Математическое и компьютерное моделирование

**Лабораторная работа № 4**

**ОДУ В Simulink**

**Цель работы:** Научиться моделировать динамические системы, описываемые ОДУ в Simulink

**Теоретическое введение**

Имитационное моделирование (ИМ) применяется для исследования и проектирования таких сложных систем и процессов, как производственные предприятия, информационные сети, системы массового обслуживания, мировые динамики в экономике, экологии, политике и т. д.

Имитационная модель системы – это программа, в которой определяются все наиболее существенные элементы и связи изучаемой системы и задаются начальные значения параметров, соответствующие некоторому «нулевому» моменту времени, а все последующие изменения, происходящие в системе, рассчитываются на ЭВМ автоматически при выполнении программы.

Выполнение имитационной модели называется имитационным экспериментом (ИЭ). В ходе ИЭ компьютер имитирует функционирование системы и вычисляет ее параметры, представленные в модели. ИЭ может дублировать натуральный эксперимент. Однако он позволяет, в отличие от натурного метода, экспериментировать с системами, которых уже нет или которые находятся в процессе разработки, позволяет предсказывать поведение существующих систем в будущем, изучать их поведение в особых условиях.

Существуют разные типы имитационных моделей. По характеру возможных изменений переменных величин модели подразделяются на непрерывные и дискретные. В непрерывных моделях величины представляют собой непрерывные функции времени. В соответствии с этим продвижение во времени, т. е. пересчет значений переменных величин в ходе модельного времени, осуществляется в имитационной модели по принципу «малых дельта-t», т. е. следующее состояние системы определяется по ее предыдущему состоянию с малым промежутком времени между этими состояниями.

В дискретных моделях изменения происходят скачкообразно и между моментами изменений состояния элементов остаются постоянными. Изменения в дискретных моделях называют событиями.

Продвижение в модельном времени в ходе ИЭ осуществляется по принципу «от события к событию», т. е. из нулевого момента времени модель перемещается сразу к моменту t1, из него – к моменту t2 и т. д. Для реальных систем можно строить как непрерывные, так и дискретные модели.

В Simulink дифференциальные уравнений записываются в виде функциональных блоков. Это означает, что при моделировании в среде Simulink реализуется принцип визуального программирования, в соответствии с которым пользователь, с помощью копирования из библиотеки стандартных блоков, установки между ними связей и задания параметров в блоках, создает структурную модель изучаемой системы и после машинной реализации модели анализирует результаты, делает прогнозы.

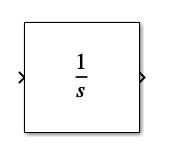
При имитационном моделировании системы пользователь может выбирать метод решения дифференциальных уравнений, способ изменения модельного времени (с фиксированным или переменным шагом). В ходе имитации имеется возможность анализировать визуально результаты моделируемого процесса. Для этого в структурную модель включаются устройства наблюдения, входящие в состав библиотеки блок-диаграмм Simulink. Результаты моделирования могут быть представлены в виде чисел, графиков, таблиц.

Изучение, как правило, начинается с простейшего типа – уравненений с разделенными переменными *dx/dt* = *f*(*t*). Общий интеграл уравнения имеет вид *x - ∫ f(t) dt = C* . Частное решение, удовлетворяющее начальному условию *x(t0) = x0* , будет иметь вид

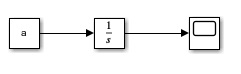
Для того, чтобы получить x из его производной *dx/dt*  нужно проинтегрировать:



В Simulink интегратор имеет вид



В случае, если уравнение *dx/dt* = *f*(*t*) рассматривается с постоянной правой частью *dx/dt* = *a* (например, *a* =1), то частное решение, удовлетворяющее начальному условию *x(t*0) = *x*0 , имеет вид *x = a(t - t0 ) + x0* . Это алгебраическое уравнение прямой на плоскости, проходящей через точку *(t0, x0)* на *R*2 . Для данного уравнения Simulink модель имеет следующий вид:



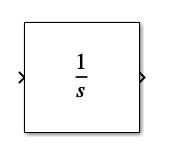
Основным конструктивным элементом модели является интегратор. На вход интегратора подается сигнал заданной формы f(t)=a, описывающий неоднородность уравнения (правая часть). С выхода интегратора снимается сигнал *x*(t), представляющий решение уравнения. Решение рассматривается во временном промежутке [t0,T]. Начальные условия задаются в параметрах блока интегратора.

