Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Локтионова Оксана Геннадьевна
Должность: проректор по учебной работе
Дата подписания: 08.09.2026 07:50:36
Уникальный программный ключ:
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра биомедицинской инженерии

УТВЕРЖДАЮ
Проректор по учебной работе
_____ О.Г. Локтионова
«_25_» ___04_____ 2023 г.

МОДЕЛИРОВАНИЕ В МЕДИЦИНЕ И БИОЛОГИИ

Методические указания по выполнению самостоятельной работы
для студентов специальности
30.05.03 «Медицинская кибернетика»

Курск 2023

УДК 004.93:61

Составители: О.В. Шаталова.

Рецензент
Кандидат технических наук, доцент *А.В. Киселев*

Моделирование в медицине и биологии: методические указания по выполнению самостоятельной работы для студентов специальности 30.05.03 «Медицинская кибернетика» / Юго-Зап. гос. ун-т; сост.: О.В. Шаталова. - Курск, 2023. - 35 с.

Предназначено для студентов специальности 30.05.03 «Медицинская кибернетика» по дисциплине «Моделирование в медицине и биологии». Может быть использована аспирантами, обучающимися по направлению подготовки 1.5.8. Математическая биология, биоинформатика.

Текст печатается в авторской редакции

Подписано в печать . Формат 60×84 1/16.
Усл. печ. л. 2,03. Уч.-изд. л. 1,84. Тираж 100 экз. Заказ 982. Бесплатно.
Юго-Западный государственный университет.
305040, г. Курск, ул. 50 лет Октября, 94.

1 Введение в Simulink

Краткие сведения о пакете

Система структурного моделирования Simulink предназначена для компьютерной реализации математических моделей динамических систем и устройств, представленных функциональной блок-схемой или системой уравнений. При этом возможны различные варианты моделирования: во временной области, в частотной области, с событийным управлением и т.д.

Для построения функциональной блок-схемы моделируемых устройств Simulink имеет обширную библиотеку блочных компонентов и удобный редактор блок-схем. Он основан на графическом интерфейсе пользователя и по существу является типичным средством визуально-ориентированного программирования. Используя библиотеку компонентов решающих блоков, пользователь с помощью мыши переносит нужные блоки из библиотеки в рабочее окно пакета Simulink и соединяет линиями связи входы и выходы блоков. Таким образом, создается блок-схема системы или устройства, то есть компьютерная модель.

Порядок работы с пакетом Simulink следующий:

1 На рабочем столе открываем пиктограмму MATLAB.

2 В открывшемся командном окне на панели инструментов нажимаем кнопку Simulink.

3 Открывается Simulink. Кроме рабочего окна с общим именем Untitled (“Безымянный”) открывается окно библиотеки Simulink с разделами:

- ***Sourses*** – источники;
- ***Sinks*** – приемники;
- ***Discrete*** – дискретные;
- ***Linear*** – линейные;
- ***Nonlinear*** – нелинейные
- ***Connections*** – связи;
- и т.д.

Порядок выполнения работы:

В качестве примера рассмотрим демонстративный пример. Для этого:

1. Нажимаем кнопку ***Demos*** в меню ***Help***.
2. В левом окошке выбираем пункт ***Simulink***.
3. Двойным щелчком выбираем пункт ***Simple models*** (в версии 6.1 и выше пункт ***General***).
4. В правом окне выбираем модель ***Spring-mass system simulation***.
5. Запускаем ее командой ***Run*** или двойным щелчком.

Окно демонстрации “Spring–mass system simulation” представлено на рисунке 1.1.

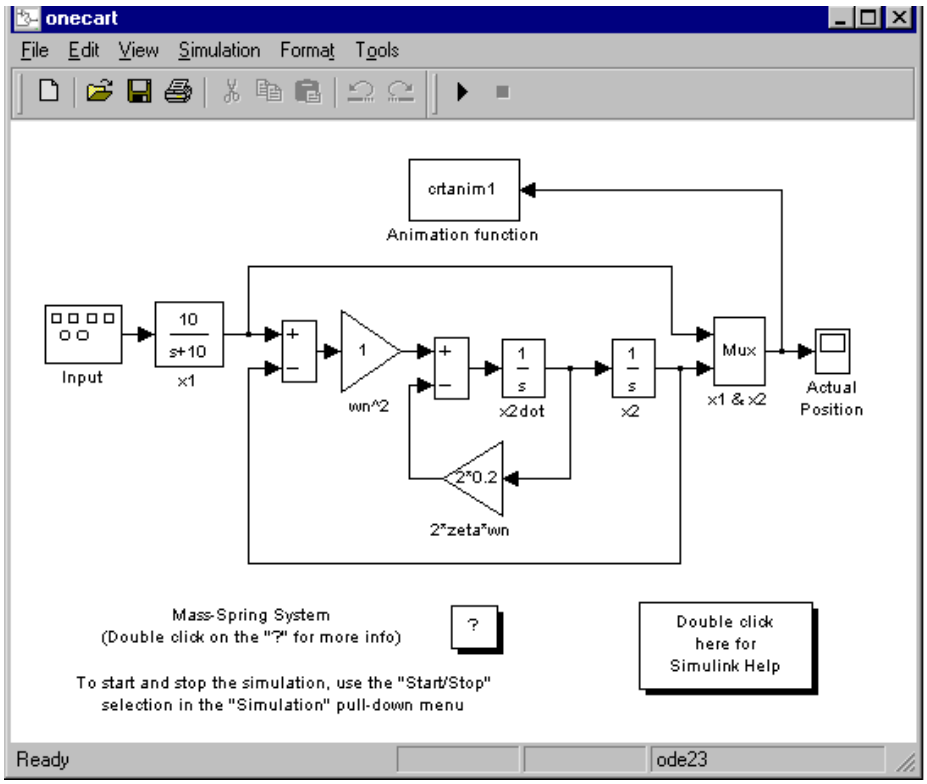


Рисунок 1.1 - Окно Демонстрации “Spring–mass system simulation”

Данная модель реализует колебательную механическую систему с одной степенью свободы. Физическим аналогом модели является переменное по направлению движение, закрепленной пружиной к “прыгающей” заделке (слева) массы по гладкой поверхности с трением. Внешняя сила приложена к заделке.

Модель реализует дифференциальное уравнение, описывающее движение кубика. В MATLAB в передаточных функциях вместо p–оператора Лапласа пишется s.

Рассмотрим блоки (слева направо):

Input – генератор, ***x1*** – звено первого порядка, сумматор, усилитель, $\frac{1}{s}$ – интегратор, ***Mux*** - смеситель (позволяет вывести более одного сигнала на ***Scope***), ***Scope (Actual position)*** – осциллограф, ***Animation function*** - блок анимации.

Запуск моделирования осуществляется кнопкой .

На экране – анимационная картинка, представленная на рисунке 1.2:

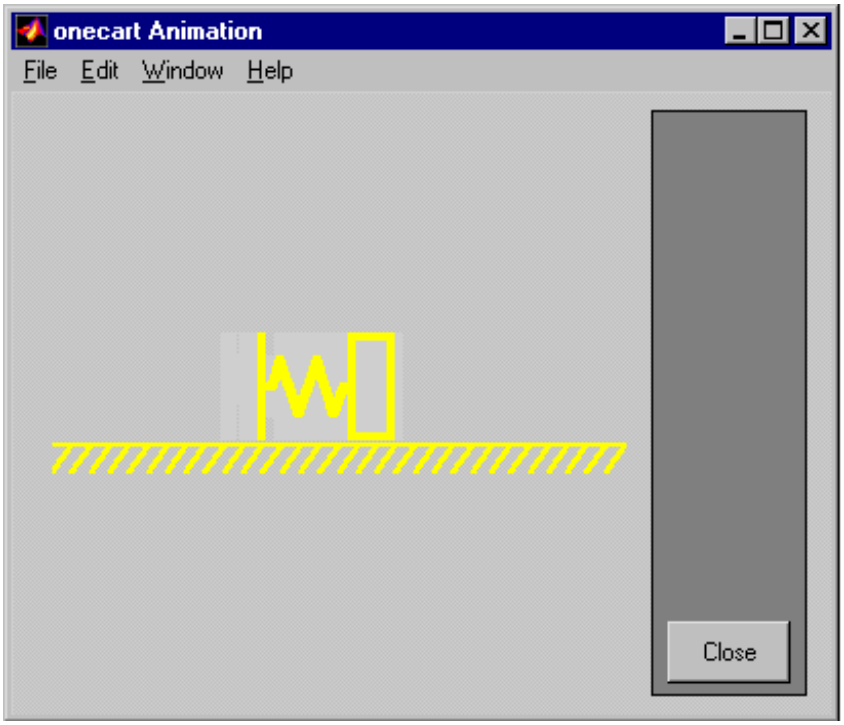


Рисунок 1.2 - Движение груза на пружине

Двойной щелчок на ***Scope*** после запуска моделирования открывает окно ***Scope*** (рисунок 1.3), в котором отображаются графики, характеризующие колебания правой массы во времени и знакопеременную внешнюю силу.

Остановить моделирование с помощью значка и закрыть окна демонстрации.

