

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ,
МОЛОДЕЖИ И СПОРТА УКРАИНЫ
НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ХАРЬКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНЫМ РАБОТАМ
по курсам: «ОБРАТНЫЕ ЗАДАЧИ ДИНАМИКИ» и
«МЕТОДЫ СИНТЕЗА И ОПТИМИЗАЦИИ СИСТЕМ УПРАВЛЕНИЯ»
для студентов специальностей 7.080202 и 7.080201

Утверждено
редакционно-издательским
советом университета,
протокол № 2 от 01.12.2010 г.

Харьков
НТУ «ХПИ»
2011

Методические указания к лабораторным работам по курсам: «Обратные задачи динамики» и «Методы синтеза и оптимизации систем управления» для студентов специальностей 7.080202 и 7.080201 / сост. Успенский В.Б., Шипулина Л.В. – Х.: НТУ «ХПИ», 2011. –28 с.

Составители: В.Б. Успенский
 Л.В. Шипулина

Рецензент А.И. Белов

Кафедра «Системы и процессы управления»

Цель лабораторных работ – освоение студентами новых методов разработки замкнутых систем автоматического регулирования и управления. В теоретическом и методическом отношении эти методы базируются на концепциях обратных задач динамики в сочетании с минимизацией локальных функционалов, характеризующих энергию движения в окрестности назначенных траекторий. Синтезированные алгоритмы имеют нетрадиционные структуры и придают системам естественные свойства адаптивности – слабой чувствительности к изменению параметров и возмущающим силам. Лабораторные работы предназначены для студентов специальностей 7.080202 и 7.080201.

The purpose of laboratory work - students mastering of new methods of developing closed systems of automatic regulation and control. In the theoretical and methodological point of view, these methods are based on the concepts of inverse problems of dynamics in conjunction with minimizing local functionals characterizing the energy of motion in the vicinity of the designated paths. Synthesized algorithms have unconventional structure and give the system the natural characteristics of adaptability - low sensitivity to changes in parameters and perturbing forces. Laboratory works designed for students 7.080201 and 7.080202 specialties .

СОДЕРЖАНИЕ

ВСТУПЛЕНИЕ.....	4
ЛАБОРАТОРНАЯ РАБОТА 1. Вычислительный алгоритм дискретного управления	5
ЛАБОРАТОРНАЯ РАБОТА 2. Метод управления по старшей производной.....	8
ЛАБОРАТОРНАЯ РАБОТА 3. Система обработки заданного программного движения	10
ЛАБОРАТОРНАЯ РАБОТА 4. Методы решения задачи терминального управления	13
4.1. Метод программного управления	13
4.2. Синтез по конечному состоянию.....	15
4.3. Синтез по методу преследования ведущей точки	16
ЛАБОРАТОРНАЯ РАБОТА 5. Оптимальное управление детерминированными и стохастическими системами	19
5.1. Детерминированная система с управлением по полному вектору состояния.....	19
5.2. Детерминированная система с измерением неполного вектора состояния.....	21
5.3. Стохастическая система с измерением неполного вектора состояния.....	24
Список литературы	26
Приложение 1. Краткое описание пакета прикладных программ VisSim	27

ВСТУПЛЕНИЕ

Лабораторные работы предназначены для освоения студентами новых методов разработки замкнутых систем автоматического регулирования и управления. В теоретическом и методическом отношении эти методы базируются на концепциях обратных задач динамики в сочетании с минимизацией локальных функционалов, характеризующих энергию движения в окрестности назначенных траекторий. Синтезированные алгоритмы имеют нетрадиционные структуры и придают системам естественные свойства адаптивности – слабой чувствительности к изменению параметров и возмущающим силам [1].

Обратная задача динамики формулируется следующим образом. Известна математическая модель движения системы $\dot{x} = f(x, u)$. Задано ее начальное состояние $x(0) = x_0$ и назначена требуемая траектория движения $x(t) = x^*(t)$. Необходимо построить управление $u^*(t, x)$, которое обеспечивает устойчивую реализацию $x(t) \rightarrow x^*(t)$. В математическом отношении концепция обратных задач динамики реализуется в форме синтеза алгоритма управления, при котором управляемая система обладает требуемыми динамическими характеристиками.

В методических указаниях к лабораторным работам описываются новые методы проектирования систем управления. В лабораторной работе 1 рассмотрен вычислительный алгоритм дискретного управления, позволяющий осуществить синтез замкнутой системы, движение которой происходит по предписанной программе, без использования модели системы.

В лабораторной работе 2 рассмотрен метод управления по старшей производной, дающий возможность построить замкнутую систему, обеспечивающую движение объекта из начальной точки по траектории, близкой к эталонной, задаваемой дифференциальным уравнением, как для стационарных, так и для нестационарных систем.

В лабораторной работе 3 приведена система отработки заданного программного движения для систем достаточно высокой сложности. В основу решения этой задачи полагается принцип управления по старшей производной.

В лабораторной работе 4 приведены три метода решения задачи терминального управления, которая состоит в переводе вектора состояния системы из начальной точки в назначенную конечную за фиксированное время.

В лабораторной работе 5 рассматривается оптимальное по быстродействию управление детерминированными и стохастическими системами и делается вывод, что использование детерминированной модели с наблюдателем состояния в условии измерения неполного вектора состояния и случайных возмущений не вполне эффективно, а применение фильтра Калмана дает наилучшую оценку переменных и наименьшее время процесса управления.

Во всех работах исследуется динамика проектируемых систем, приводятся схемы и результаты математического моделирования.

ЛАБОРАТОРНАЯ РАБОТА 1

ВЫЧИСЛИТЕЛЬНЫЙ АЛГОРИТМ ДИСКРЕТНОГО УПРАВЛЕНИЯ

Постановка задачи

Математическая модель движения системы имеет вид

$$\dot{x} = f(x, u) \quad (1.1)$$

Задано ее начальное состояние $x(0) = x_0$ и назначена требуемая траектория движения $x(t) = x^*(t)$. Необходимо построить управление $u(x)$, которое обеспечивает устойчивую реализацию $x(t) \rightarrow x^*(t)$.

Описание метода

Введем дискретные равноотстоящие моменты времени

$$t_0, t_1, \dots, t_N \quad \text{с интервалом} \quad \Delta t = t_{n+1} - t_n$$

и перейдем от непрерывной модели (1.1) к конечно-разностной по схеме Эйлера

$$x_{n+1} = x_n + f_n \cdot \Delta t, \quad (1.2)$$

где $f_n = f(x_n, u_n)$.