Более сложно реализуется уравнение вида *dx/dt = f(x,t)*. В этом случае правая часть содержит саму неизвестную функцию, но без знака производной. В этом случае необходимые операции с функцией *x(t)* производятся после интегрирования.

**Как решать дифференциальные уравнения в Simulink**

**1. Обыкновенные дифференциальные уравнения первого порядка**

В браузере библиотеки Simulink есть блок с именем Integrator, показанный на рисунке ниже

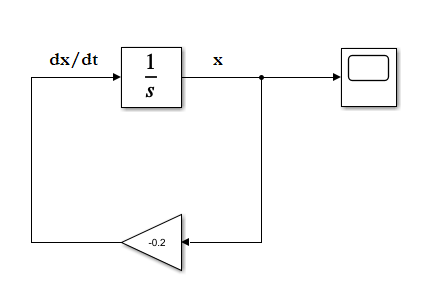


Как следует из названия, этот блок используется для вычисления интеграла от сигнала, подаваемого на вход, то есть с левой стороны блока. В случае решения дифференциального уравнения главное, что мы должны сделать, - это проинтегрировать данное уравнение и получить функцию без производной. Количество интегральных блоков, используемых в блок-схеме, равно порядку дифференциального уравнения, которое мы собираемся решить в задаче. Например, если мы хотим решить дифференциальное уравнение 1-го порядка, нам понадобится 1 интегральный блок, а если уравнение является дифференциальным уравнением 2-го порядка, количество используемых блоков равно двум. Давайте теперь рассмотрим простой пример с использованием Simulink, в котором мы будем решать дифференциальное уравнение первого порядка.

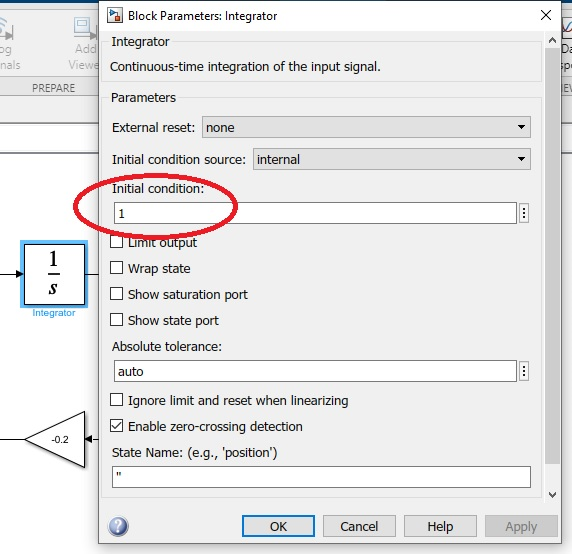
**Пример 1.** Нужно решить уравнение

с начальным условием

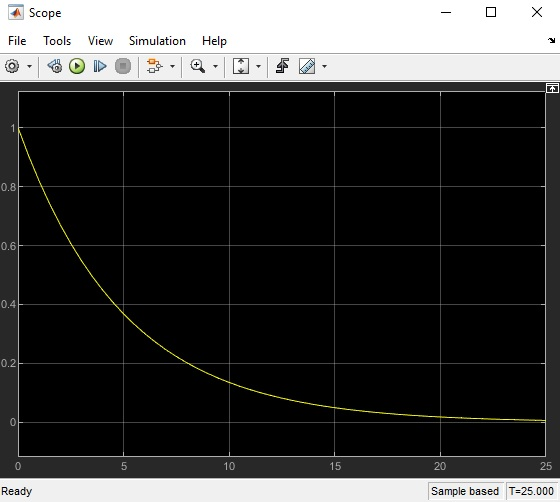
Используя интегратор, осциллограф и блок умножения на число составим схему:



Начальные условия задаются в параметрах интегратора. Для этого нужно дважды щелкнуть по значку интегратора и поставить 1 в Initial conditions:



Промежуток изменения t (времени) зададим от 0 до 25. Тогда результат будет



**Пример 2.** Смоделируем простую модель популяции бактерий в чашке Петри. Модель выглядит следующим образом:

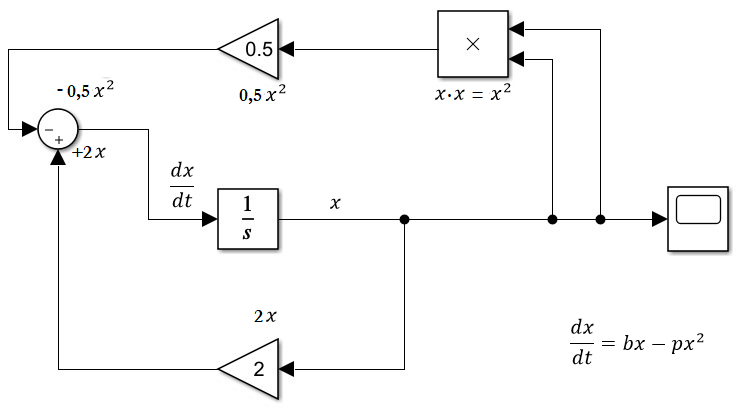
рождаемость = 𝑏𝑥

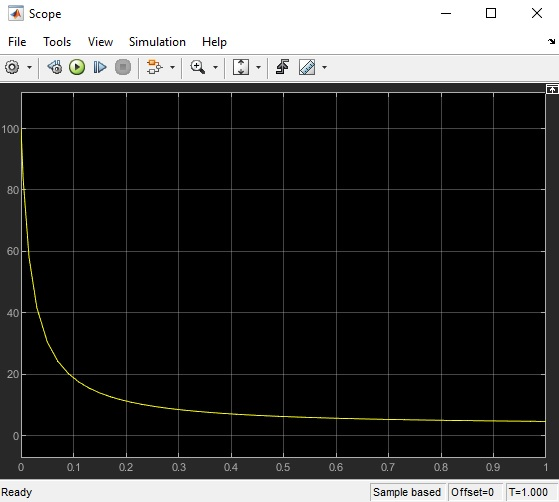
смертность = 𝑝𝑥2

Тогда общая скорость изменения популяции бактерий составит:

Пусть b = 2 бактерии в час и p = 0,5 бактерий в час.

Смоделируем количество бактерий в банке через 1 час (Stop time=1), при условии, что изначально присутствует 100 бактерий. Получим схему и результат.



****

**2. Обыкновенные дифференциальные уравнения второго порядка**

**Колебания пружинного маятника**

*Модель 1*

*Пружинный маятник с безмассовой пружиной*

Рассмотрим свободные гармонические колебания на примере пружинного маятника – груза массой *m*, подвешенногона пружине с коэффициентом жёсткости *k*. В этой модели мы будем пренебрегать массой самой пружины.

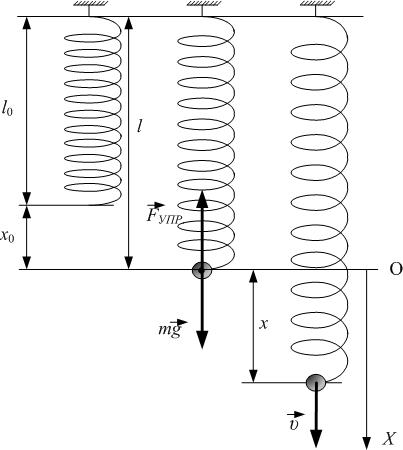


Рис.1

Пусть длина недеформированной пружины (без груза) равна ; с грузом массой *m* длина пружины в состоянии равновесия равна *l*, то есть увеличивается на  (рис.1). Сила упругости уравновешивается силой тяжести:

. (1)

Груз, выведенный из положения равновесия и предоставленный сам себе, начинает колебаться вдоль вертикальной оси. За начало отсчёта удобно принять равновесное положение груза, а ось *OX* направить вниз, то есть,  – это смещение груза из положения равновесия,  – его скорость. Изменение длины пружины от первоначального недеформированного состояния будет равно . Энергия системы складывается из потенциальной энергии упругой деформации пружины:

, (2)

потенциальной энергии груза в поле силы тяжести:

 (3)

и кинетической энергии груза:

. (4)

Полная механическая энергия системы при отсутствии сил трения (сопротивления среды) сохраняется:

. (5)

Продифференцируем (5) по времени:

. (6)

Поскольку , то после сокращения на  получим из (6):

, или

, (7)

так как . С учётом (1) получим из (7):

. (8)

Разделив (8) на *m*, получим дифференциальное уравнение гармонических колебаний вида:

. (9)

Общее решение этого дифференциального уравнения имеет вид:

, (10)

где  – частота собственных колебаний маятника с безмассовой пружиной. Период колебаний такого маятника

. (11)

Заметим, что наличие поля силы тяжести не влияет на частоту колебаний пружинного маятника; изменяется лишь равновесное положение груза. Так что в дальнейшем потенциальную энергию в поле силы тяжести можно не учитывать.