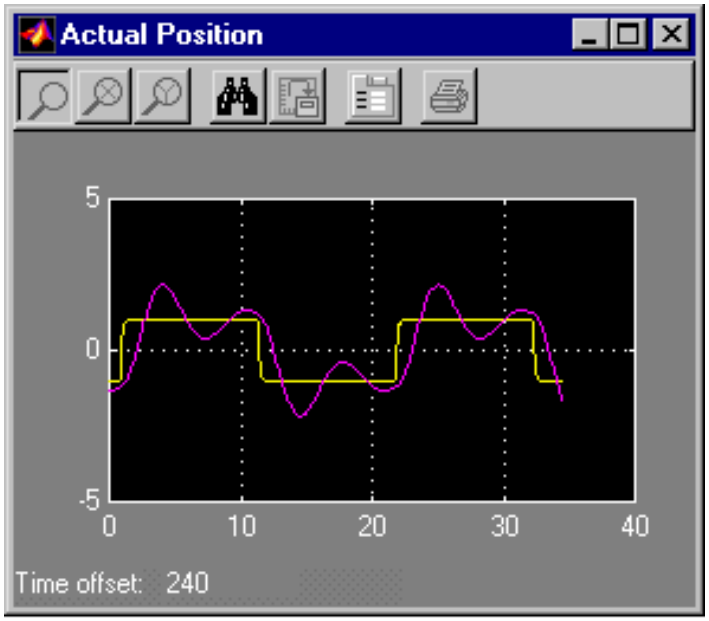


Рисунок 1.3 - Графики, характеризующие работу системы

Ход работы:

1. На экране – открыть окно библиотеки и рабочее окно Simulink.
2. В окне библиотеки двойным щелчком раскрывается каждый из разделов. В разделе библиотеки ***Sources*** (Источники) выбираем генератор ***Signal Generator***, перетаскиваем его в рабочее окно и закрываем окно ***Sources***.
3. В разделе ***Sinks*** **(**Приемники) выбираем осциллограф ***Scope***, перетаскиваем в рабочее окно и закрываем окно ***Sinks***.
4. В разделе ***Linear*** (Линейные блоки) или ***Math*** (для версии 6.1 и выше) выбираем блок усилитель – ***Gain*** с регулируемым коэффициентом усиления (КУ) и сумматор – ***Sum***, а также интегратор – ***Integretor*** (в версии 6.1 и выше данный блок находится в разделе ***Continuous***). Все блоки последовательно перетаскиваем в рабочее окно и закрываем окно ***Linear*** (***Math***).
5. В разделе ***Connections*** (Связи) или ***Signals & Systems*** (для версии 6.1 и выше) выбираем смеситель – ***Mux*** и перетаскиваем в рабочее окно.
6. В рабочем окне приступаем к соединению блоков. Схема модели должна иметь вид (рисунок 1.4):

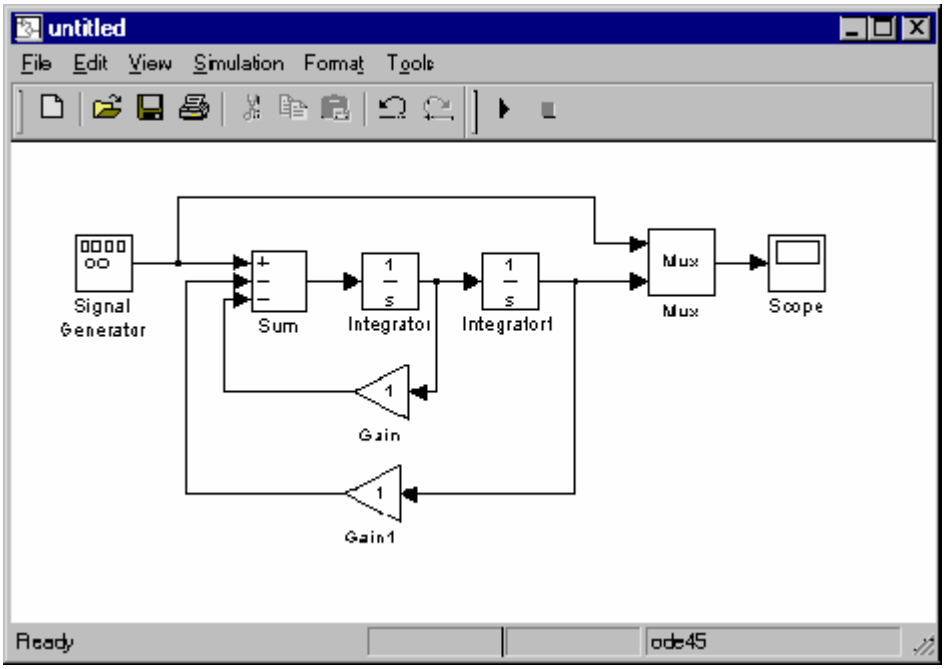


Рисунок 1.4 - Схема модели лабораторной работы

Копирование блока осуществляется путем перетаскивания с нажатой клавишей Ctrl. Поворот блока осуществляется путем выделения блока и нажатием комбинации клавиш Ctrl+F.

7. Перейдем к настройке системы. Открытие окна настройки блока осуществляется путем двойного щелчка. В блоке ***Signal Generator*** рассмотрим доступные нам сигналы. Выбираем ***Squre*** – частоту в Гц – 0.02. Смотрим, как система отреагирует на скачкообразное возмущение. Система не является колебательной.

8. Уменьшим КУ по внутреннему контуру (**Gain**). $КУ_{Gain}$=0.1, т.е. уменьшим демпфирующую силу в 10 раз. Моделируем. Имеем колебательную систему. Теперь нужно поменять *собственную частоту,* которую определяет нижний $КУ_{Gain1}$. Увеличим его в 10 раз. Видим, что увеличилась частота и уменьшилась амплитуда колебаний.

9. В ***Signal Generator*** увеличим амплитуду в 5 раз. На внутреннем ***Gain*** вместо 0.1 сделаем 0.3 – увеличим коэффициент демпфирования.

Мы убедились, что частота колебаний регулируется усилителем на внешнем контуре, а затухание колебаний регулируется на внутреннем контуре.

10. Ознакомиться с демонстрационными примерами ***Tracking a bouncing boll***, ***Simple pendulum simulation***, ***Toilet bowl flushing animation*** и описать один из них по указанию преподавателя:

- назначение системы;
- состав модели;
- особенности движения системы.

2 Моделирование нелинейных и дискретных систем

Порядок составления структурной модели в Simulink для системы, описываемой дифференциальным уравнением

1. Разрешить дифференциальное уравнение относительно старшей производной.
2. С помощью сумматора Sum собрать модель правой части уравнения.
3. Используя цепочки из последовательно включенных интеграторов Int (количество которых определяется порядком уравнения), получить неизвестную функцию и ее производные.
4. С помощью усилителей Gain с регулируемым коэффициентом передачи или соответствующих нелинейных блоков получить слагаемые правой части уравнения (полученного в п.1) и подключить их к сумматору Sum.
5. Внешнюю силу F(t) моделировать соответствующим генератором.
6. В полученной схеме установить коэффициенты в соответствии с заданным вариантом.
7. Проверить модель на структурную устойчивость: число перемен знака сигнала в каждом замкнутом контуре должно быть нечетным.
8. Набрать модель в рабочем окне Simulink.
9. Добавить в модель необходимые приемники сигналов Scope, Display, X-Y plot.

Уравнение Ван дер Поля

Классическое нелинейное дифференциальное уравнение Ван дер Поля обычно записывается в форме:

$$\frac{d^2x}{dt^2} - \varepsilon \cdot \left(1 - x^2\right) \cdot \frac{dx}{dt} + x = 0. \qquad (2.1)$$

Это уравнение описывает колебательную систему с переменным коэффициентом демпфирования. Если перемещения

малы, то коэффициент при $\frac{dx}{dt}$ отрицателен, что соответствует отрицательному демпфированию и, следовательно, развитию автоколебаний. При больших перемещениях демпфирование становится положительным, то есть устойчивые автоколебания могут существовать только при отсутствии демпфирования. Уравнение достаточно хорошо описывает работу некоторых генераторов колебаний. Качественный характер решения зависит от значения параметра ε. При малых сравнительно с единицей значениях ε получаются решения одного вида, тогда как при больших значениях ε решения имеют другой вид.

Задание: промоделировать уравнение (2.1) при значениях $\varepsilon = 1,\ x(0) = 2,\ \dot{x}(0) = 0$.

Уравнение Рэлея

Уравнение Рэлея эквивалентно уравнению Ван дер Поля (2.1),

$$\frac{d^2y}{dt^2} - \varepsilon \cdot \left[\frac{dy}{dt} - \frac{1}{3} \cdot \left(\frac{dy}{dt} \right)^3 \right] + y = 0 \qquad (2.2)$$

но проще, так как нелинейное слагаемое содержит лишь производную $\frac{dy}{dt}$, а не две переменные x и $\frac{dx}{dt}$.

Задание: промоделировать уравнение (2.2) при значениях $\varepsilon = 0.2,\ y(0) = 1,\ \dot{y}(0) = 0$.

Уравнение Матье

$$\frac{d^2y}{dt^2} + (a - 2 \cdot q \cdot \cos 2 \cdot t) \cdot y = 0. \qquad (2.3)$$

Уравнение Матье представляет собой уравнение второго порядка с периодическим коэффициентом, где a и q - постоянные.

