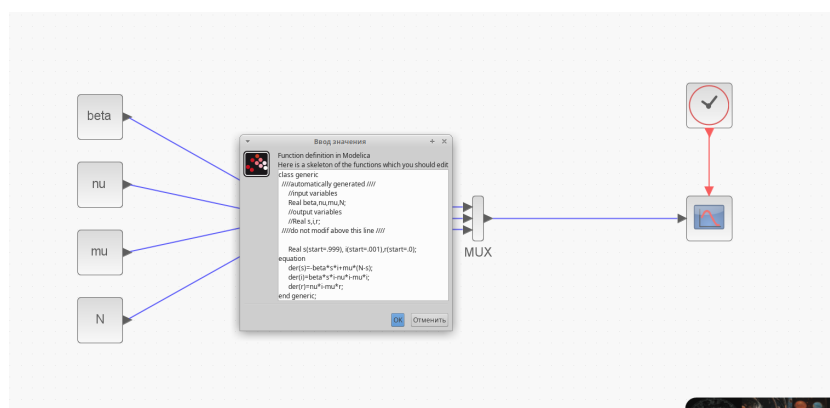


Рис. 4.36: Параметры блока Modelica для модели (с параметром μ и N). Ввод значений - функция

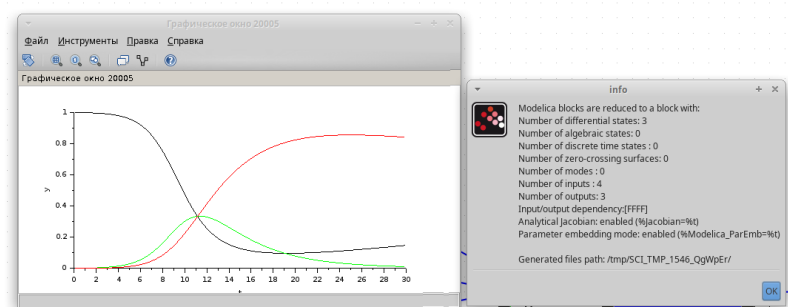


Рис. 4.37: Эпидемический порог модели SIR (с параметром μ и N), график построен с помощью блока Modelica в xcos

4.3.3 Реализация модели в OpenModelica

Реализуем модель SIR с учетом демографических процессов на OpenModelica.

Для начала я создала класс, а после написала код (рис. 4.38, 4.39).

```
1  model sir_sam_work3
2    parameter Real beta = 1;
3    parameter Real nu = 0.3;
4    parameter Real mu = 0.01;
5
6    Real s(start=0.999);
7    Real i(start=0.001);
8    Real r(start=0);
9
10   equation
11   // N = s+i+r -> N-s = i+r
12   der(s)=-beta*s*i + mu*i + mu*r;
13   der(i)=beta*s*i-nu*i - mu*i;
14   der(r)=nu*i - mu*r;
15 end sir_sam_work3;
```

Рис. 4.38: Код для модели SIR в OpenModelica (с параметром μ)

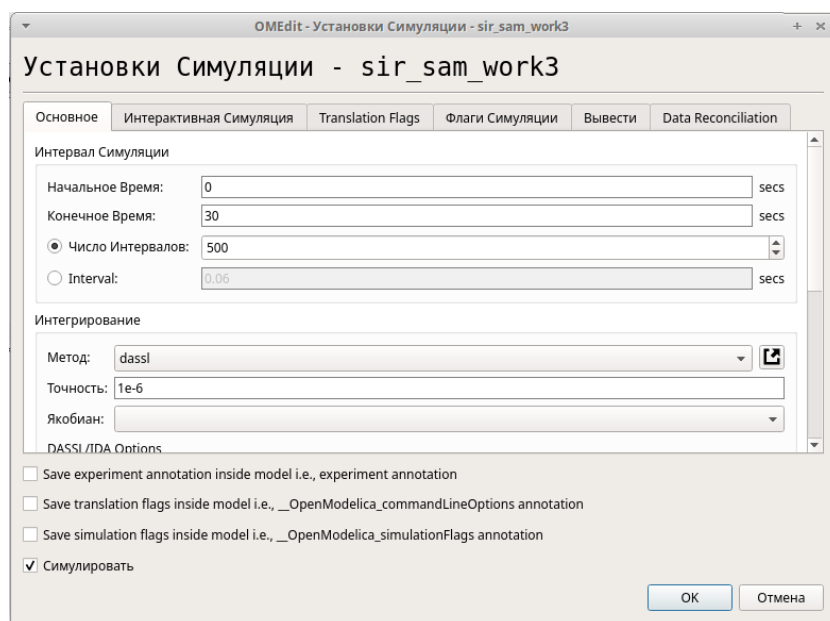


Рис. 4.39: Установки симуляции (конечное время = 30)

