Лабораторная работа № 5

Симулирование модели эпидемии SIR

Хамдамова Айжана

Содержание

Список иллюстраций

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 Выполнение лабораторной работы

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.

3.1 Реализация модели в xcos

Зафиксируем начальные данные: $\beta = 1$, $\nu = 0,3$, $s(0) = 0,999$, $i(0) = 0,001$, $r(0) = 0$.

В меню Моделирование, Установить контекст зададим значения переменных β и ν (рис. [3.1]).

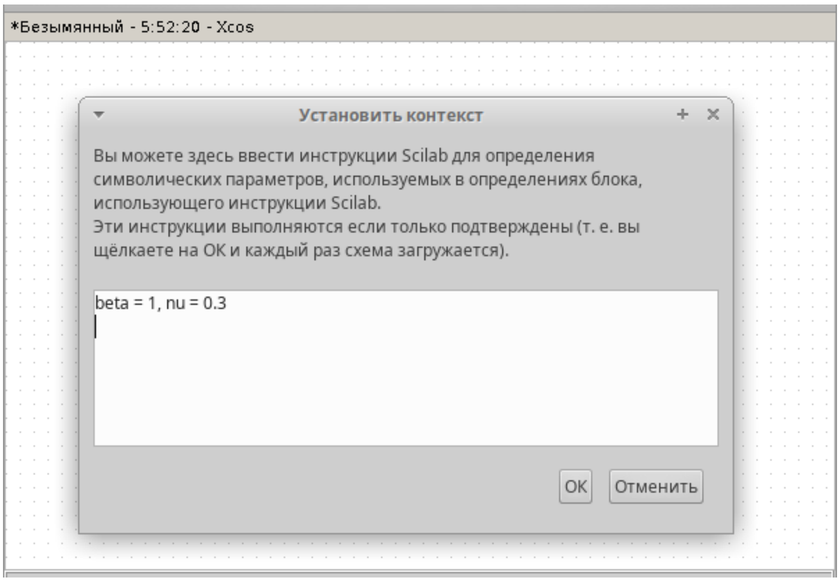


Рис. 3.1: Задание переменных окружения в xcos

Для реализации модели (рис. [3.2]) потребуются следующие блоки xcos:

- `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.

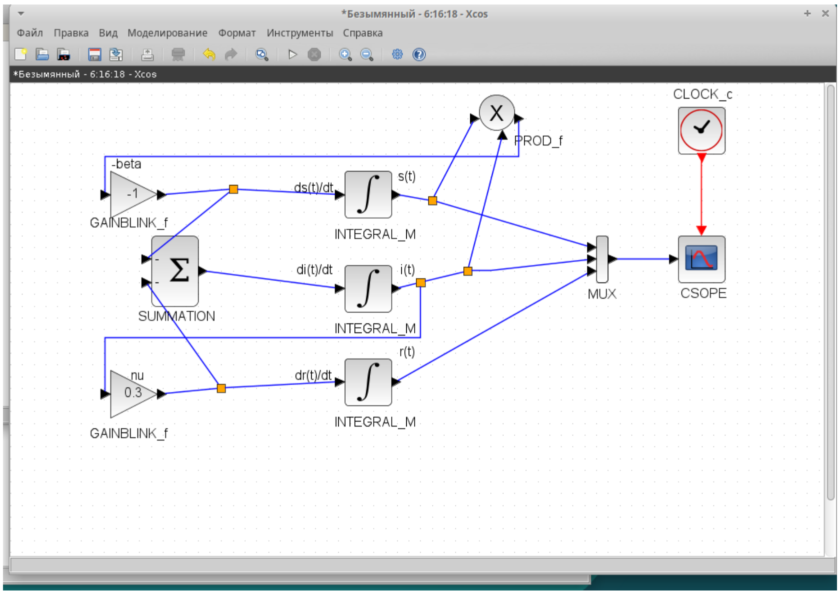


Рис. 3.2: Модель SIR в xcos

В параметрах верхнего и среднего блока интегрирования необходимо задать начальные значения $s(0) = 0,999$ и $i(0) = 0,001$ (рис. [3.3],[3.4]).