Задание: промоделировать уравнение (2.3) при значениях

$$a = 1,\ q = 0.02,\ y(0) = 1,\ \dot{y}(0) = 0.$$

Линейные уравнения второго порядка с переменными коэффициентами

Уравнение Лежандра:

$$\left(1 - t^2\right) \cdot \frac{d^2 y}{dt^2} - 2 \cdot t \cdot \frac{dy}{dt} + n \cdot (n+1) \cdot y = 0. \qquad (2.4)$$

Задание: промоделировать уравнение (2.4) при значениях $n = 3,\ y(0) = 3,\ \dot{y}(0) = 1.$

Модель нелинейной электрической цепи

$$L \cdot \left(1 - a \cdot i^2\right) \cdot \frac{di}{dt} + R \cdot i = E \cdot \sin \omega \cdot t. \qquad (2.5)$$

Данное уравнение приближенно описывает цепь, содержащую индуктивность с насыщающимся стальным сердечником – нелинейный элемент. Слагаемое, содержащее $i^2 \cdot \frac{di}{dt}$, - нелинейный член уравнения.

Задание: промоделировать уравнение (2.5) при значениях $\omega = 10,\ E = 10,\ R = 1,\ a = 0.1,\ L = 1,\ i(0) = 0.7.$

Резонансный контур с переменной емкостью

$$\frac{d^2 q}{dt^2} + \omega_0^2 \cdot (1 - m \cdot \cos \omega_1 \cdot t) \cdot q = 0. \qquad (2.6)$$

Используя данное уравнение, определяют мгновенное значение заряда конденсатора q входящего в состав электрического контура с постоянной индуктивностью L и переменной емкостью, мгновенное значение которой определяется как $C = C_0 \cdot (1 + m \cdot \cos \omega_1 \cdot t)$.

C_0 - среднее значение; m - относительное изменение ($m << 1$); ω_1 - круговая частота изменений емкости ($\omega_1 << \omega_0$); $\omega_0^2 = \frac{1}{L \cdot C_0}$.

Задание: промоделировать уравнение (2.6) при значениях $m = 0.0001,\ \omega_1 = 0.005,\ \omega_0 = 10,\ q(0) = 10,\ \dot{q}(0) = 0$.

Модель нелинейной системы

$$\frac{d^2x}{dt^2} + 25 \cdot \left(1 + 0.1 \cdot x^2\right) \cdot x = 0. \qquad (2.7)$$

С помощью данного уравнения описывают движение постоянной массы, прикрепленной к нелинейной пружине, жесткость которой растёт по мере возрастания деформации.

Задание: промоделировать уравнение (2.7) при значениях $x(0) = 3,\ \dot{x}(0) = 0$.

Уравнения Вольтерра

Данные уравнения описывают борьбу между двумя видами животных:

$$\left.\begin{aligned} \frac{dN_1}{dt} &= k_1 \cdot N_1 - k_3 \cdot N_1 \cdot N_2 \\ \frac{dN_2}{dt} &= k_2 \cdot N_2 - k_4 \cdot N_1 \cdot N_2 \end{aligned}\right\}. \qquad (2.8)$$

В этих уравнениях N_1 и N_2 - численности двух видов, живущих на какой-либо определённой территории. Эти виды находятся в конкурентной борьбе друг с другом, хотя бы потому, что они потребляют один и тот же корм. k_1, k_2, k_3 и k_4 - положительные действительные параметры, причём $k_2 > k_1$. Первый член правой части соответствует нормальному увеличению численности, которое имело бы место в отсутствии ограничивающих факторов. Второй член выражает уменьшение численности, которое возникает в результате конкурентной борьбы двух видов животных.

Задание: промоделировать систему (2.8) при значениях $k_1 = 2$, $k_2 = 8$, $k_3 = 1$, $k_4 = 4$, $N_1(0) = 10$, $N_2(0) = 10$.

Бомбометание с малых высот с учетом сопротивления воздуха

При движении снаряда с умеренной скоростью сила сопротивления воздуха примерно пропорциональна квадрату этой скорости и направлена ей навстречу. Движение снаряда массой M описывается уравнениями:

$$\left.\begin{aligned} \dot{\theta} &= \frac{g}{\upsilon} \cdot \cos\theta \\ \dot{\upsilon} &= g \cdot \sin\theta - h \cdot \upsilon^2 \end{aligned}\right\}, \qquad (2.9)$$

где k - постоянная, определяющая влияние сопротивления воздуха; $h = k/M$; υ - скорость движения по траектории; θ - угол, под которым направлен вектор скорости, отсчитываемый вниз от горизонтальной оси.

Задание: промоделировать уравнение (2.9) при значениях $g = 9.8$, $h = 2$, $\theta(0) = 0$, $\upsilon(0) = 96$.

Модель логической системы

Составить модель “Выбор”, моделирующую такую ситуацию: “Пусть у больного поднялась температура 39° C. Не зная принять

ли ему лекарство, больной решает: “Если температура поднимется выше, то выпью лекарство, если снизится – не буду принимать. А если состояние ухудшится – тогда выпью комбинированное лекарство”. В любом случае больной должен сообщить о своем состоянии врачу”. Выбор больного должен имитировать блок *Combinatorial logic*, на вход которого должны поступать объединенные сигналы с выходов блоков *Constant*, имитирующих повышение температуры у больного (например “1” – температура повысилась, “0” - нет). Результаты, полученные в ходе очередного сеанса моделирования, сохранить в рабочей области MATLAB. Теперь предположим, что больной принимает решение не сразу выпить одно или другое лекарство, или оба вместе, а через некоторое время, решает принять каждое по отдельности и уже потом звонит врачу. Для такой ситуации создать модель “Решение”, логика работы которой состоит в следующем: 1. Выходной сигнал, сформированный в модели “Выбор”, должен быть считан из рабочей области MATLAB и разделен на три компоненты (*Решение первое лекарство*, *Решение второе лекарство и Сообщение врачу*); 2. Значение первых двух компонентов выводятся одновременно в окнах двух блоков Display (каждое в своем); 3. На следующем шаге работы модели в третьем окне Display выводится значение компоненты *Сообщение родителям*.

Примечание: в обоих моделях в разделе ***Simulation-Simulation parameters*** на вкладке ***Solver*** установить: Stop time=1, Type=Fixed-step (discrete), Fixed step size=auto для модели «Выбор» и 0.2 для модели «Выбор».

Моделирование случайных событий

Создать модели: 1. которая позволяет подсчитать число попаданий некоторого признака в заданный интервал, то есть:

- генерирует последовательность случайных чисел (СЧ), равномерно распределенных на интервале [0;7],
- подсчитывает число попаданий СЧ в диапазон [0;5],
- выводит результат подсчета оператору.

2. которая обеспечивает регистрацию момента превышения случайным числом заданного значения:

- генерирует последовательность случайных чисел (СЧ), равномерно распределенных на интервале [0;7],
- подсчитывает количество моментов превышения случайным числом заданного значения =4,
- выводит результат подсчета оператору.

3 Генерация случайных процессов с заданной спектральной плотностью

Основные определения

Случайный сигнал – сигнал, мгновенное значение которого заранее не известно и может быть предсказано лишь с некоторой вероятностью, меньшей единицы.

Случайный процесс (СП) – математическая модель изменяющегося во времени случайного сигнала.

Случайный процесс X(t) – функция, значения которой в любой момент времени t, являются случайными величинами.

СП X(t) стационарен в широком смысле, если его математическое ожидание M_x и дисперсия D_x не зависят от времени, а корреляционная функция R_x зависит не от самих моментов времени, а только от интервала между ними $\varphi=t_1-t_2$:

$$R_x(t_1,t_2)=R_x(t_2-t_1)=R_x(\varphi),$$
$$M_x(t)=\text{const}, D_x(t)=\text{const}.$$

Спетральная плотность – величина, которая пропорциональна средней мощности процесса в интервале частот от ω до $\omega+d\omega$.

Белый шум – стационарный СП, спектральная плотность мощности которого постоянна на всех частотах: $W(\omega)=W_0=\text{const}$.

Теоретическая часть

Многие сигналы являются детерминированными, то есть их временная зависимость y(t) известна и не меняется при неоднократном моделировании. Такие сигналы удобны для аналитического описания и анализа систем. Но на практике большинство реальных сигналов не являются детерминированными. Одной из причин этого является засоренность сигналов шумами и помехами, имеющими случайный характер с тем или иным спектром. Таким образом, для получения

реального сигнала, используемого на практике, необходимо уметь генерировать помехи и прежде всего белый шум.

Предположим, что в нашем распоряжении имеется генератор шума, выходной сигнал которого имеет спектральную плотность $S_x(\omega)$. Для имитации шумов имеющих место в реальной системе, требуется воспроизвести случайный сигнал со спектральной плотностью $S_y(\omega)$ (рисунок 3.1). Для этого необходимо определить передаточную функцию $\Phi(j\omega)$ линейного формирующего фильтра.