Потребуем, чтобы $x_{n+1} = x_{n+1}^* = x^*(t_{n+1})$ и представим f_n в виде

$$f_n = f_{n-1} + \frac{\partial f}{\partial u} \cdot \Delta u = \frac{x_n - x_{n-1}}{\Delta t} + \frac{x_n - 2x_{n-1} + x_{n-2}}{\Delta u_{n-1} \cdot \Delta t} \cdot \Delta u_n \quad (1.3)$$

Подставив (1.3) в (1.2), получим приращение функции управления

$$\Delta u_n = \frac{x_{n+1}^* - 2x_n + x_{n-1}}{x_n - 2x_{n-1} + x_{n-2}} \cdot \Delta u_{n-1} \quad (1.4)$$

и значение управления на n -ном шаге

$$u_n = u_{n-1} + \Delta u_n \quad (1.5)$$

Для реализации данного алгоритма не требуется знание о модели объекта. Управление u_n является функцией состояния x_{n-2}, x_{n-1}, x_n и обеспечивает (приближенно) реализацию программного значения x_{n+1}^* .

Пример. Движение системы описывается уравнением

$$\dot{x} = 0,1x^2 + u \quad (1.6)$$

Начальное условие $x_0 = -30$, интервал дискретизации $\Delta t = 0,1$ с. и назначенное программное движение $x^* = 0,01t^2$.

Для численной реализации дискретного алгоритма (1.5) приращение Δu задается постоянным на первых трех шагах, а далее рассчитывается по формуле (1.4):

пусть $\Delta u = 0,1 \quad (t < 3 \cdot \Delta t)$

$$\Delta u_n = k \cdot \Delta u_{n-1}, \quad \text{где} \quad k = \frac{x_{n+1}^* - 2x_n + x_{n-1}}{x_n - 2x_{n-1} + x_{n-2}} \quad (t > 3 \cdot \Delta t) \quad (1.7)$$

Управление на каждом шаге рассчитывается по формуле (1.5), а реализованное движение системы находится по формуле (1.2):

$$x_n = x_{n-1} + (0,1x_{n-1}^2 + u_{n-1}) \cdot \Delta t. \quad (1.8)$$

На рисунке 1.1 представлена схема моделирования в VisSim [2] системы (1.6) с дискретным управлением (1.5). Запустить процесс интегрирования с шагом 0,1.

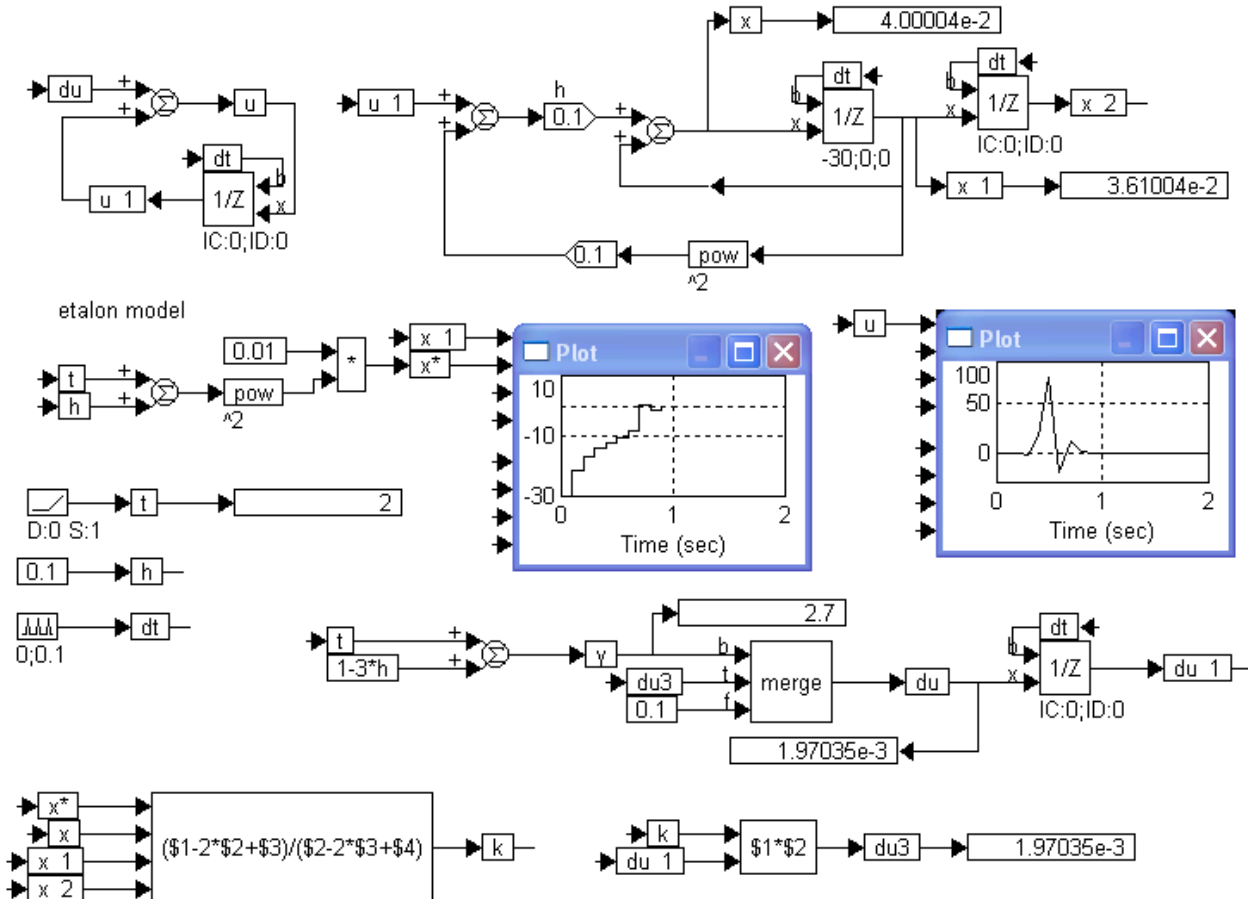


Рисунок 1.1

Схема решения разностных уравнений (1.5) и (1.8) построена с помощью блоков единичного запаздывания unitDelay (Blocks, Time Delay), которые осуществляют запаздывание дискретных переменных u , x на один интервал Δt (в схеме интервал Δt обозначен переменными h и dt). Управляющие импульсы (dt) поступают на верхний вход блока unitDelay от генератора импульсов PulseTrain (Blocks, Signal Producer). Интервал $\Delta t = 0.1c$ необходимо установить в окне свойств блока PulseTrain в поле Time Between Pulses (для этого необходимо щелкнуть правой кнопкой мыши (ПКМ) по пиктограмме блока).