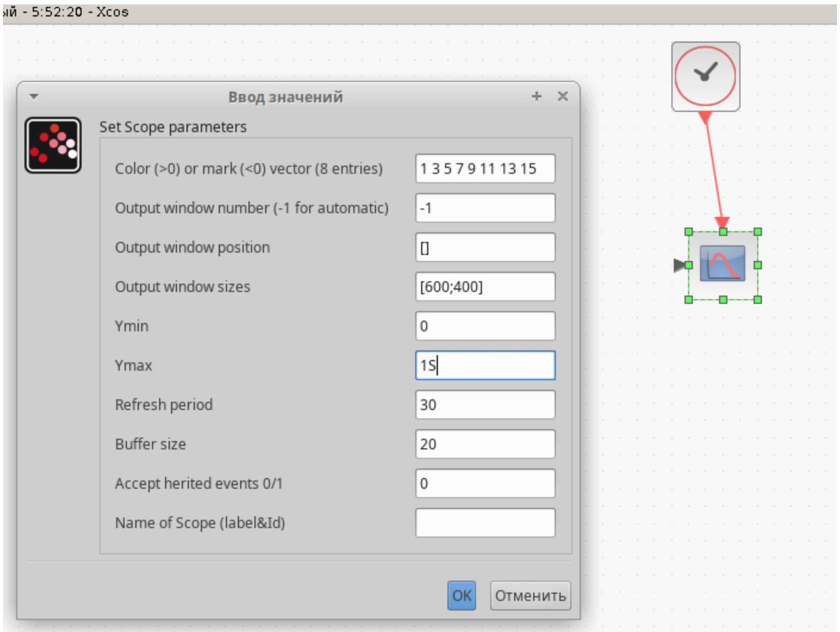


Рис. 3.3: Задание начальных значений в блоках интегрирования

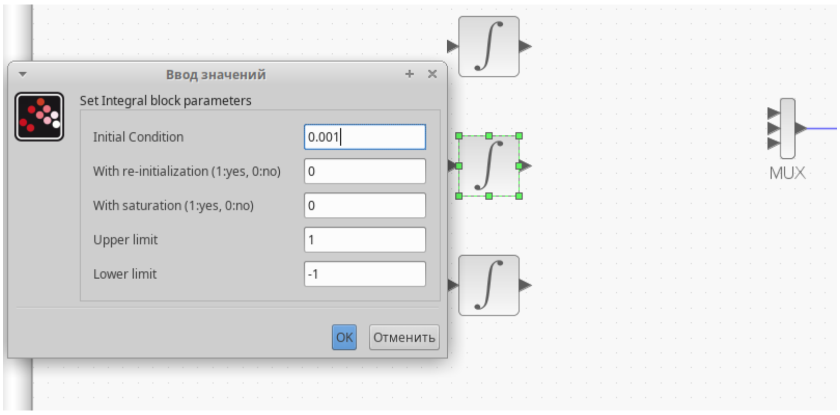


Рис. 3.4: Задание начальных значений в блоках интегрирования

В меню Моделирование, Установка зададим конечное время интегрирования, равным времени моделирования, в данном случае 30 (рис. [3.5]).

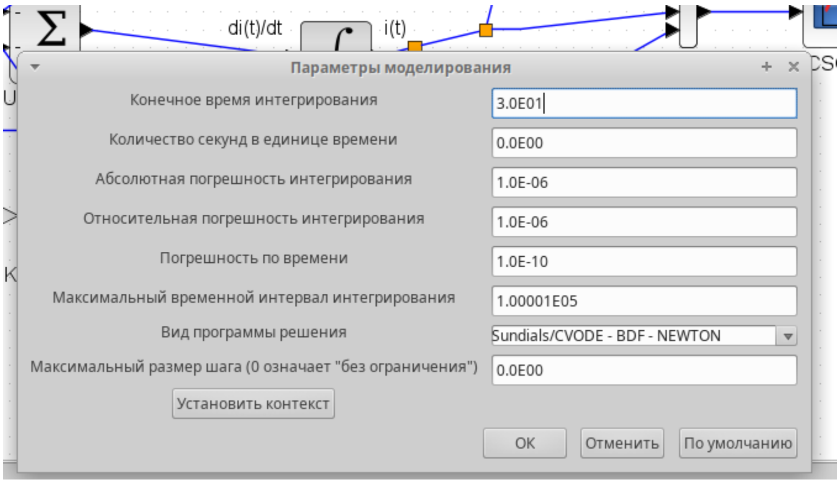


Рис. 3.5: Задание конечного времени интегрирования в xcos

Результат моделирования представлен на рис. [3.6], где черной линией обозначен график $s(t)$ (динамика численности уязвимых к болезни особей), красная линия определяет $r(t)$ — динамику численности выздоровевших особей, наконец, зеленая линия определяет $i(t)$ — динамику численности заражённых особей. Пересечение трёх линий определяет порог эпидемии.