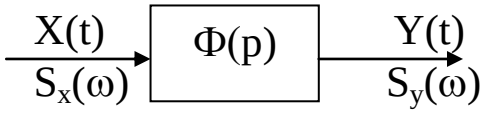


Рисунок 3.1 - Линейная система

Из теории автоматического управления известно, что спектральная плотность сигнала на выходе линейной системы равна квадрату модуля частотной функции, умноженного на спектральную плотность входного сигнала:

$$S_y(\omega) = |\Phi(j \cdot \omega)|^2 \cdot S_x(\omega).$$

На основании этого можно найти:

$$|\Phi(j\omega)|^2 = \frac{S_y(\omega)}{S_x(\omega)}.$$

Если в качестве входного сигнала используется белый шум, для которого $S_x(\omega)=1$, то

$$|\Phi(j\omega)|^2 = S_y(\omega).$$

То есть спектральная плотность сигнала на выходе в этом случае зависит только от параметров линейного фильтра.

В данной работе на вход фильтра кроме белого шума необходимо также подавать периодический сигнал (синусоиду), так как в реальных системах кроме помех, имеющих случайный характер, могут присутствовать помехи неслучайного характера.

Как правило, в качестве линейного формирующего фильтра используется колебательное звено:

$$W_{ф}(p) = \frac{1}{T^2 \cdot p^2 + 2 \cdot \xi \cdot T \cdot p + 1}, \qquad (3.1)$$

где $T = \frac{1}{\omega_0}$, ω_0 – резонансная частота.

При создании активных узкополосных фильтров с передаточной функцией вида (3.1) применяют избирательные усилители, которые из поступающей на вход совокупности сигналов широкого диапазона частот выделяют группу сигналов близких частот, несущих полезную информацию.

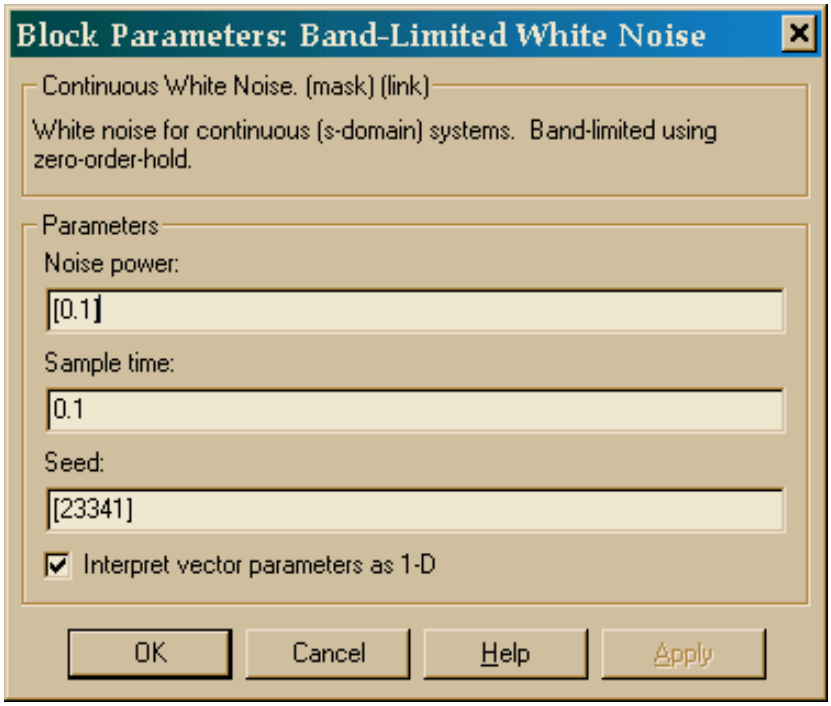
Порядок выполнения работы

1. В пакете Simulink составить и набрать структурную модель, позволяющую генерировать случайный процесс (белый шум), содержащий периодическую составляющую с частотой ω_1. Данная модель должна представлять собой фильтр (колебательное звено с передаточной функцией (4.1)), на вход которого подается суммарный сигнал белого шума и синусоиды $f(t)=A \cdot \sin(\omega_1 \cdot t)$.
2. Изменяя коэффициент демпфирования о (в интервале 0.3÷0.7 методом деления пополам) в передаточной функции фильтра получить на его выходе случайный сигнал с заданной спектральной плотностью.

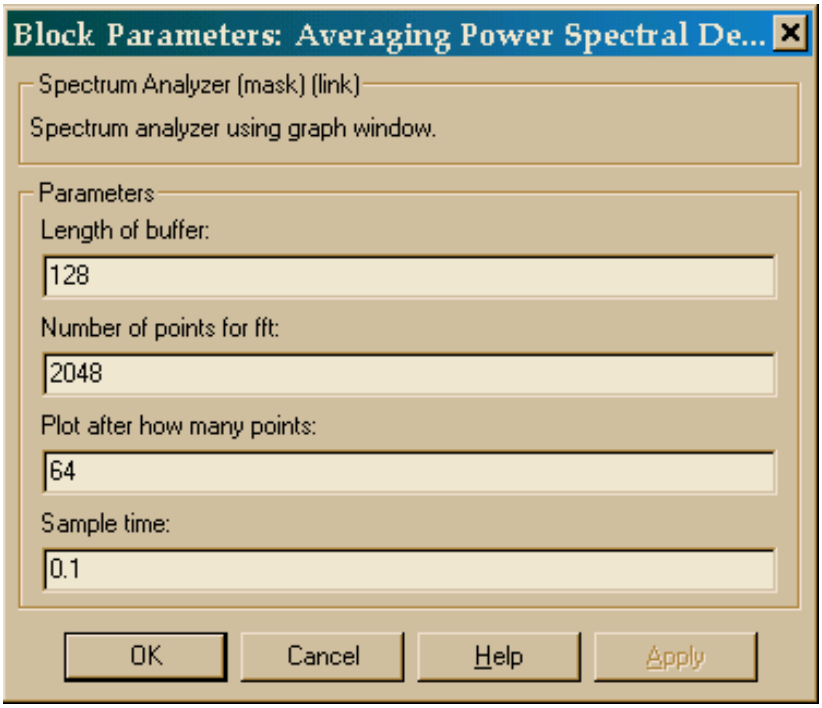
При составлении модели необходимы будут блоки:

- Band Limited White Nose – генератор белого шума (рисунок 3.2, А);
- Блок Averaging Power Spectral Density из библиотеки Simulink Extras – предназначен для наглядного представления формы выходного сигнала и его энергетического спектра;

необходимо использовать следующие настройки данных блоков (рисунке 3.2, Б):



А



Б

Рисунок 3.2 – Блоки, необходимые для составления модели в Simulink

Метод основан на делении текущего отрезка [a, b] на две равные части с последующим выбором одной из половин, в которой находится максимум в качестве следующего текущего отрезка. Чем ближе коэффициент демпфирования о к 0.3, тем более выражена резонансная частота в спектральной плотности выходного сигнала и наоборот.

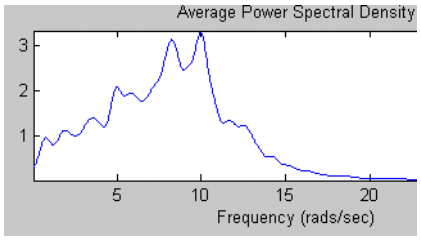
В таблице 3.1 представлены характеристики, необходимые для построения случайный сигнал с заданной спектральной плотностью.

Таблица 3.1 – Индивидуальные задания

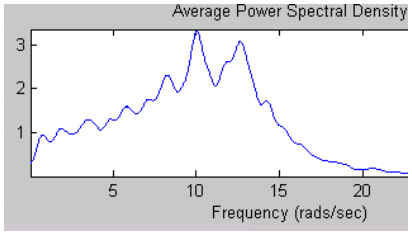
№	A	ω_1	ω_0	Stop time	$S_y(\omega)$
1	0.1	5	10	400	рисунок 4.3, 1
2	0.1	7	12.5	400	рисунок 4.3,.2
3	0.1	3	4.5	200	рисунок 4.3, 3
4	0.1	4.5	6.7	250	рисунок 4.3, 4
5	0.1	3	5	200	рисунок 4.3, 5
6	0.1	4	8	400	рисунок 4.3, 6
7	0.1	6	11.1	300	рисунок 4.3, 7
8	0.1	2	3.84	250	рисунок 4.3, 8
9	0.1	4	7.69	400	рисунок 4.3, 9
10	0.01	2	4	400	рисунок 4.3, 10
11	0.04	5	9	300	рисунок 4.3, 11
12	0.01	2.5	5	400	рисунок 4.3, 12

На рисунке 3.3 представлены случайные сигналы с заданной спектральной плотностью.

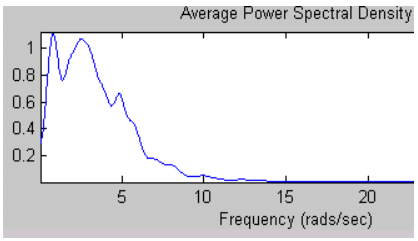
Stop time – время моделирования (Simulation → Simulation parameters → Stop time).