Для нахождения констант в (10) нужно дополнить модель (9) начальными условиями:

– начальное отклонение пружины,

– начальная скорость пружины.

Тогда

,

,

.

Получим

Построим эту модель в Simulink. Зададим колебания уравнением

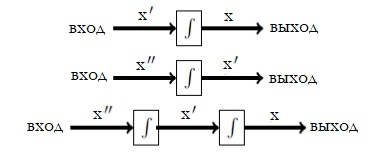


с начальными условиями

– начальное отклонение пружины,

– начальная скорость пружины.

Чтобы решить дифференциальное уравнение второго порядка с помощью Simulink нужно использовать два интегратора для вывода x’(t) и x (t).



Первый интегратор выдаст нам x’(t), поэтому в него нужно задать второе начальное условие , второй интегратор выдаст саму функцию x, поэтому в него надо задать первое начальное условие .

Пусть

k=5 – жесткость пружины,

m=2 – масса груза,

– начальное отклонение груза от положения равновесия,

– начальная скорость груза,

0≤t≤20 время выполнения.

Нам потребуются блоки

1. Integrator – 2 шт.,
2. Gain – 2 шт.,
3. Scope – 1 шт.,
4. Bus creator – 1 шт.

Блок Bus Creator комбинирует набор входных элементов в шину.

Имена блоков в модели должны быть уникальными и состоять хотя бы из одного символа. Чтобы изменить имя блока, нужно выполнить щелчок на имени, а затем, используя обычные приемы редактирования, внести необходимые изменения. Для изменения шрифта надо выделить блок, по нажатию правой клавиши мыши открыть всплывающее меню, затем выбрать команду Format - Font (шрифт).

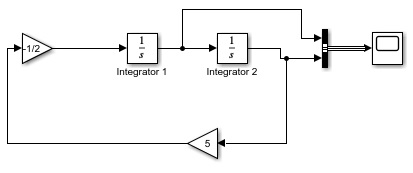
Добавим первый интегратор и в настройках зададим второе начальное условие x’(0)=0. Добавим второй интегратор после первого и для второго интегратора добавим первое начальное условие x(0)=1.

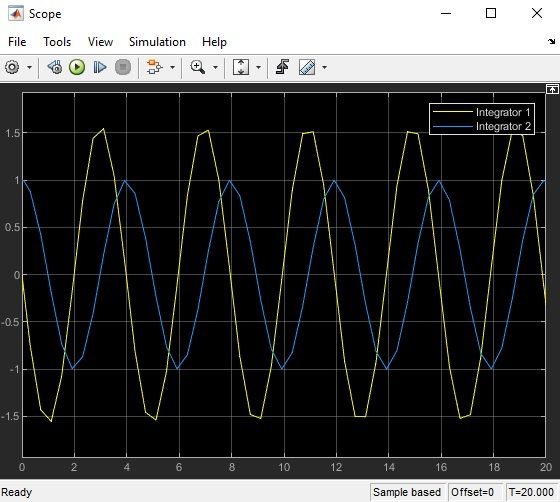
Сигнал из второго интегратора-это уже решение, то передаем его на вход Bus Creator, который соединяем с осциллографом Scope.

Теперь, записав уравнение в виде

,

смоделируем . Функцию x(t), полученную на выходе из второго интегратора, умножим на k при помощи блока Gain и, затем поделим на m при помощи блока Gain со значением -1/2. Также соединим выход из первого интегратора (скорость) со вторым входим шины Bus Creator, чтобы скорость тоже вывести на экран. Тогда получим схему и результат работы. Чтобы были подписи графиков надо в настройках экрана осциллятора поставить галочку у Legend.





**Задание 1.**

В MATLAB/Simulink смоделируйте модели популяционной динамики из Лабораторной работы Задания 2,3,4 и 5.

**Задание 2.**

При помощи Simulink yа отрезке [*a*, *b*] найти решение дифференциального уравнения вида с начальными условиями *y*(0) , *y*'(0) .

Варианты заданий представлены в табл.1. Построить графики функций y(x) и y’(x).

**Таблица 1.**