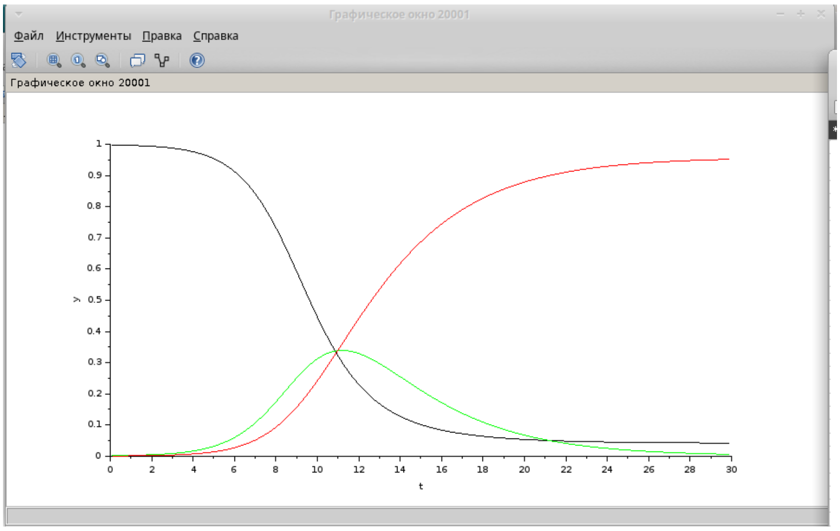


Рис. 3.6: Эпидемический порог модели SIR при $\beta = 1, \nu = 0.3$

3.2 Реализация модели с помощью блока Modelica в xcos

Готовая модель SIR представлена на рис. [3.7].

Для реализации модели SIR 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. Задаём значения переменных β и ν (рис. [3.1]).

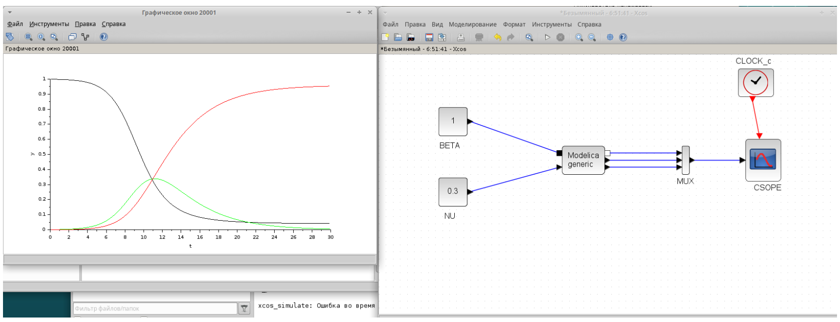


Рис. 3.7: Модель SIR в xcos с применением блока Modelica

Параметры блока Modelica представлены на рис. [3.8],[3.9]. Переменные на входе (“beta”, “nu”) и выходе (“s”, “i”, “r”) блока заданы как внешние (“E”).