Назначенное программное движение $x^* = 0,01t^2$ формируется как эталонная модель (etalon model), приращение Δu задается по формулам (1.7) с помощью блоков expression ((Blocks) и merge (Blocks, Nonlinear).

Применение блока expression позволяет уменьшить количество блоков в схеме и рассчитать величину k по формуле (1.7). В блоке expression формальные

переменные \$1, \$2,... \$n обозначают переменные, подаваемые на входы №1, №2,... №n блока для записи операций над ними (умножения, сложения, деления и т.д.). Для увеличения числа входов блока нажать $\xrightarrow{+}$ на панели инструментов и щелкнуть на изображении входа.

Блок merge реализует конструкцию условного оператора if...then...else и задает приращение Δu по формуле (1.7) в зависимости от условия boolean:

$$du_n = k \cdot du_{n-1}, \text{ если } |y| \geq 1 \quad (true)$$

$$du = 0.1, \quad \text{если } |y| < 1 \quad (false)$$

Работу блока merge можно проследить, если пошагово пройти начальные 3-4 интервала Δt , нажимая на Single Step на панели инструментов или F10.

Собрать схему, изображенную на рис. 1.1, и запустить процесс, используя шаг интегрирования 0.1 с. (Simulate, Simulation Properties, Time Step).

По результатам моделирования можно сделать вывод, что в течение 1с. управление u_n обеспечивает реализацию программного значения x_{n+1}^* .

Задание:

1) Для иллюстрации устойчивости дискретного алгоритма в схеме моделирования (рис. 1.1) изменять начальные условия $(-30 \div +5)$ и назначенное программное движение $x^* = (1 \div 0,01)t^2$.

Сделать выводы по результатам моделирования при внесении изменений.

2) Составить схему моделирования программного управления (рис. 1.2). Назначена траектория $xp^* = 0.01t^2 - 30$ при начальном условии $xp_0^* = -30$. Программное управление, найденное из уравнения (1.6) равно: $up^* = \dot{x}p^* - 0.1(xp^*)^2$.

Сравнить результаты моделирования на рис. 1.1 и рис. 1.2.

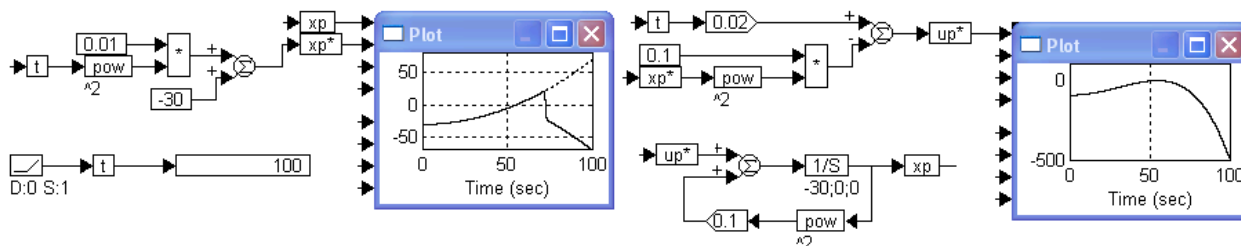


Рисунок 1.2

Контрольные вопросы к лабораторной работе 1

1. Прямая и обратная задача динамики управляемых систем.
2. Построение программного управления движением динамических систем. Недостаток программного управления.
3. Построение вычислительного алгоритма дискретного управления и его достоинства.

ЛАБОРАТОРНАЯ РАБОТА 2 МЕТОД УПРАВЛЕНИЯ ПО СТАРШЕЙ ПРОИЗВОДНОЙ

Описание метода

В этом методе в качестве управляющего сигнала формируется интеграл от невязки старшей производной. Этот закон управления соответствует тому, что система практически оптимальна по критерию минимума энергии ускорения.

Построение алгоритмов управления (синтез структуры алгоритмов) выполняется в такой последовательности:

а) формируется эталонная модель, динамические характеристики которой соответствуют требованиям, предъявляемым к проектируемой автоматической системе;

б) для заданной структуры модели управляемого объекта и принятой структуры эталонной модели выписывается закон управления.

Пусть из уравнения движения системы выражена старшая производная через остальные переменные

$$\dot{x}^{(n)} = f(x, \dot{x}, \dots, x^{(n-1)}, u) \quad (2.1)$$

Невязка старшей производной равна

$$\Delta x^{(n)} = x^{*(n)} - x^{(n)}, \quad (2.2)$$

где $x^{*(n)}$ – желаемое значение по эталонной модели, а $x^{(n)}$ – располагаемое значение старшей производной.

Управление формируется в виде интеграла от невязки старшей производной (2.2)

$$u = k \int \Delta x^{(n)} dt, \quad (2.3)$$

где k выбирается большим. При $k \rightarrow \infty$ система, почти оптимальная по критерию минимума энергии ускорения (2.3), обладает замечательными свойствами: ее динамические характеристики полностью совпадают с динамическими характеристиками эталонной модели и остаются неизменными при изменении параметров управляемого объекта.

Пример.

Пусть движение управляемым объектом задается в виде:

$$\ddot{x} + a_1(t)\dot{x} + a_0(t)x = u, \quad (2.4)$$

где $a_1(t)$, $a_0(t)$ – параметры, медленно меняющиеся во времени в некоторых пределах.

Заданы начальные условия $x(0) = x_0$, $\dot{x}(0) = \dot{x}_0$.

Требуется построить замкнутую систему, обеспечивающую движение объекта из начальной точки близким к траектории, задаваемой эталонным дифференциальным уравнением

$$\ddot{x}^* + b_1\dot{x}^* + b_0x^* = 0, \quad (2.5)$$

где коэффициенты b_1, b_0 выбираются такими, чтобы корни соответствующего характеристического уравнения имели отрицательные вещественные части.

Формируем управление в виде интеграла от невязки старшей производной

$$u = k \int (\ddot{x}^* - \ddot{x}) dt = k \int (-b_1 \dot{x} - b_0 x - \ddot{x}) dt$$

Если k достаточно велико, то при таком управлении уравнение (2.4) будет стремиться к эталонному (2.5). Таким образом динамические характеристики проектируемой системы будут идентичны динамическим характеристикам эталонной модели независимо от изменения параметров системы (свойство параметрической адаптивности).

Пусть заданы параметры системы и модели:

$$a_1(t) = 0.5 + 0.2 \sin\left(\frac{2\pi}{100}t\right), \quad a_0(t) = 0.06 + 0.1 \cos\left(\frac{2\pi}{130}t\right), \quad b_1 = 0.05, \quad b_0 = 6 \cdot 10^{-4}$$

и начальные условия $x(0) = 1, \dot{x}(0) = 0$.