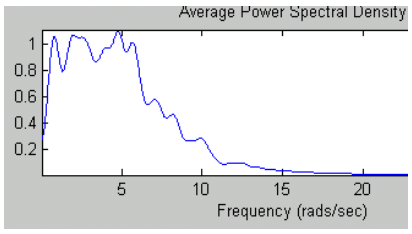
1



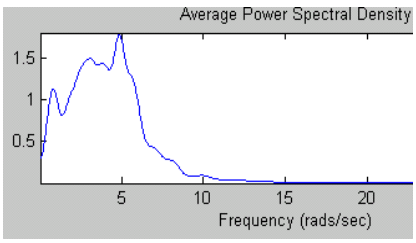
2



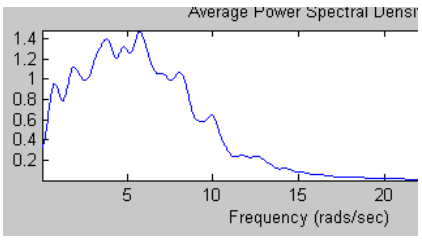
3



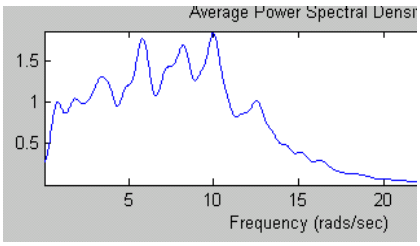
4



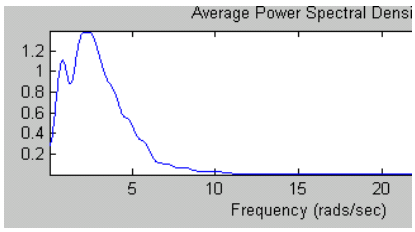
5



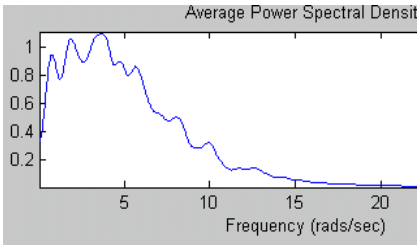
6



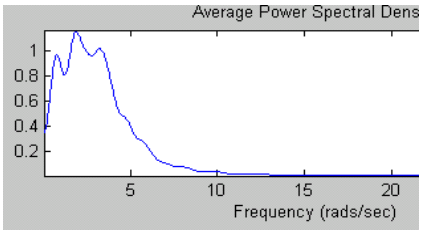
7



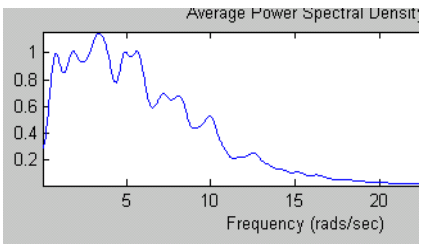
8



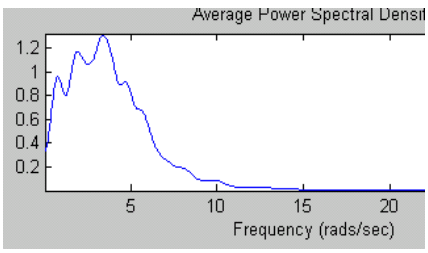
9



10



11



12

Рисунок 3.3 - Случайные сигналы с заданной спектральной плотностью

4 Идентификация динамических объектов по переходным функциям

Теоретические сведения

Идентификация (identificare, *лат.*- узнавать) – оценивание структуры и параметров системы по результатам экспериментов с ней. Наиболее информативна идентификация системы по результатам *активных экспериментов* (с вмешательством в работу объекта с помощью пробных сигналов). Наиболее удобный пробный сигнал – скачкообразное изменение управляющего сигнала или возмущения, действующего на систему.

Металлургические машины и агрегаты, как правило, работают в непрерывно-дискретном режиме, поэтому вместо подачи пробных сигналов часто можно использовать скачкообразные технологические возмущения (например, загрузка очередной порции шихты в доменную печь, начало продувки в конверторе, захват заготовки прокатными валками и др.).

Зарегистрированный график изменения выходной величины при скачкообразном возмущении на входе – *переходную функцию* технологического объекта управления (ТОУ) можно использовать для его идентификации.

Если поставлена задача автоматизации ТОУ, то идентификацию следует проводить средствами теории автоматического управления (ТАУ), т.е. представить объект в виде некоторого соединения типовых звеньев ТАУ, линейных и нелинейных.

Получив график переходной функции, вначале выполняют *структурную идентификацию* – оценивание порядка передаточной функции объекта, вида его нелинейностей. При этом основываются на особенностях графиков переходных функций.

Далее выполняют *параметрическую идентификацию* – оценивание параметров системы (коэффициентов передаточной функции либо уравнений нелинейных звеньев) приближенными численными методами ТАУ.

Графически можно определить постоянную времени инерционного звена (как это показано в таблице) или постоянную времени колебательного звена:

$$T=T_к/2\pi,$$

где $T_к$ – период собственных колебаний, а коэффициент демпфирования колебаний определяют по графику в зависимости от величины относительного перерегулирования (рисунок 4.1)

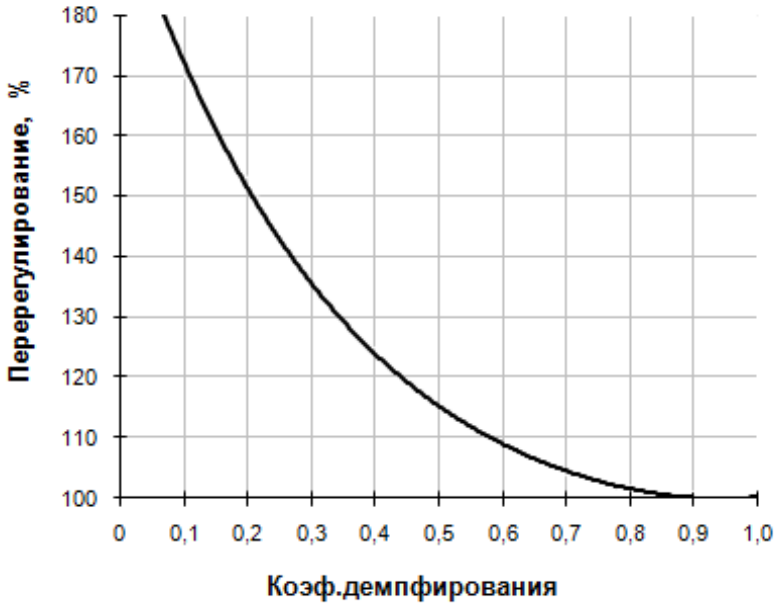


Рисунок 4.1 – График зависимости от величины относительного перерегулирования

Идентификация с помощью настраиваемой модели

Более точно выполнить параметрическую идентификацию можно на ПЭВМ, например, с помощью системы структурного моделирования Simulink.

Порядок проведения работы

1. По результатам структурной идентификации собрать из блоков Simulink модель в виде соединения типовых звеньев. На

вход модели подать скачкообразный входной сигнал с генератора **Step.**

2. По результатам предварительной численной параметрической идентификации задать предполагаемые значения параметров системы.

3. Заданную переходную функцию в виде МА№ . mat – файла ввести в модель с помощью генератора **From File.**

4. Собрать схему для качественного сравнения заданной переходной функции и выходного сигнала системы на осциллографе **Scope.**

5. Собрать схему для количественного сравнения этих сигналов по минимуму среднеквадратичной ошибки или средней ошибки модуля.

6. Подобрать оптимальные значения параметров модели, пользуясь, например, методом покоординатного спуска.

Записать полученный численный вид передаточной функции.

5 Моделирование объектов с распределенными параметрами

Краткие теоретические сведения

Дифференциальное уравнение теплопроводности для одной пространственной координаты будет иметь вид

$$\frac{\partial u}{\partial t} = a\left(\frac{\partial^2 u}{\partial x^2}\right) + \frac{q_v}{c \cdot \rho}, \qquad (5.1)$$

где $\frac{\partial u}{\partial t}$ – частная производная температуры по времени; $\frac{\partial^2 u}{\partial x^2}$ – частная производная второго порядка температуры по координате; a – *коэффициент температуропроводности*; q_v – *тепловой поток* внутренних источников тепла [Вт/м²]; ρ – *плотность* [кг/м³]; ***c*** – *теплоемкость* [Дж/кг·град].

В терминах векторного анализа уравнение (5.1) будет иметь вид

$$\frac{\partial u}{\partial t} = a \cdot \Delta u + \frac{q_v}{c \cdot \rho} \qquad (5.2а)$$

$$\frac{\partial u}{\partial t} = a \cdot \nabla^2 u + \frac{q_v}{c \cdot \rho} \qquad (5.2б)$$

$$\frac{\partial u}{\partial t} = \mathrm{div}\left(a \cdot \mathrm{grad}(u)\right) + \frac{q_v}{c \cdot \rho} \qquad (5.2с)$$

в этих уравнениях Δu – оператор Лапласа; $\nabla^2 u$ – оператор Гамильтона; $\mathrm{div}(\mathrm{grad}(u)) = \Delta u = \nabla^2 u$ – дивергенция *градиента температуры*;

Градиент температуры – это вектор, направленный по нормали к изотермической поверхности в сторону возрастания температуры и численной равный производной от температуры по этому направлению.