Результат моделирования выглядит вот так (абсолютно такой же, что и предыдущие – значит, код верный) (рис. 4.40)

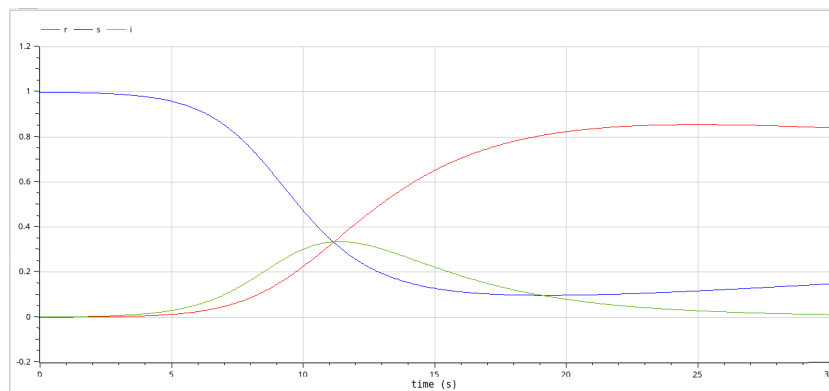


Рис. 4.40: Эпидемический порог модели SIR (с параметром μ), график построен в OpenModelica

И давайте рассмотрим **второй вариант**, который реализуется без раскрытия скобок (рис. 4.41, 4.42)

```

1  model sir_sam_work_var2
2    parameter Real N = 1;
3    parameter Real beta = 1;
4    parameter Real nu = 0.3;
5    parameter Real mu = 0.01;
6
7    Real s(start=0.999);
8    Real i(start=0.001);
9    Real r(start=0);
10
11  equation
12    der(s)=-beta*s*i + mu*(N-s);|
13    der(i)=beta*s*i-nu*i - mu*i;
14    der(r)=nu*i - mu*r;
15  end sir_sam_work_var2;

```

Рис. 4.41: Код для модели SIR в OpenModelica (с параметром μ и N)

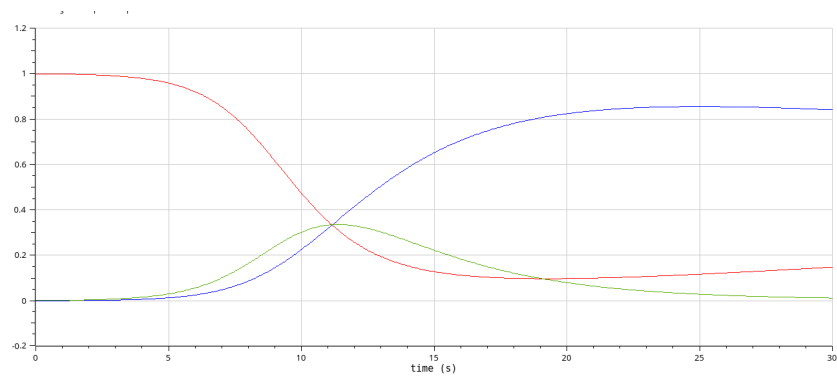


Рис. 4.42: Эпидемический порог модели SIR (с параметром μ и N), график построен в OpenModelica

4.3.4 Графики эпидемического порога при различных значениях параметров модели

Далее я буду приводить примеры графиков с различными параметрами (рис. 4.43, 4.44, 4.45, 4.46, 4.47, 4.48)