Составить схему моделирования системы с управлением по старшей производной (рис. 2.1) и запустить процесс, используя метод Рунге-Кутты 4-го порядка с шагом интегрирования 0.001 (RK4, 0.001).

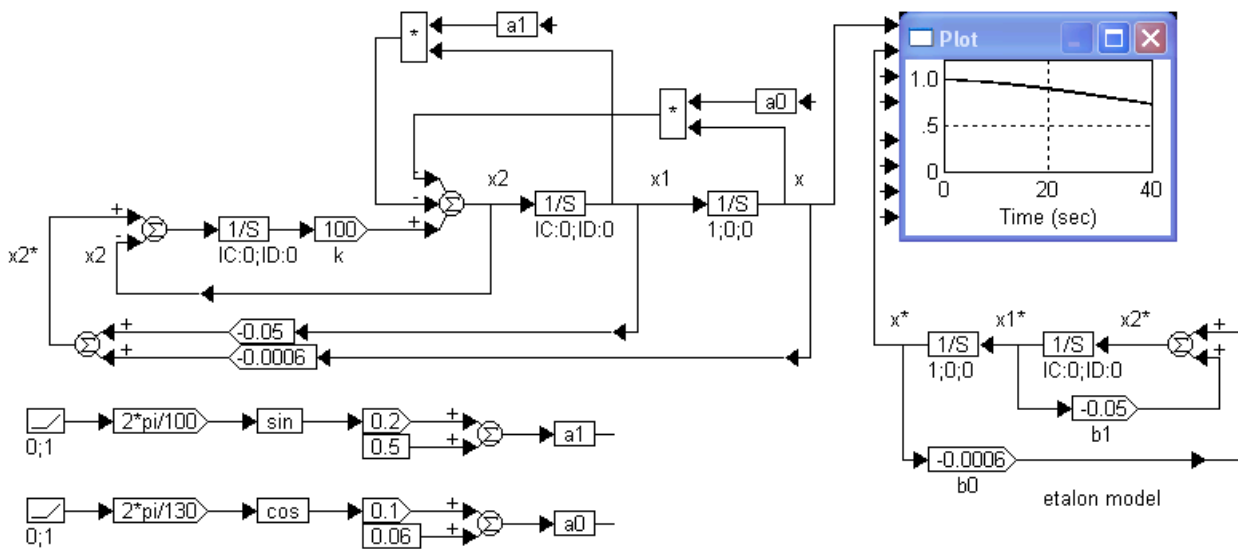


Рисунок 2.1

По результатам моделирования из рис. 2.1 видно, что движение системы близко траектории эталонной модели.

Задание:

1) экспериментально проверить влияние на близость реализуемой и эталонной траекторий коэффициента k , меняя его от 10 до 1000.

2) изменяя параметры $a_1(t)$ и $a_0(t)$, убедиться, что динамические характеристики проектируемой системы по-прежнему близки к динамическим характеристикам эталонной модели (свойство параметрической адаптивности).

ЛАБОРАТОРНАЯ РАБОТА 3 СИСТЕМА ОТРАБОТКИ ЗАДАННОГО ПРОГРАММНОГО ДВИЖЕНИЯ

Рассматривается задача структурного синтеза управления программным движением системы.

Постановка задачи

Пусть объект описывается уравнениями

$$\begin{aligned}\ddot{x}_1 &= u_1 + \dot{x}_1 + \dot{x}_2 \\ \ddot{x}_2 &= u_2,\end{aligned}\tag{3.1}$$

где u_1, u_2 - управляющие силы, x_1, x_2 - координаты точки. Заданы начальные условия $x_1(0) = 0, \dot{x}_1(0) = 1, x_2(0) = 0, \dot{x}_2(0) = 0.1$.

Требуется найти управление, при котором движение точки происходит таким образом, что скорость $v = \sqrt{\dot{x}_1^2 + \dot{x}_2^2}$ и координата x_1 меняются по некоторому заданному закону:

$$\begin{aligned}\dot{x}_1^2 + \dot{x}_2^2 - v_0^2 &= 0 \\ x_1 &= \sin \omega t\end{aligned}$$

Описание метода

Введем невязку – ошибку выполнения первого условия δ_1 и второго условия - δ_2 :

$$\begin{aligned}\delta_1 &= \dot{x}_1^2 + \dot{x}_2^2 - v_0^2 \\ \delta_2 &= x_1 - \sin \omega t\end{aligned}\tag{3.2}$$

Необходимо с помощью управления обеспечить, чтобы $\delta_1(t) \rightarrow 0, \delta_2(t) \rightarrow 0$ при $t \rightarrow \infty$.

Для переменной δ_1 введем эталонную модель

$$\dot{\delta}_1 + a\delta_1 = 0,\tag{3.3}$$

где $a > 0$, тогда $\delta_1(t) = \delta_{10} \cdot e^{-at}$ и $\delta_1(t) \rightarrow 0$ при $t \rightarrow \infty$.

Для переменной δ_2 в качестве модели вводим уравнение 2-го порядка

$$\ddot{\delta}_2 + b_1\dot{\delta}_2 + b_0\delta_2 = 0.\tag{3.4}$$

Числовые значения b_1 и b_0 определяют вид переходного процесса, поэтому с помощью этих параметров можно назначать такие динамические характеристики эталонной модели (3.4), которым соответствуют требуемые показатели переходного процесса: время достижения заданной окрестности стационарного состояния ($\delta_2(t) \rightarrow 0$ при $t \rightarrow \infty$), величину перерегулирования и др.

Для того, чтобы $\delta_2(t) \rightarrow 0$ при $t \rightarrow \infty$, необходимо, чтобы корни характеристического уравнения $s^2 + b_1s + b_0 = 0$ имели отрицательную вещественную часть. Пусть $s_1 = -2, s_2 = -3$, тогда

$$b_1 = -(s_1 + s_2) = 5$$

$$b_2 = s_1 s_2 = 6.$$

Преобразуем уравнение (3.3), подставив δ_1 из (3.2)

$$2\dot{x}_1\ddot{x}_1 + 2\dot{x}_2\ddot{x}_2 + a(\dot{x}_1^2 + \dot{x}_2^2 - v_0^2) = 0 \quad (3.5)$$

Преобразуем уравнение (3.4), подставив δ_2 из (3.2)

$$\begin{aligned} \dot{\delta}_2 &= \dot{x}_1 - \omega \cos \omega t; & \ddot{\delta}_2 &= \ddot{x}_1 + \omega^2 \sin \omega t; \\ \ddot{x}_1 + \omega^2 \sin \omega t + b_1(\dot{x}_1 - \omega \cos \omega t) + b_0(x_1 - \sin \omega t) &= 0 \end{aligned} \quad (3.6)$$