Коэффициент температуропроводности характеризует скорость изменения температуры. Коэффициент температуропроводности является мерой теплоинерционных свойств тела. Из уравнения (5.1) следует, что изменение температуры во времени $\frac{\partial u}{\partial t}$ для любой точки пространства пропорционально величине a. Скорость изменения температуры в любой точке тела будет тем больше, чем больше коэффициент температуропроводности. Поэтому при прочих равных условиях выравнивание температур во всех точках пространства будет происходить быстрее в том теле, которое обладает большим коэффициентом температуропроводности. Величина коэффициента температуропроводности зависит от природы вещества. Коэффициент температуропроводности можно вычислить по формуле

$$\alpha = \frac{\lambda}{\rho \cdot c}, [м^2/с] \qquad (5.3)$$

λ – коэффициент теплопроводности [Вт/м·град].

Подставляя выражение (5.3) уравнение (5.1) получим

$$c \cdot \rho \cdot \frac{\partial u}{\partial t} = \lambda \cdot \left(\frac{\partial^2 u}{\partial x^2} \right) + q_v \qquad (5.4)$$

Коэффициент теплопроводности характеризует способность тел проводить тепло. Чем больше коэффициент, тем быстрее прогревается тело.

Различают *стационарный* и *нестационарный* процессы типы передачи. При установившемся (*стационарном* режиме) температура тела не зависит от времени и поэтому левая часть дифференциального уравнения теплопроводности равна нулю. При нестационарном режиме температура тела зависит от времени. В случае стационарного режима используется для решения *эллиптическое* уравнение, а в случае нестационарного режима

используется *параболическое* уравнение. В случае, если в теле нет внутренних источников тепла, выражение $q_v = 0$.

Для численного решения задачи распространения тепла в пластине необходимо задать *начальные* условия и *граничные* (*краевые*) условия.

Начальные условия устанавливают распределение температуры в теле в определенный момент времени, чаще всего в момент времени $t = 0$.

Краевые условия характеризуют тепловые условия на поверхности тела, которые должны быть известны в любой момент времени.

Различают следующие виды граничных (краевых) условий.

- *Граничные условия* первого рода. При этом задается распределение температуры на поверхности тела для каждого момента времени. По другому их называют еще *условиями Дирихле*.
- *Граничные условия* второго рода. При этом задаются величины теплового потока для каждой точки поверхности тела и любого момента времени. По другому их называют еще *условиями Неймана*. Тепловой поток пропорционален $\frac{du}{dn}$.
- *Граничные условия* третьего рода. При этом задается температура окружающей среды и закон теплообмена между поверхностью и окружающей средой. Граничное условие третьего рода характеризует закон теплообмена между поверхностью и окружающей средой в процессе охлаждения и нагревания тела.
- *Граничные условия* четвертого рода характеризуют условия теплообмена системы тел или тела с окружающей средой по закону теплопроводности.

Решение уравнений теплопроводности в системе MATLAB

В системе MATLAB дифференциальные уравнения в частных производных (PDE) могут быть решены методом конечных элементов двумя способами:

1. Путем составления программы на языке MATLAB;
2. При помощи специальной панели инструментов – графического интерфейса пользователя (GUI) PDE Toolbox.

В данной работе будет рассматриваться второй способ. Он позволяет решать поставленные задачи с граничными условиями *Дирихле* и *Неймана*.

Для запуска PDE Toolbox в командной строке окна управления нужно выполнить команду pdetool. После чего откроется окно, в котором находятся инструменты и рабочая область, в которой строится и отображается численное решение уравнений.

Пример

Решить задачу нагрева пластины толщиной 2 см., высотой 1,6 см. Посредине пластины имеется прямоугольное отверстие шириной 0,1 см. и высотой 0,8 см. Продолжительность нагрева 5 секунд.

Граничные условия:

- u = 100 – на левой границе;
- u = -10 – на правой границе;
- на остальных границах $\frac{du}{dn} = 0$.

Начальное условие $u(0)=20$. Определить распределение температур по длине пластинки.

Решение

1. Необходимо нарисовать объект в среде PDE Toolbox. Для более точного построения необходимо перейти в режим сетки. Для этого необходимо зайти в меню Options и активизировать пункты Grid (включает и отключает режим сетки) и Snap (привязка к узлам сетки) (рисунок 5.1). Пункт Axes limits устанавливает границы координатных осей, а Axes Equal выравнивает масштабы осей. Для построения отверстия необходимо зайти в подменю Grid spacing и нанести вспомогательные линии сетки. Для изменения координатной сетки необходимо отключить режим Auto.

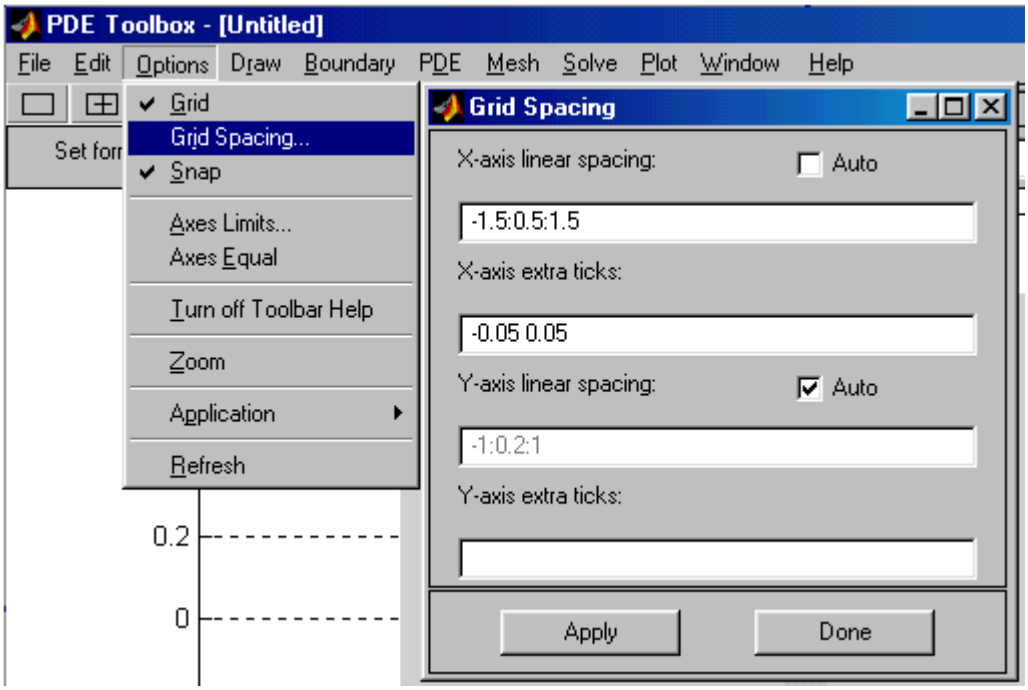


Рисунок 5.1 - Задание параметров осей в PDE Toolbox

2. После этого производится построение двух областей, как показано на рисунке 5.2. В строке формул исправляется R1+R2 на R1-R2.

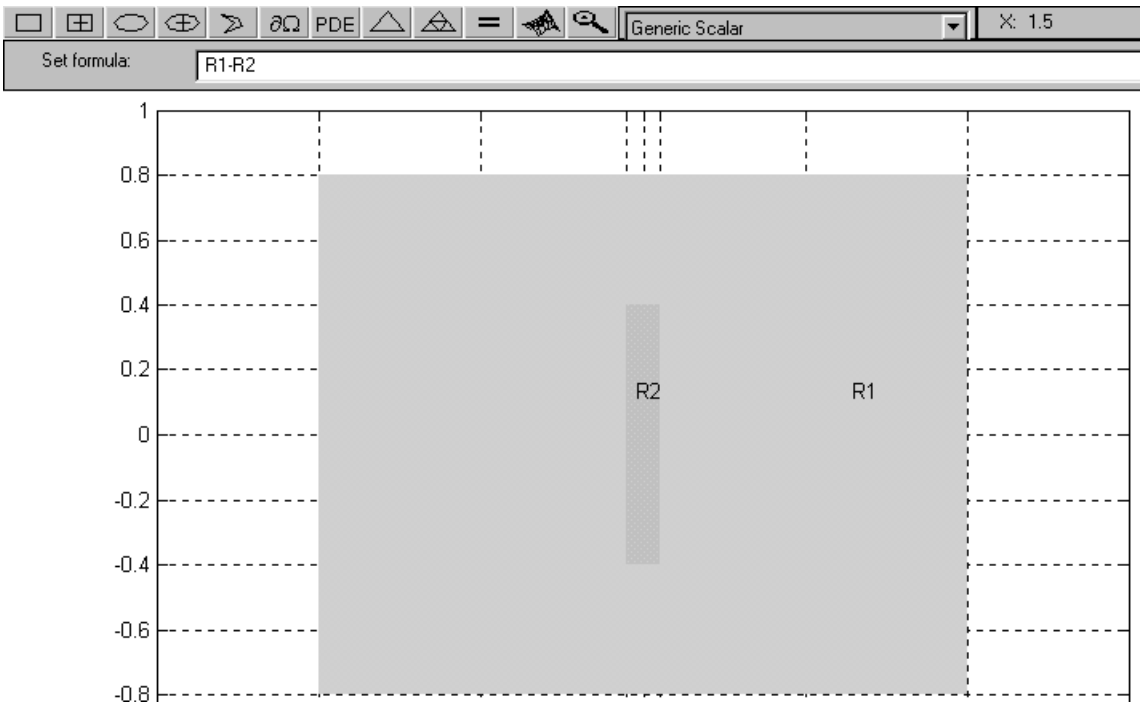


Рисунок 5.2 - Вид пластины

3. В PDE Toolbox можно задавать граничные условия *Дирихле* и *Неймана*. Зададим сначала граничные условия Неймана, которые задаются следующим образом:

• Нажимаем кнопку после того как границы пластины будут отображаться в виде стрелок черного цвета необходимо нажать комбинацию клавиш Ctrl+A для выделения всего периметра. Равенство нулю теплового потока есть частный случай условий *Неймана*. Необходимо зайти в меню Boundary и выбрать пункт Specify boundary conditions. В открывшемся диалоговом окне указать тип – условия *Неймана* (рисунок 5.3). Для получения теплоизолированных верхних и нижних границ и таких же границ пластины с отверстием необходимо, чтобы коэффициенты g (тепловой поток) и q (*коэффициент теплоотдачи*) были равны нулю. После этого стрелки по периметру изменят цвет на синий.