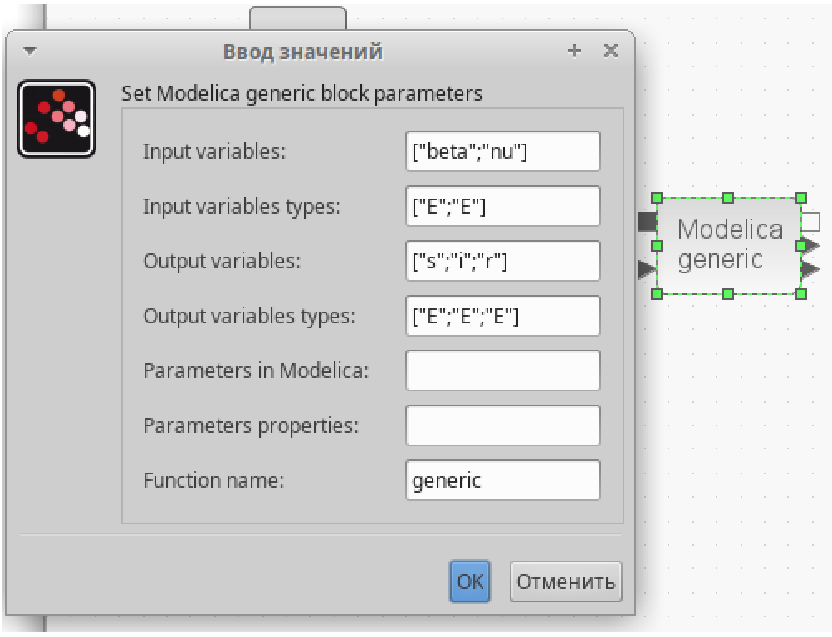


Рис. 3.8: Параметры блока Modelica для модели SIR

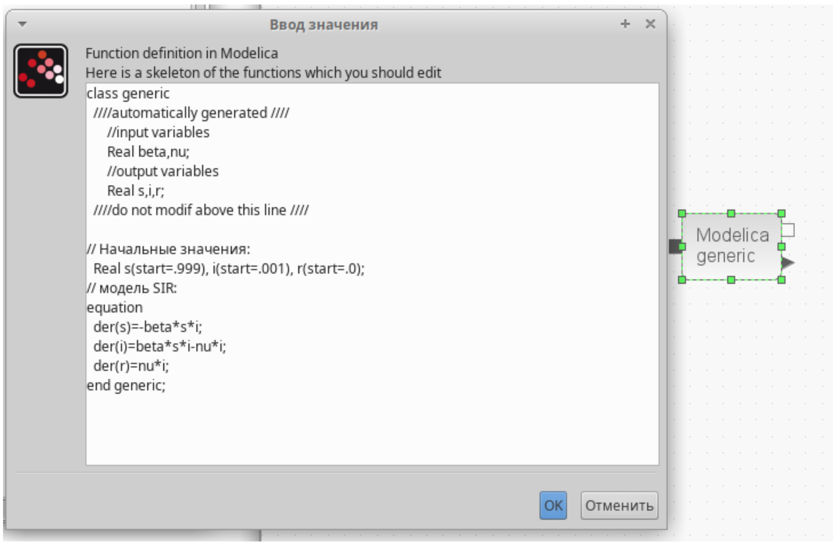


Рис. 3.9: Параметры блока Modelica для модели SIR

В результате получаем график (рис. [3.10]), построенный с помощью блока

Modelica идентичный графику (рис. [3.6]), построенному без них.

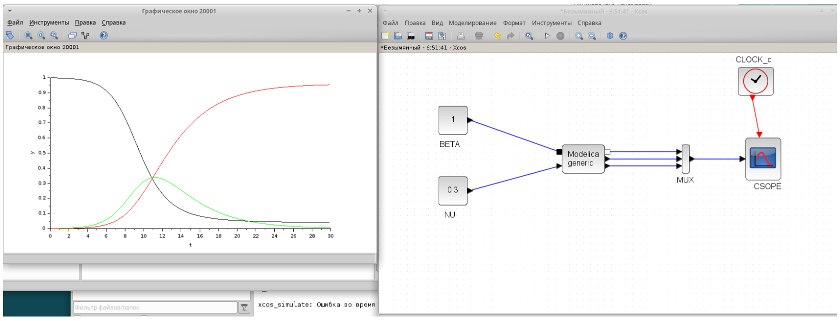


Рис. 3.10: Эпидемический порог модели SIR при $\beta = 1, \nu = 0.3$

3.3 Упражнение

В качестве упражнения нам надо построить модель SIR на OpenModelica. Синтаксис почти такой же как и на Modelica. Нужно задать параметры, начальные значения и систему дифференциальных уравнений (рис. [3.11]).

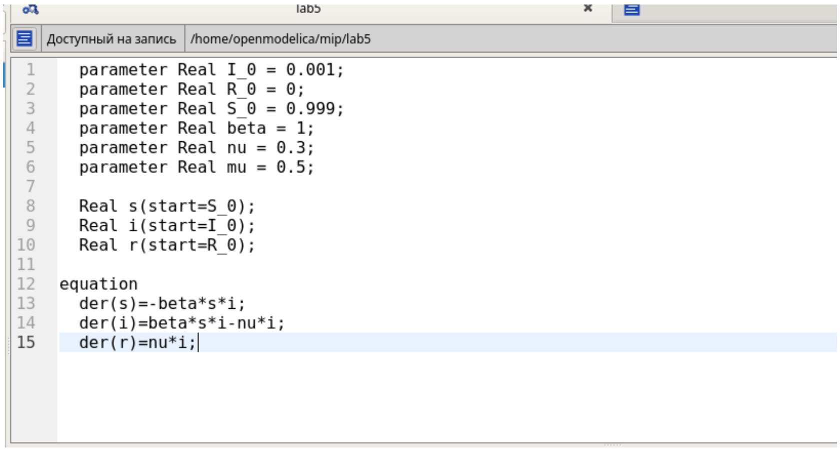


Рис. 3.11: Установка симуляции в OpenModelica

Теперь выполним симуляции, задав конечное время 30 с (рис. [3.12]).