Выразим эталонные значения \ddot{x}_1^* , \ddot{x}_2^* из (3.5) и (3.6)

$$\ddot{x}_1^* = -[\omega^2 \sin \omega t + b_1(\dot{x}_1 - \omega \cos \omega t) + b_0(x_1 - \sin \omega t)] \quad \text{или} \quad \ddot{x}_1^* = F11$$

$$\ddot{x}_2^* = -[a(\dot{x}_1^2 + \dot{x}_2^2 - v_0^2) + 2\dot{x}_1 \cdot F11] / 2\dot{x}_2$$

Из уравнений (3.1) находим управление

$$u_1 = F11 - \dot{x}_1 - \dot{x}_2 \quad (3.7)$$

$$u_2 = \ddot{x}_2^*$$

На рисунке 3.1 представлена схема моделирования замкнутой системы с найденным управлением (3.7) при следующих ограничениях на управление:

$$|u_1| \leq 3$$

$$|u_2| \leq 1.$$

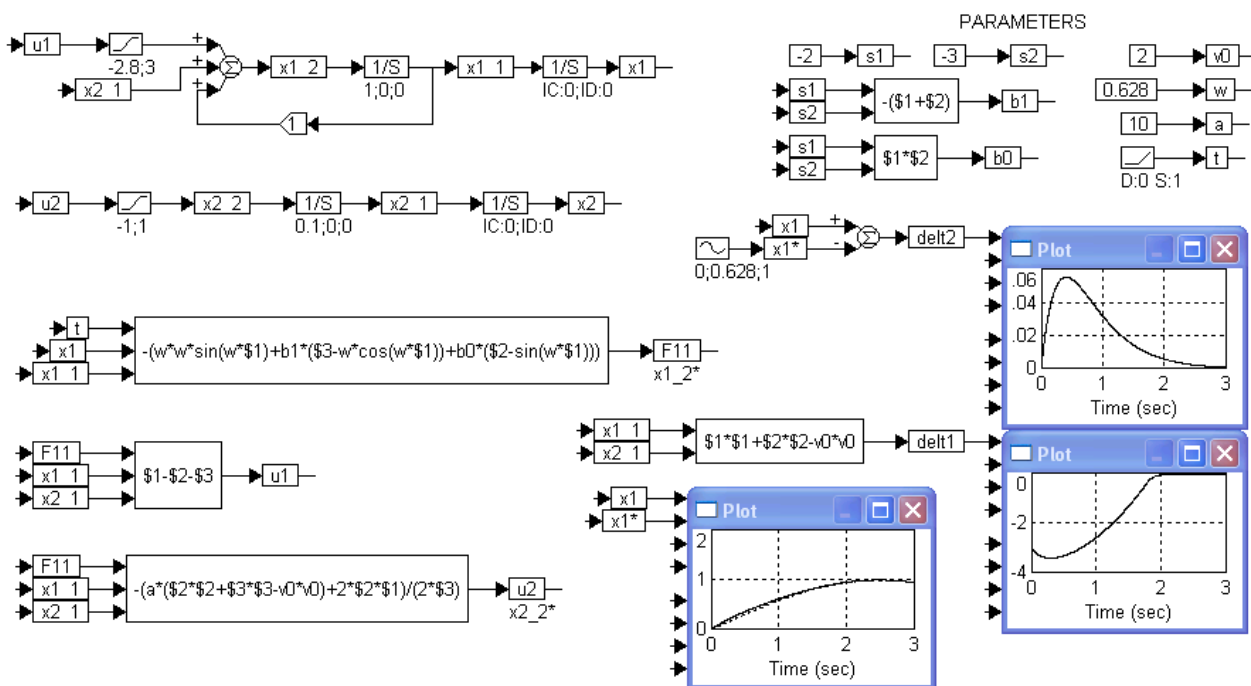


Рисунок 3.1

По результатам моделирования можно сделать вывод, что переходной процесс заканчивается за 3 с. и далее δ_1 и δ_2 (в схеме delt1 и delt2 соответственно) практически равны нулю.

Индивидуальное задание по лабораторной работе 3

Построить замкнутую систему управления объектом

$$\ddot{x}_1 = f_1(x_1, x_2, \dot{x}_1, \dot{x}_2, u)$$

$$\ddot{x}_2 = f_2(x_1, x_2, \dot{x}_1, \dot{x}_2, u)'$$

обеспечивающую его движение в соответствии с заданной программой

$$\varphi_1(x, \dot{x}) = C_1 \quad \varphi_2(x, \dot{x}) = C_2.$$

Все исходные данные взять из таблицы 3.1.

Таблица 3.1

№ варианта	Уравнения объекта	Программное движение
1	$\ddot{x}_1 = x_1 + x_2^2 + u_1$ $\ddot{x}_2 = -u_2 + x_1^2$	$\dot{x}_2(t) = v = -0.1$ $x_1^2(t) + x_2(t) = 0$
2	$\ddot{x}_1 = -x_1x_2 + u_1$ $\ddot{x}_2 = u_2$	$\dot{x}_2(t) = v = 0.1$ $x_1(t) + x_2(t) = 2$
3	$\ddot{x}_1 = x_1x_2 + u_1$ $\ddot{x}_2 = u_2$	$\dot{x}_1(t) = v = 0.1$ $x_1(t) + x_2(t) = 2$
4	$\ddot{x}_1 = 2x_1x_2 + u_1$ $\ddot{x}_2 = u_2 + \dot{x}_2$	$x_1(t) + x_2(t) = 2$ $\dot{x}_1(t) = 0.1$
5	$\ddot{x}_1 = u_1 - \dot{x}_1 - \dot{x}_2$ $\ddot{x}_2 = -x_1x_2 + u_2$	$x_1(t) - x_2(t) = 2t + 2$ $\dot{x}_2(t) = 0.1$
6	$\ddot{x}_1 = \dot{x}_1 + u_1$ $\ddot{x}_2 = 2x_2 + u_2$	$x_2(t) = 0.1$ $\sqrt{\dot{x}_1^2(t) + \dot{x}_2^2(t)} = v = 2$
7	$\ddot{x}_1 = \dot{x}_1 + u_1$ $\ddot{x}_2 = -x_1 + u_2$	$x_1(t) = 2t - 2$ $\dot{x}_1^2 + \dot{x}_2^2 = 4$
8	$\ddot{x}_1 = x_1 + x_2^2 + u_1$ $\ddot{x}_2 = -u_2$	$\dot{x}_2(t) = v = -0.1$ $x_1^2(t) + x_2(t) = 0$
9	$\ddot{x}_1 = x_2 + u_1$ $\ddot{x}_2 = x_1 - u_2$	$\dot{x}_1^2(t) + \dot{x}_2^2(t) = v^2 = 9^2$ $x_1(t) = 2t + 2$
10	$\ddot{x}_1 = -x_1 + u_1$ $\ddot{x}_2 = x_1 + u_2$	$\dot{x}_1^2(t) + \dot{x}_2^2(t) = v^2 = 5^2$ $x_2(t) = -2t + 2$