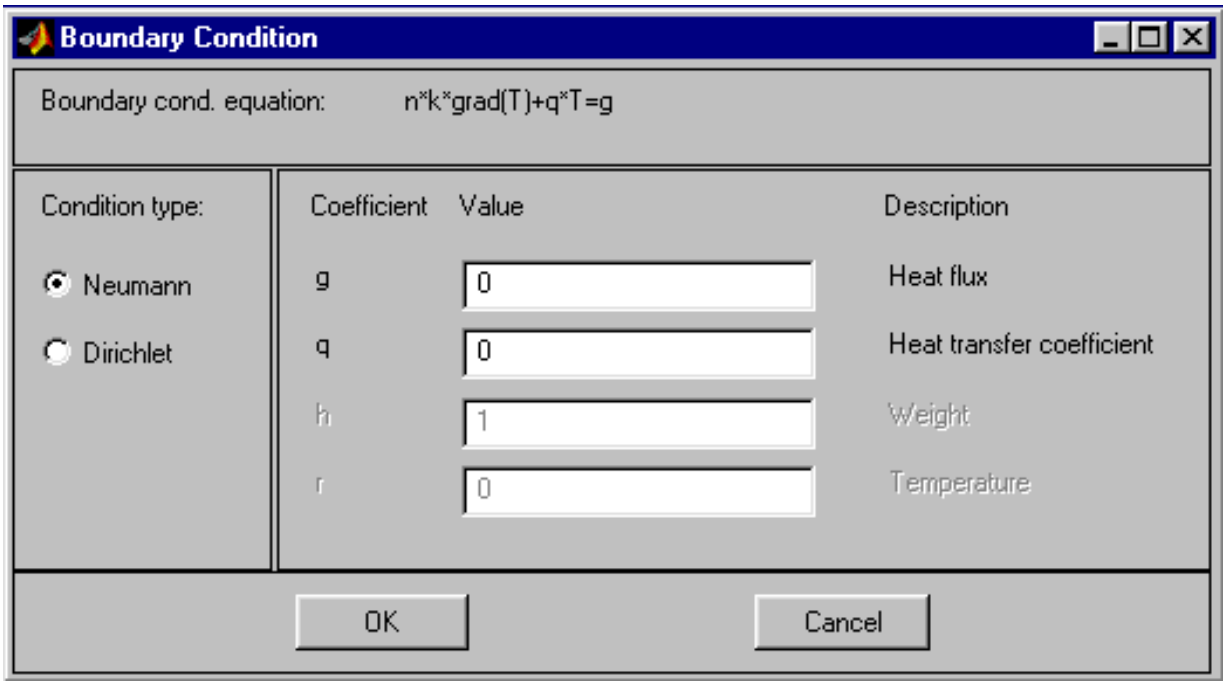


Рисунок 5.3 - Задача условий Неймана

• После этого на правой (рисунок 5.5) и на левой границах (рисунок 5.4) пластины зададим условия Дирихле. h – весовой коэффициент, а r – заданная температура (рисунок 5.4).

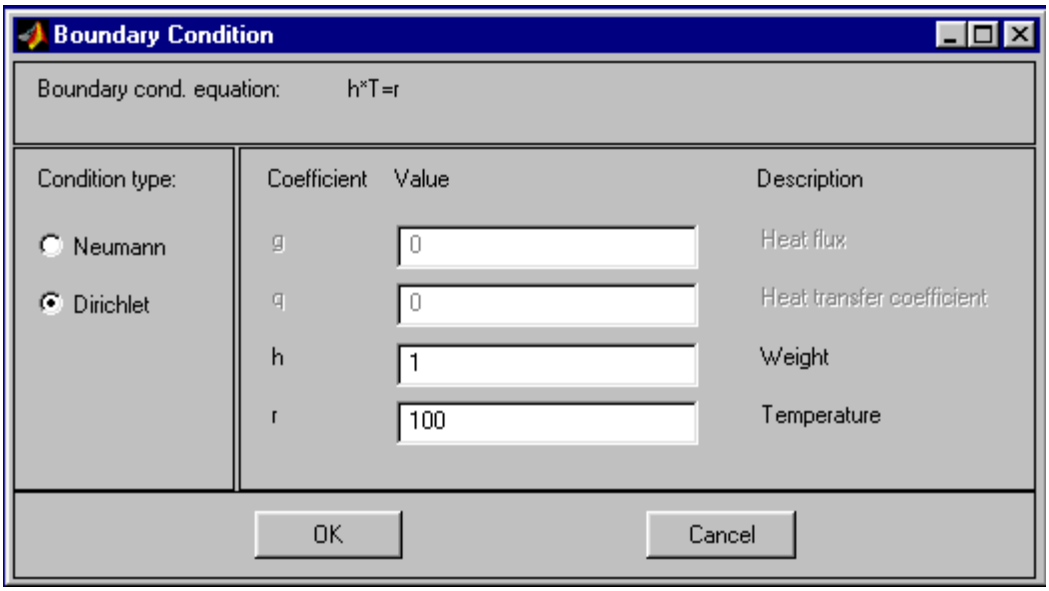


Рисунок 5.4 - Задача условий *Дирихле* для левой границы

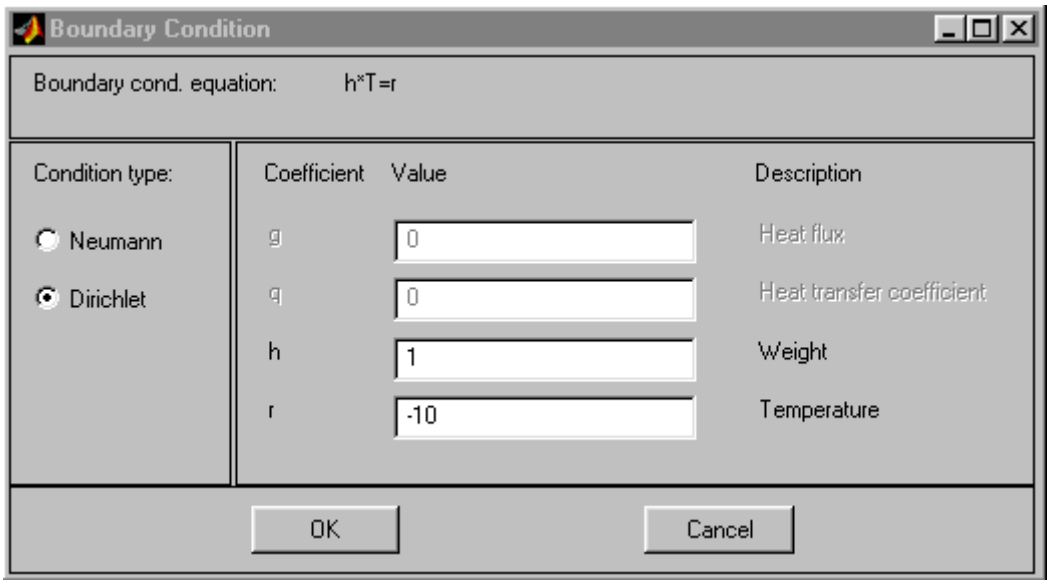


Рисунок 5.5 - Задача условий *Дирихле* для правой границы

4. После задачи граничных условий необходимо разбить площадь на конечные элементы, нажав или несколько раз.

5. После этого необходимо указать тип и параметры дифференциального уравнения. В пункте Options в подпункте Applications необходимо выбрать Heat Transfer. Так как в нашем случае процесс *нестационарный*, выбираем уравнение *параболического* типа

Приведенное уравнение полностью совпадает с приведенным выше уравнением (5.4), за исключением того момента, что в

уравнении, приведенном в диалоговом окне, учитывается также и конвективная составляющая.

В диалоговом окне записано следующее уравнение:

$$\mathrm{rho}\cdot C\cdot T' - \mathrm{div}\left(k\cdot \mathrm{grad}(T)\right) = Q + h\cdot(\mathrm{Text} - T) \qquad (9.5)$$

где $\mathrm{rho}\cdot C\cdot T' = c\cdot\rho\cdot\dfrac{\partial u}{\partial t}$; $\mathrm{div}\left(k\cdot\mathrm{grad}(T)\right) = \lambda\cdot\left(\dfrac{\partial^2 u}{\partial x^2}\right)$;

$Q = q_v$ – внутренняя энергия; $h\cdot(\mathrm{Text} - T)$ – конвективная составляющая, реализующая закон Ньютона – Рихмана, h – *коэффициент теплоотдачи*. Зададим параметры, как показано на рисунке 5.6.