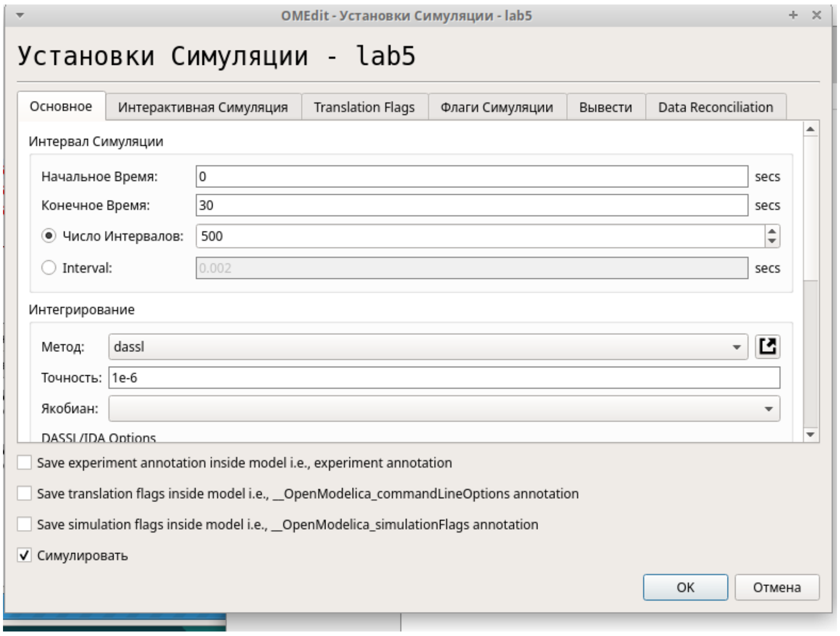


Рис. 3.12: Установка симуляции в OpenModelica

В результате получаем следующий график (рис. [3.13]). Он идентичен предыдущим графикам выполненным в $xcos$.

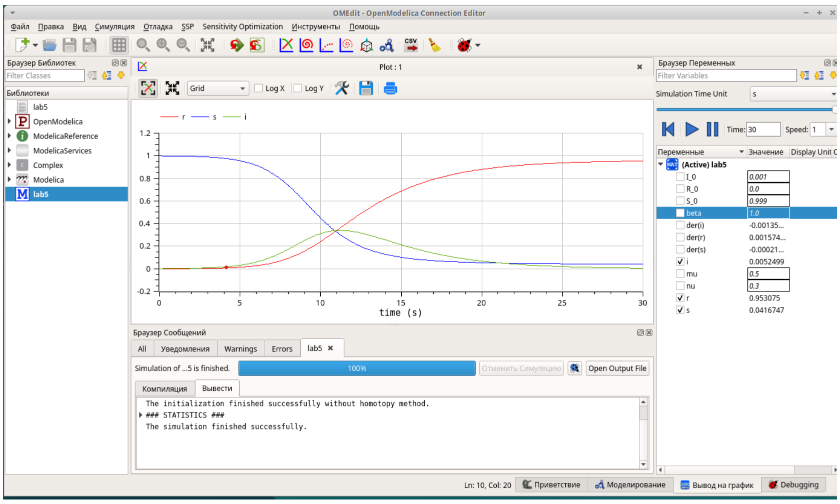


Рис. 3.13: Эпидемический порог модели SIR при $\beta = 1, \nu = 0.3$

3.4 Задание для самостоятельного выполнения

Предположим, что в модели SIR учитываются демографические процессы, в частности, что смертность в популяции полностью уравновешивает 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.

Реализуем эту модель в *xcos*. Тут нам понадобятся три блока суммирования и 4 блока констант (добавляется константа ν). (рис. [3.14]).

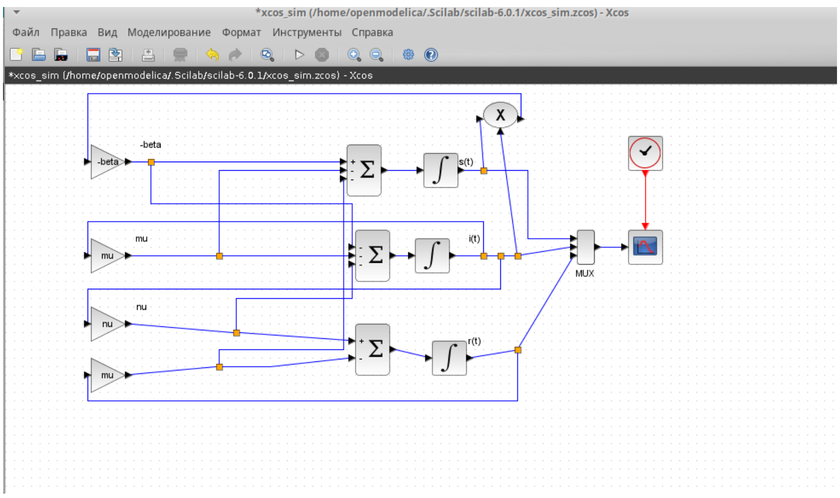


Рис. 3.14: Модель SIR с учетом демографических процессов в xcos

В результате получаем следующий график (рис. [3.15]).