Начальные условия для всех вариантов:

$$x_1(0) = 0.1 \quad x_2(0) = -1$$

$$\dot{x}_1(0) = 1 \quad \dot{x}_2(0) = 0.01.$$

ЛАБОРАТОРНАЯ РАБОТА 4

МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ ТЕРМИНАЛЬНОГО УПРАВЛЕНИЯ

Задача терминального управления состоит в переводе вектора состояния системы из начальной точки в назначенную конечную за фиксированное время.

Постановка задачи

Рассмотрим следующую задачу: движение управляемой системы описывается уравнением

$$\ddot{x} = u + 2.5\pi^2 x, \quad (4.1)$$

где x и u - выходная переменная и управляющая функция.

Задано фиксированное время $T = 1$ с, начальное и конечное состояние

$$x_0 = 1, \quad x_T = 0, \quad \dot{x}_0 = 0, \quad \dot{x}_T = 0. \quad (4.2)$$

Требуется найти управление, переводящее систему из начального состояния в конечное за фиксированное время.

Рассмотрим три метода решения этой задачи.

4.1. Метод программного управления

Введем программную траекторию, соединяющую начальную и конечную точки, в виде полинома третьей степени

$$x^*(t) = at^3 + bt^2 + ct + d, \quad (4.3)$$

где параметры a, b, c, d находятся из условий (4.2) и имеют следующий вид:

$$\begin{aligned} d = x_0, \quad c = \dot{x}_0, \quad b = [3(x_T - x_0 - \dot{x}_0 T) - T(\dot{x}_T - \dot{x}_0)]/T^2, \\ a = [T(\dot{x}_T - \dot{x}_0) - 2(x_T - x_0 - \dot{x}_0 T)]/T^3. \end{aligned} \quad (4.4)$$

Используя параметры a, b, c, d (4.4) и формулу (4.1), можно найти программное изменение первой и второй производных $\dot{x}^*(t)$ и $\ddot{x}^*(t)$ и управление $u^*(t)$:

$$\begin{aligned} \dot{x}^*(t) = 3at^2 + 2bt + c, \quad \ddot{x}^*(t) = 6at + 2b, \\ u^*(t) = \ddot{x}^*(t) - 2.5\pi^2 x^*(t) = 6at + 2b - 2.5\pi^2 (at^3 + bt^2 + ct + d) \end{aligned} \quad (4.5)$$

На рис. 4.1 представлена схема моделирования разомкнутой системы с программным управлением $u^*(t)$, а на рис. 4.2 приводится содержание Comround block-ов для вычисления параметров a, b, c, d по формулам (4.4), $x^*(t)$ по (4.3), первой, второй производных $\dot{x}^*(t)$, $\ddot{x}^*(t)$ и управления $u^*(t)$ по (4.5).

По результатам моделирования можно сделать вывод, что $x(t)$ и $x^*(t)$ совпадают, однако при введении возмущения, например, постоянного, равного 1, совпадения уже не будет.

Это объясняется тем, что управление формируется в виде программы, т.е. без учета реализованного значения вектора состояния.