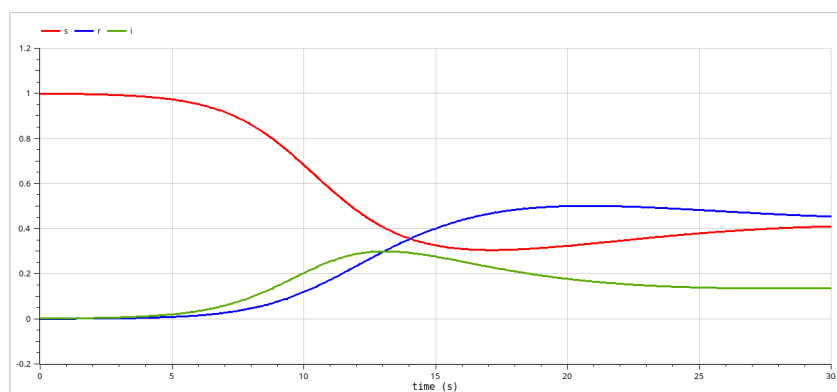


Рис. 4.43: $\beta = 1, \nu = 0.3, \mu = 0.1$

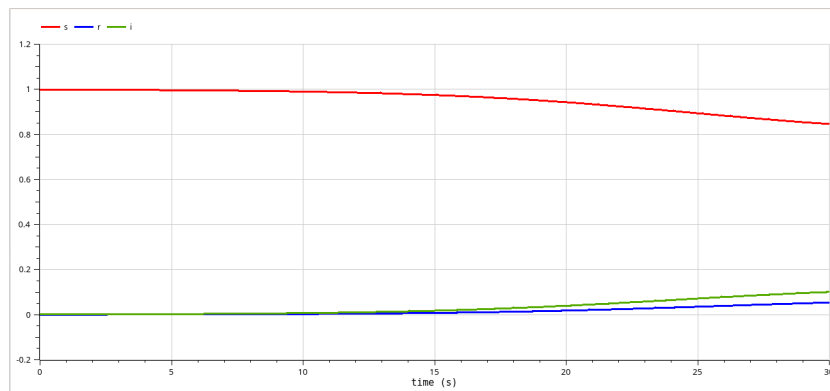


Рис. 4.44: $\beta = 1, \nu = 0.3, \mu = 0.5$

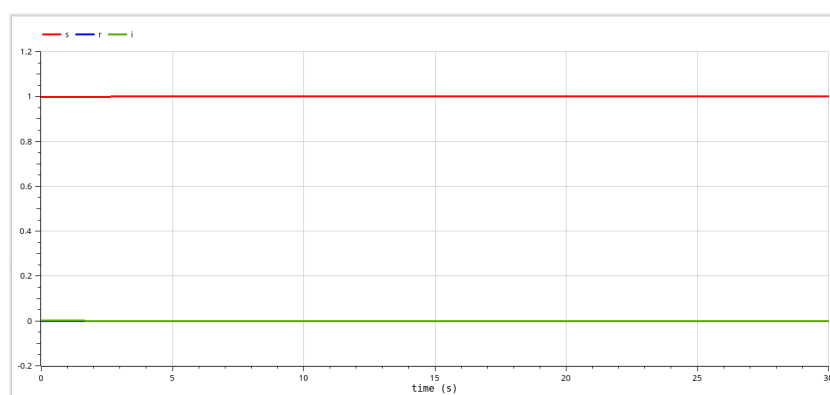


Рис. 4.45: $\beta = 1, \nu = 0.3, \mu = 1$

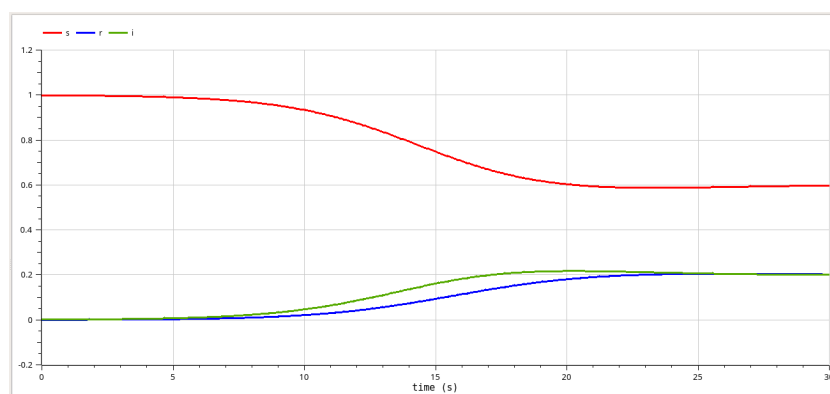


Рис. 4.46: $\beta = 1, \nu = 0.3, \mu = 0.3$

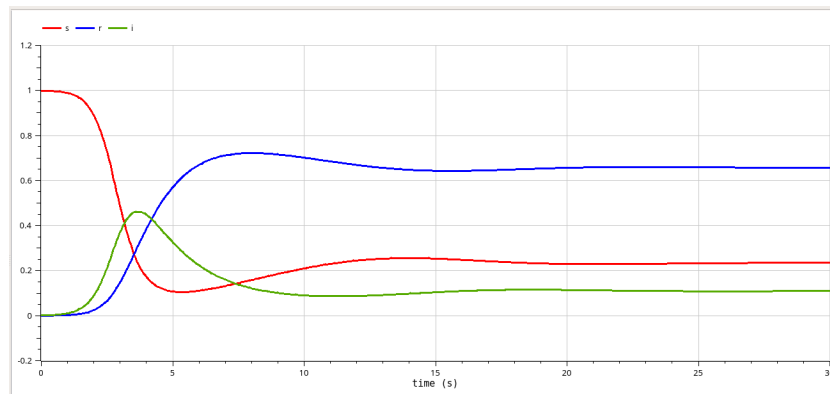


Рис. 4.47: $\beta = 3, \nu = 0.6, \mu = 0.1$

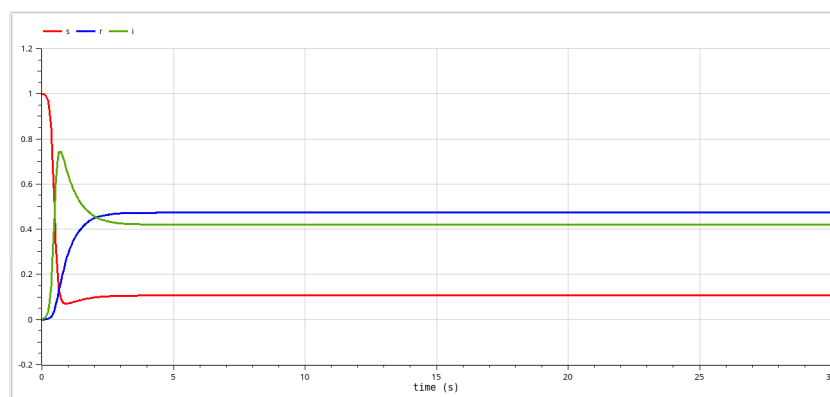


Рис. 4.48: $\beta = 16, \nu = 0.9, \mu = 0.8$

Приведу некоторые размышления по поводу графиков:

Изменение β (скорости заражения):

Параметр β (скорость заражения) оказывает существенное влияние на динамику эпидемии. Чем выше β , тем быстрее распространяется болезнь и тем больше людей заражается.

- $\beta = 3$: Высокая скорость заражения приводит к быстрому распространению эпидемии. Число инфицированных быстро растёт, достигает пика, а затем снижается.
- $\beta = 1$: Низкая скорость заражения приводит к медленному или незначительному распространению эпидемии. Число инфицированных остаётся

низким или постепенно увеличивается.

Изменение ν (скорости выздоровления):

Более высокая скорость выздоровления способствует быстрому снижению числа инфицированных после пика.

• Изменение μ (коэффициента смертности и рождаемости):

В самых базовых моделях SIR (Susceptible - восприимчивые, Infected - инфицированные, Recovered - выздоровевшие), которые часто используются для начального анализа, рождаемость и смертность (и, следовательно, параметр μ) обычно не учитываются. Это делается для упрощения модели и сосредоточения внимания на динамике распространения инфекции. В таких моделях население считается постоянным.

- $\mu = 1$: Высокий коэффициент μ приводит к быстрому обороту популяции, что предотвращает распространение эпидемии.