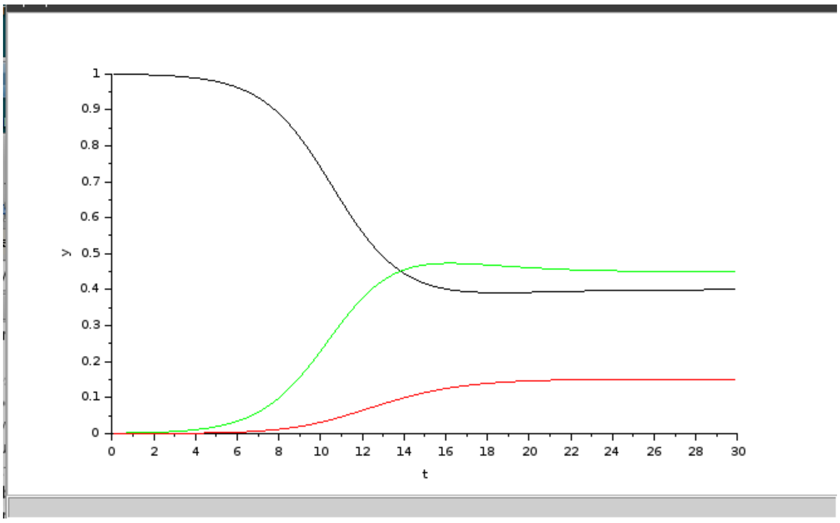


Рис. 3.15: График модели SIR с учетом демографических процессов

Теперь реализуем модель SIR с учетом демографических процессов в *xcos* с помощью блоков Modelica (рис. [3.16]).

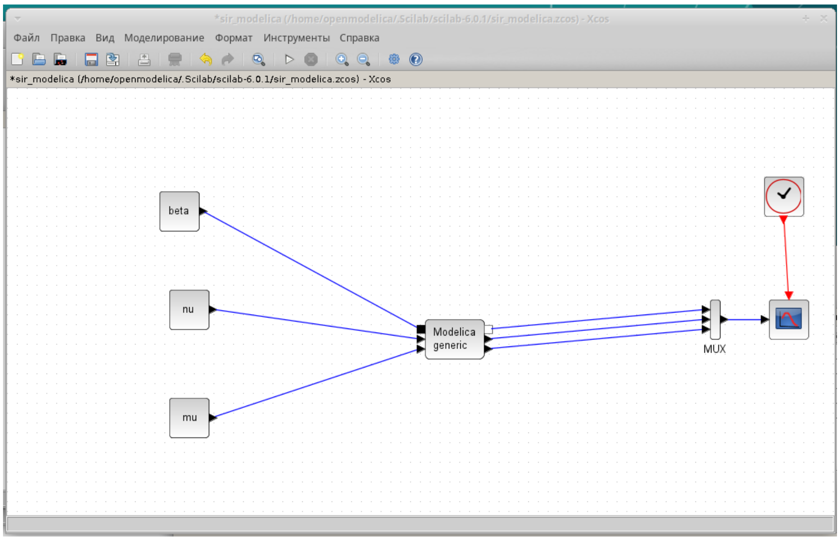


Рис. 3.16: Модель SIR с учетом демографических процессов в xcos с применением блока Modelica

В результате получаем следующий график (рис. [3.17]).

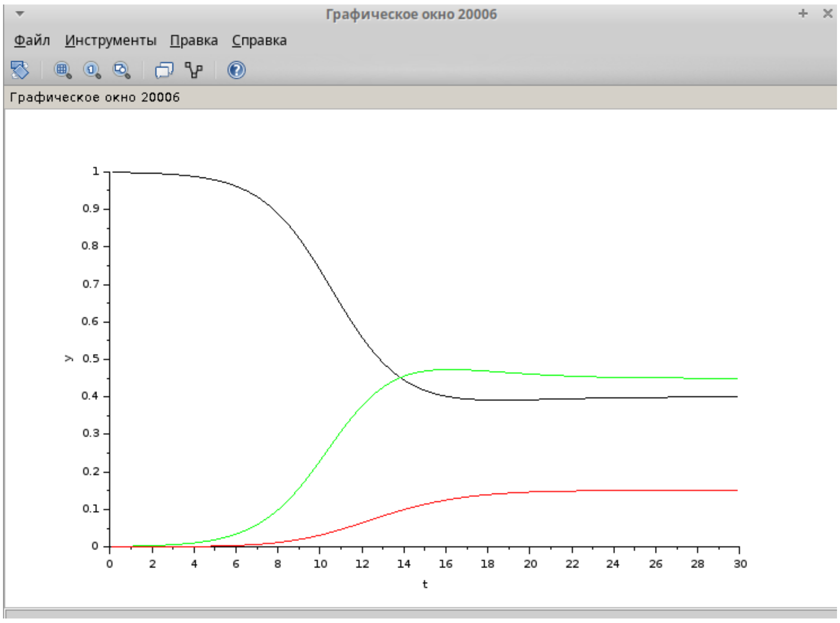


Рис. 3.17: График модели SIR с учетом демографических процессов

Реализуем модель SIR с учетом демографических процессов на OpenModelica.(рис. [3.18])