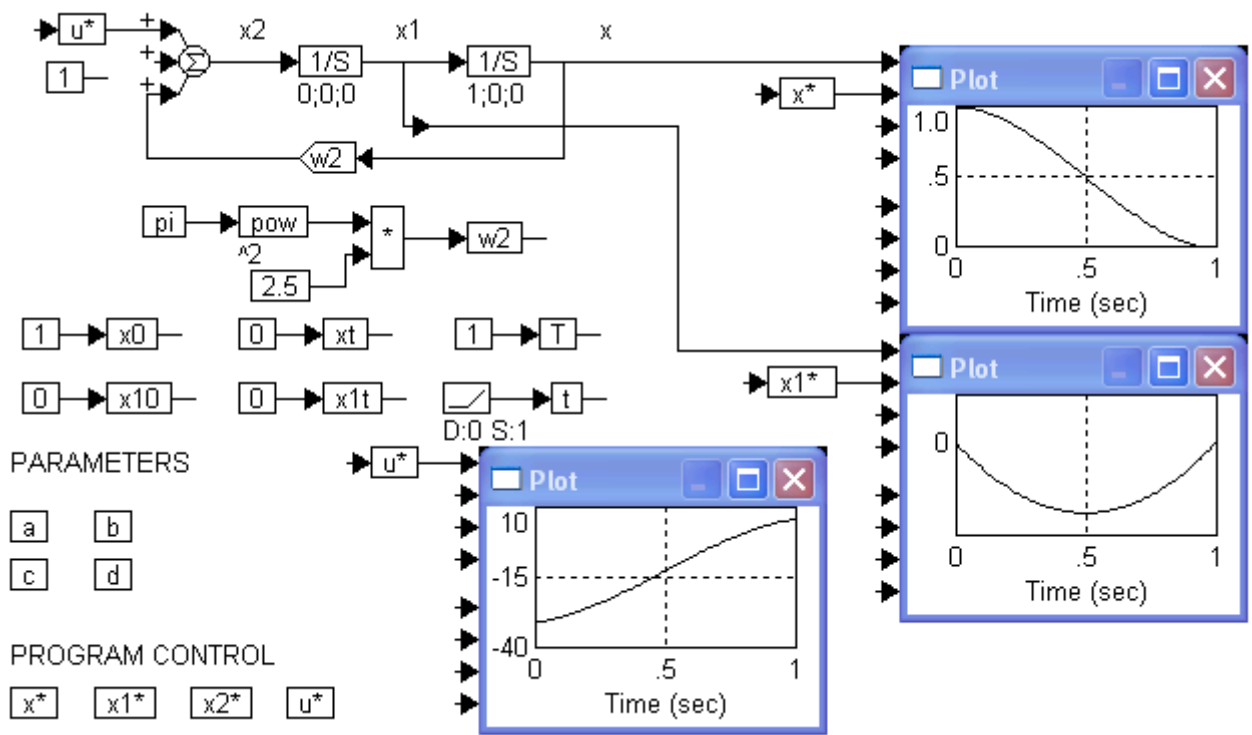


Рисунок 4.1

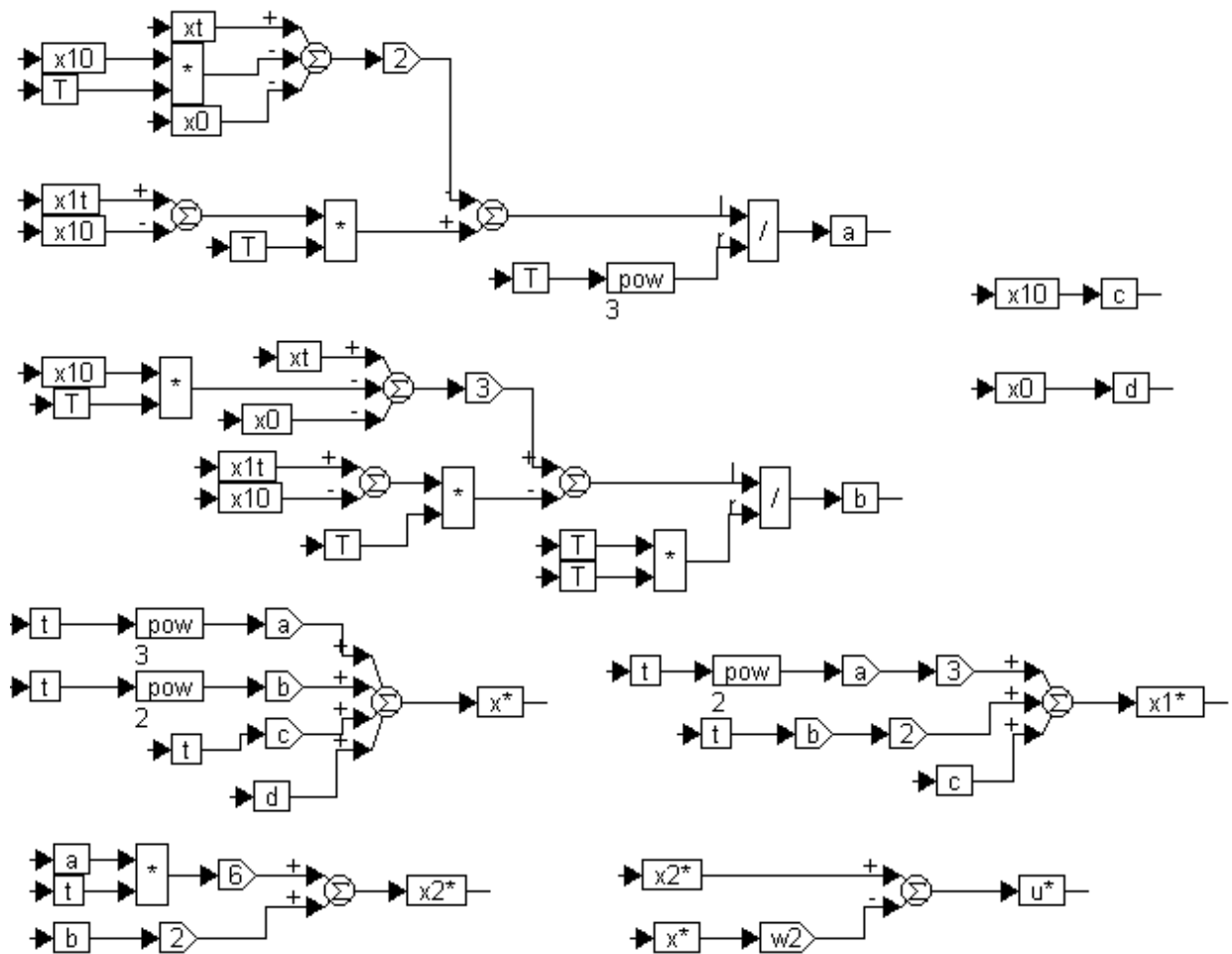


Рисунок 4.2

4.2. Синтез по конечному состоянию

Идея метода состоит в следующем: в каждый текущий момент времени t текущее состояние $x(t)$ и $\dot{x}(t)$ принимается в качестве начального, и заново решается задача перехода в конечную точку за время $T - t$. Коэффициенты $a1, b1, c1, d1$ находятся при этом по аналогии с (4.4):

$$d1 = x, \quad c1 = \dot{x}, \quad b1 = [3(x_T - x - \dot{x}(T - t)) - (T - t)(\dot{x}_T - \dot{x})]/(T - t)^2, \quad (4.6)$$

$$a1 = [(T - t)(\dot{x}_T - \dot{x}) - 2(x_T - x - \dot{x}(T - t))]/(T - t)^3.$$

Управление $u(t)$ с учетом того, что текущий момент принимается за начальный, рассчитывается по формуле (4.5) при $t = 0$:

$$u(t) = 2b1 - 2.5\pi^2 d1 \quad (4.7)$$

На рис. 4.3 представлена схема моделирования управления системы по конечному состоянию, а на рис. 4.4 приводится содержание Compound block-ов для вычисления параметров $a1, b1, c1, d1$ по формулам (4.6) и управления $u(t)$ по (4.7).