- $\mu = 0.5$: Средний коэффициент μ позволяет эпидемии распространяться медленно.
- $\mu = 0.1$: Низкий коэффициент μ позволяет эпидемии распространяться быстрее, но в совокупности с изменениями других коэффициентов быстрый рост сопровождается и быстрым ростом “переболевших”.

$$\beta = 3, \nu = 0.6, \mu = 0.1$$

- Начальная популяция восприимчивых быстро уменьшается, поскольку болезнь распространяется.
- Число инфицированных быстро растет, достигая пика, а затем постепенно снижается.
- Число выздоровевших увеличивается и стабилизируется на определенном уровне.

- Система достигает состояния равновесия, где популяция восприимчивых стабилизируется на уровне около 0.25, инфицированных - на уровне около 0.1, а выздоровевших - на уровне около 0.65.

$$\beta = 1, \nu = 0.3, \mu = 1$$

- Число восприимчивых остается неизменным.
- Число инфицированных остается очень низким.
- Число выздоровевших также остается неизменным.
- Болезнь не может распространиться.

$$\beta = 1, \nu = 0.3, \mu = 0.5$$

- Число восприимчивых немного уменьшается.
- Число инфицированных остается низким, но постепенно увеличивается.
- Число выздоровевших также постепенно увеличивается.

$$\beta = 1, \nu = 0.3, \mu = 0.1$$

- Начальная популяция восприимчивых значительно уменьшается.
- Число инфицированных быстро растет, достигая пика, а затем снижается.
- Число выздоровевших увеличивается и стабилизируется на определенном уровне.
- Система достигает состояния равновесия, где популяция восприимчивых стабилизируется на уровне около 0.4, инфицированных - на уровне около 0.17, а выздоровевших - на уровне около 0.45.

5 Выводы

Я получила практические навыки работы со средствами моделирования xcos, Modelica и OpenModelica. Была рассмотрена модель эпидемии (SIR).

Список литературы

1. Руководство к выполнению упражнения [Электронный ресурс]. URL: <https://esystem.rudn.ru/mod/resource/view.php?id=1223346>.