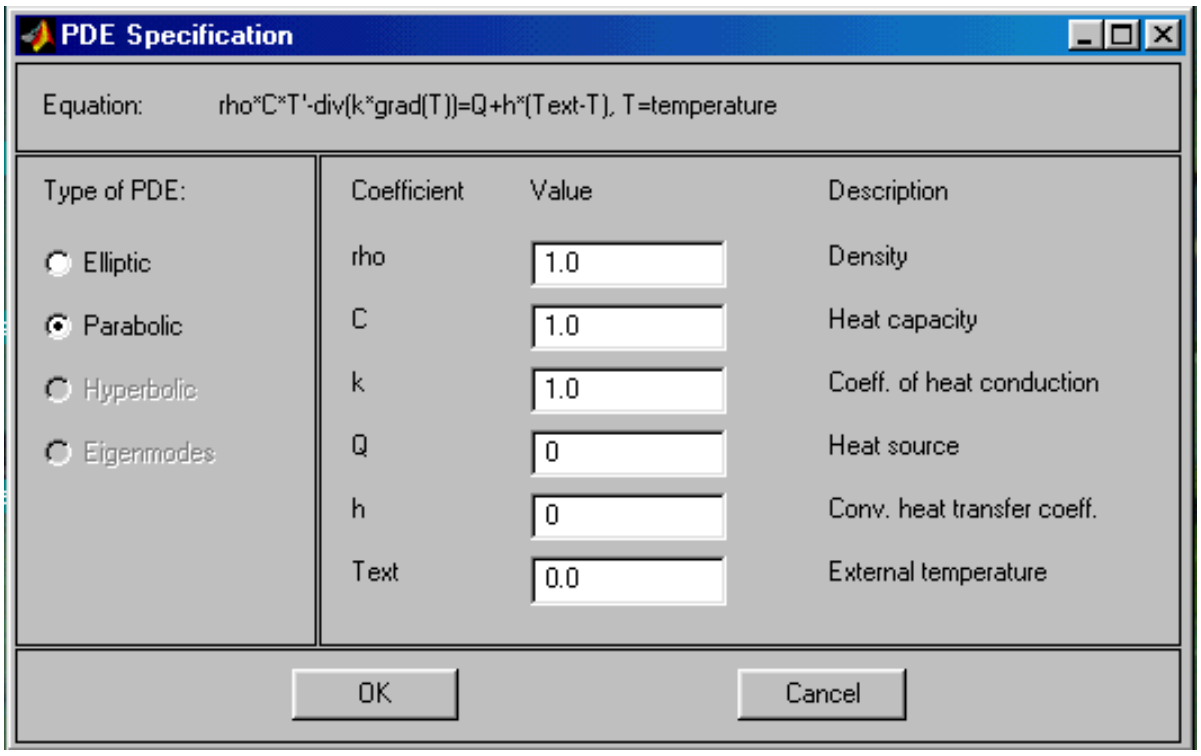


Рисунок 5.6 - Задача параметров дифференциального уравнения

6. Начальные условия зададим в меню Solve. В пункте Parameters зададим время моделирования и начальные условия: u(0), u'(0)=0. Его можно задать отрезком (0:5) или интервалами (0:0.1:0.2:03:5). Интервалы не обязательно равные. Значения отделяются двоеточием (рисунок 5.7).

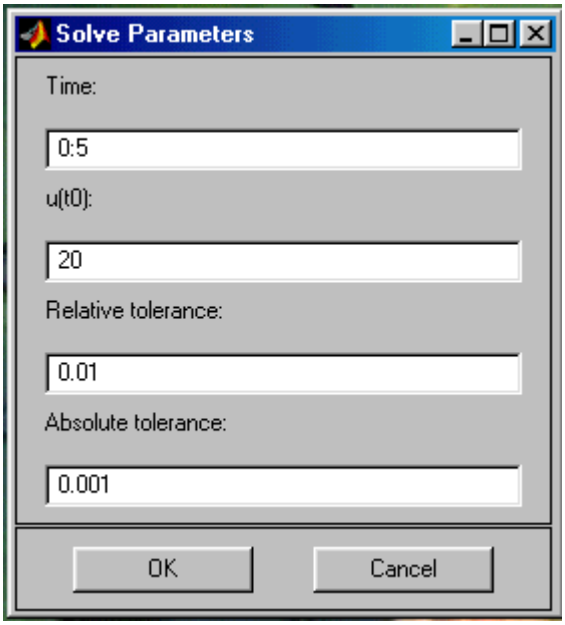


Рисунок 5.7 - Параметры времени и начальных условий

7. Для улучшения качества отображения решения можно сделать некоторые настройки. Нажать и в открывшемся диалоговом окне указать (рисунок 5.8):

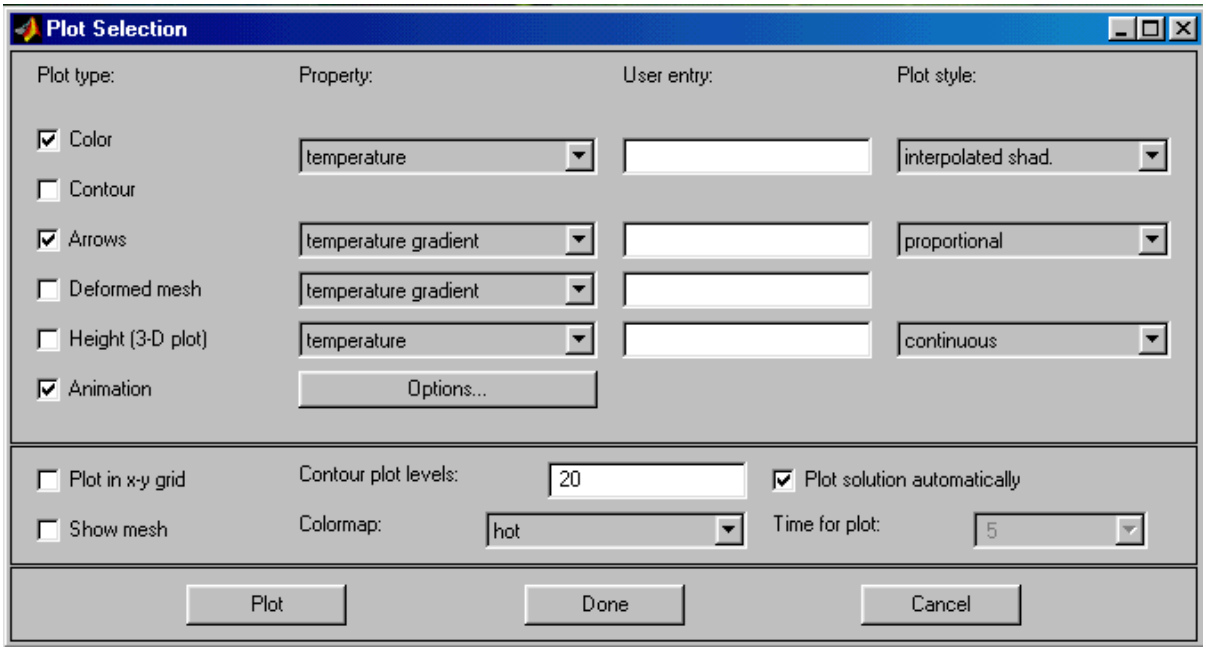


Рисунок 5.8 - Расширенные параметры вывода решения

8. Подав команду Plot наблюдаем искомое распределение температур.

Задания для самостоятельного выполнения

Задача 1

Решить задачу нагрева пластины толщиной 2 см., высотой 1,6 см. По средине пластины имеется круглое отверстие радиусом 0,2 см. Продолжительность нагрева 20 секунд. Граничные условия:

- u = 500 – на внутренних границах (на границах с отверстия);
- u = 300 – на правой и левой границе пластины;
- на остальных границах y_i .

Начальное условие u(0) = 20.

Вывести распределение температуре по длине пластинки. Плотность и теплоемкость пластины принять равными 1, а коэффициент теплопроводности принять равным 200.

Задача 2

Задать стержень радиусом 1 см. В стержне имеются два отверстия толщиной 0,2 см и высотой 0,8 см, которые могут располагаться произвольно в любом месте сечения. Получить распределение температур в пределах области. Продолжительность нагрева 20 секунд. Граничные условия:

- u = 250 – на внешних границах стержня;
- u = 150 – на внутренних границах;

Начальное условие u(0) = 20 по длине стержня. Плотность и теплоемкость стержня принять равными 1, а коэффициент теплопроводности принять равным 20. При решении уравнения учесть конвективную составляющую. Температуру окружающей среды принять равной 20°С. Коэффициент теплоотдачи k принять равным 2.

Библиографический список

1. Дьяконов, В.П. SIMULINK 4. Специальный справочник / В.П. Дьяконов. – СПб: Питер, 2002. – 528 с.
2. Дьяконов, В.П. MATLAB 6: Учеб. курс / В.П. Дьяконов. - СПб.: Питер, 2001. – 592 с.
3. Дюк, В. Информационные технологии в медико-биологических исследованиях / В. Дюк, В. Эмануэль. - СПб.: Питер, 2013. – 528 с.