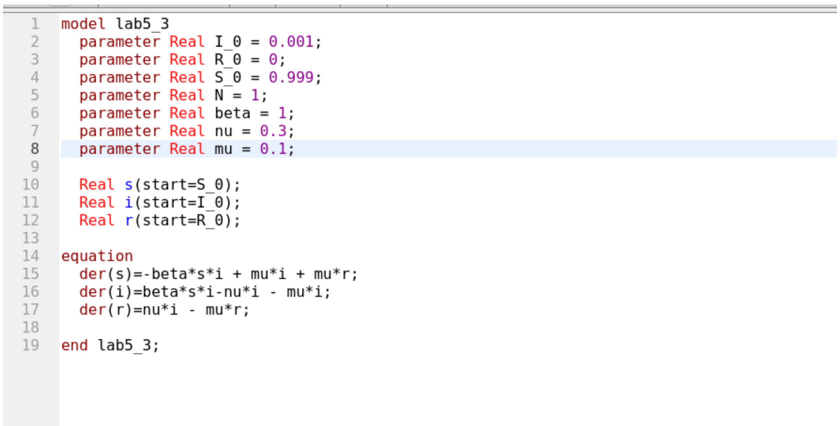
```
  parameter Real I_0 = 0.001;
  parameter Real R_0 = 0;
  parameter Real S_0 = 0.999;
  parameter Real N = 1;
  parameter Real beta = 1;
  parameter Real nu = 0.3;
  parameter Real mu = 0.1;

  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;



```
model lab5_3
  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 = 0.1;

  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 lab5_3;
```

Рис. 3.18: Задание параметров в open modelica

Выполнив симуляцию, получим следующий график (рис. [3.19]).

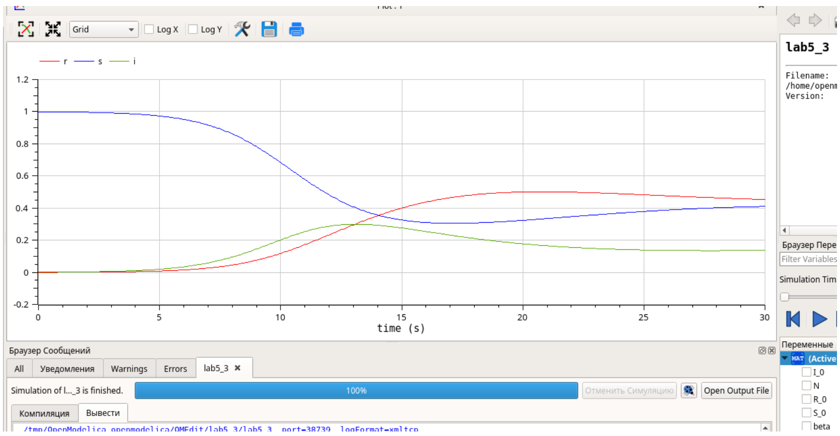


Рис. 3.19: График модели SIR с учетом демографических процессов

Теперь построим графики при разных значениях параметров.

$\beta = 1, \nu = 0.3$

- $\mu = 0.1$

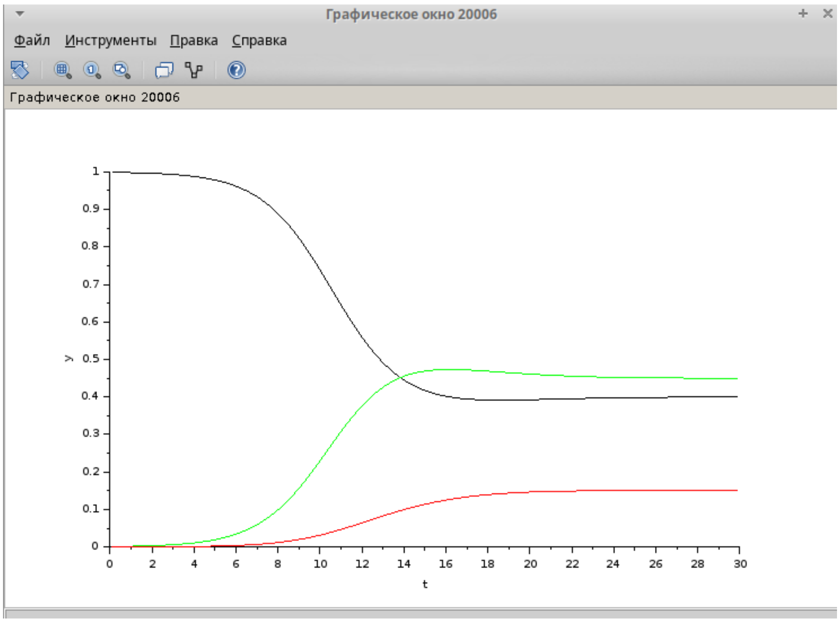


Рис. 3.20: График модели SIR с учетом демографических процессов

- $\mu = 0.3$

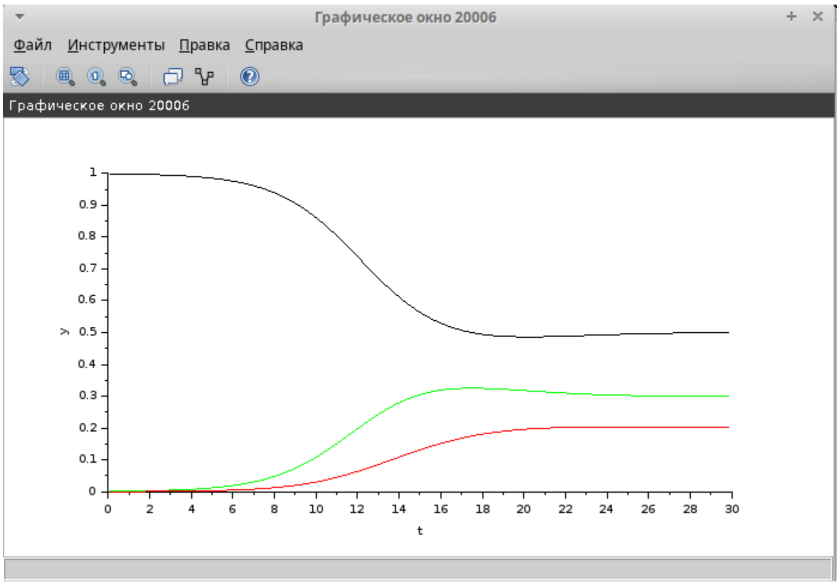


Рис. 3.21: График модели SIR с учетом демографических процессов

- $\mu = 0.5$

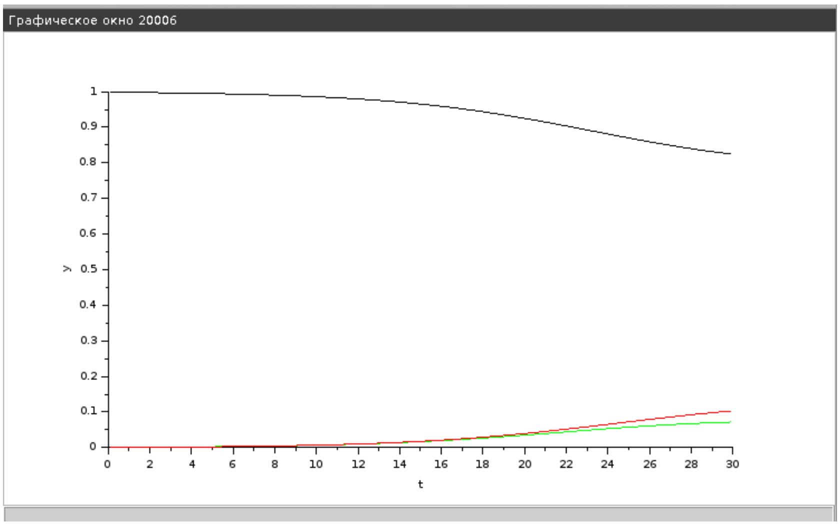


Рис. 3.22: График модели SIR с учетом демографических процессов

- $\mu = 0.9$

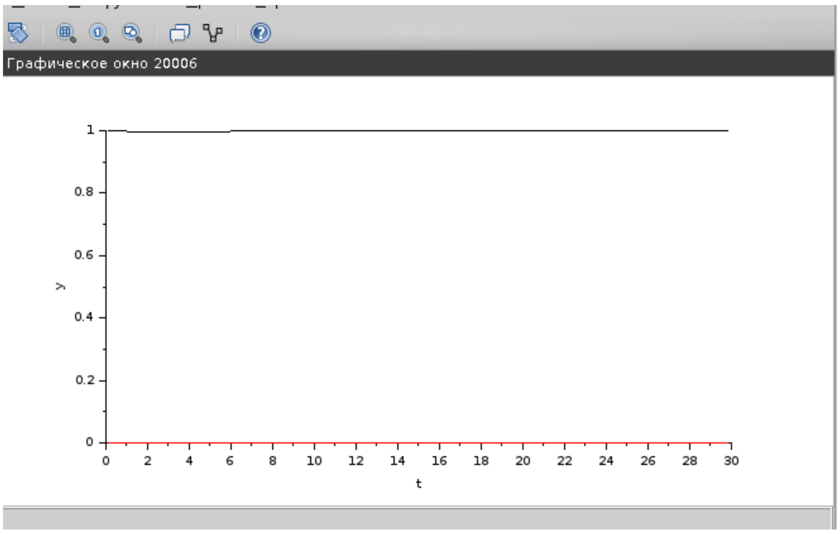


Рис. 3.23: График модели SIR с учетом демографических процессов

Исходя из анализа графиков, можно сделать вывод, что чем выше значение любого из параметров, тем быстрее система достигает стационарного состояния. При высоком коэффициенте заражения β система быстро проходит через пик развития эпидемии и достигает стационарного состояния.

4 Выводы

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