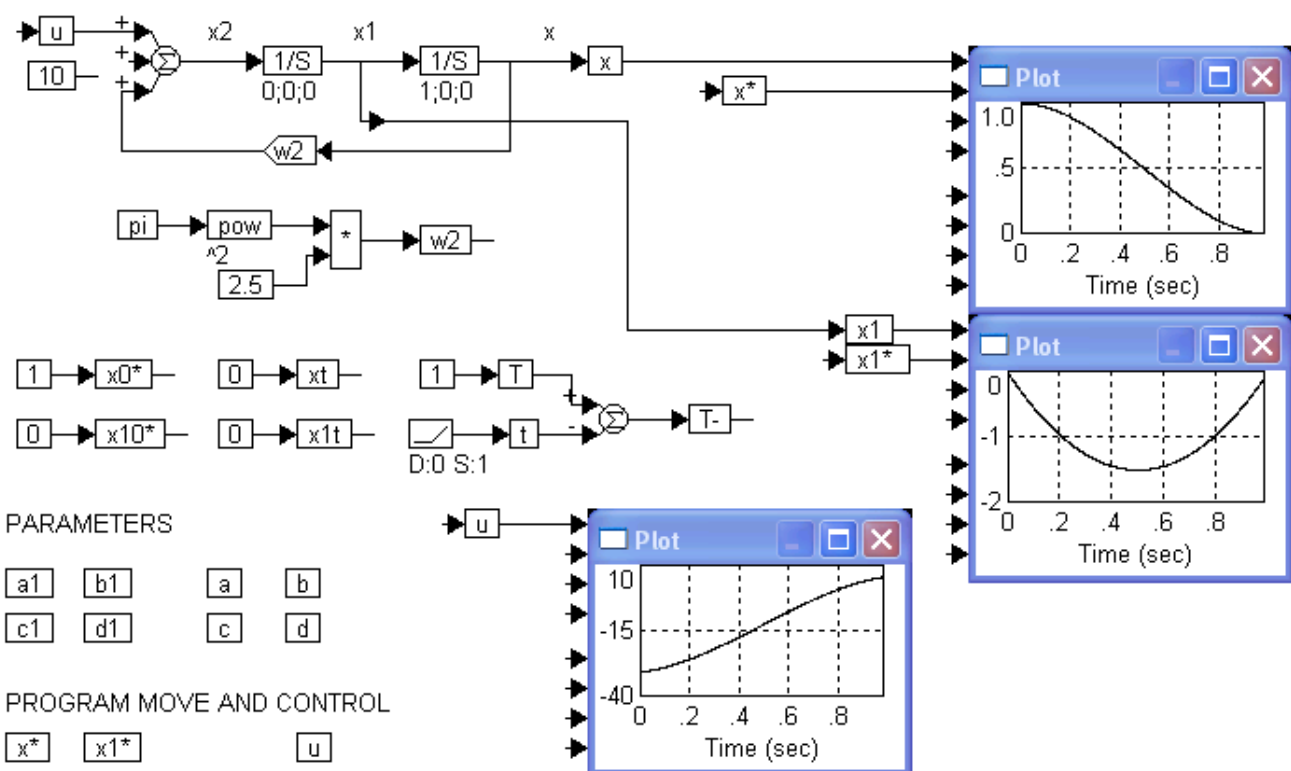


Рисунок 4.3

По результатам моделирования можно сделать вывод, что $x(t)$ и $x^*(t)$ совпадают, однако при введении возмущения, например, постоянного, равного 10, конечные точки реализуемой и эталонной траекторий близки, но полного совпадения траекторий уже не будет. Кроме того, главный недостаток метода состоит в том, что при $t \rightarrow T$ величина времени прогноза $T - t \rightarrow 0$, а это приводит к увеличению ошибок как в вычислениях, так и в управлении (в эксперименте в Simulation Properties выставлено Time End = $0.98 \cdot T$).

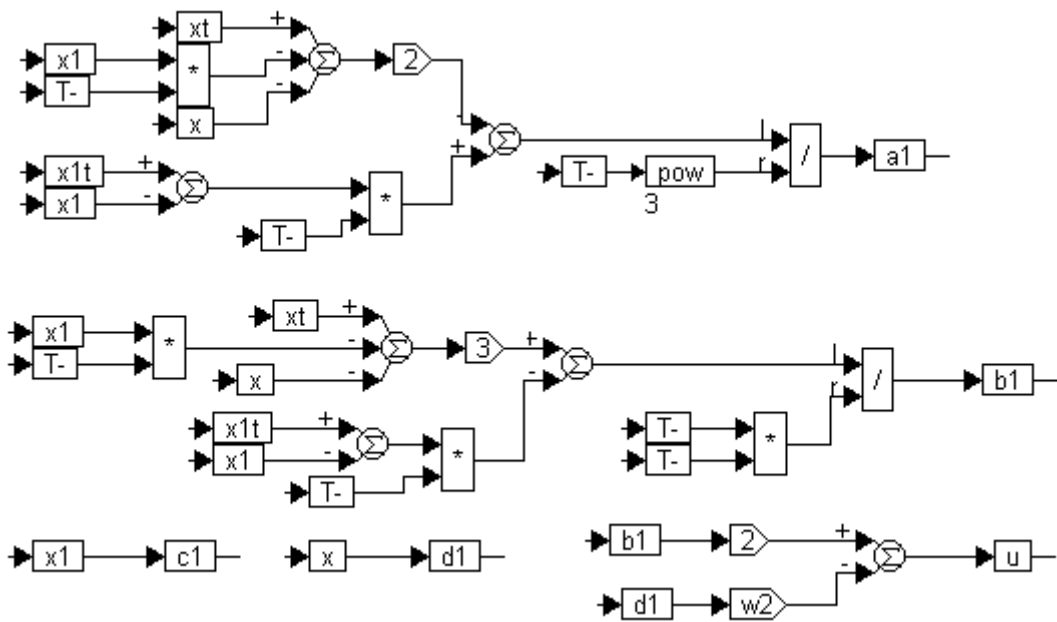


Рисунок 4.4

4.3. Синтез по методу преследования ведущей точки

Задача преследования ведущей точки состоит в переводе системы из текущей точки за время Δt в прогнозируемую точку, лежащую на исходной программной траектории. В каждый момент времени решается задача:

$$\begin{matrix} x(t) \\ \dot{x}(t) \end{matrix} \xrightarrow{\Delta t} \begin{cases} x^*(t + \Delta t) \\ \dot{x}^*(t + \Delta t) \end{cases}, \quad \text{где } \Delta t \text{ – глубина прогноза.}$$

Основные соотношения данного метода (по аналогии с (4.4)):

$$\begin{aligned} d2 = x, \quad c2 = \dot{x}, \quad b2 = [3(x^*(t + \Delta t) - x - \dot{x}\Delta t) - \Delta t(\dot{x}^*(t + \Delta t) - \dot{x})] / \Delta t^2, \\ a2 = [\Delta t(\dot{x}^*(t + \Delta t) - \dot{x}) - 2(x^*(t + \Delta t) - x - \dot{x}\Delta t)] / \Delta t^3. \end{aligned} \quad (4.8)$$

Эта схема синтеза, как и предыдущая, основана на перевычислении параметров $a2$, $b2$, $c2$, $d2$. При этом в законе управления $u(t, x, \dot{x})$ величина t – нулевой момент времени:

$$u(t, x, \dot{x}) = 2b2 - 2.5\pi^2 d2 \quad (4.9)$$

На рис. 4.5 представлена схема моделирования синтеза по методу преследования ведущей точки, а на рис. 4.6 приводится содержание Compound block-ов для вычисления параметров $a2$, $b2$, $c2$, $d2$ по формулам (4.8), программных траекторий $x^*(t)$, $\dot{x}^*(t)$, $x^*(t + dt)$, $\dot{x}^*(t + dt)$ по (4.3), (4.5) и управления $u(t, x, \dot{x})$ по (4.9).

По результатам моделирования можно сделать вывод, что $x(t)$ и $x^*(t)$ совпадают практически для всех t и даже при введении возмущения, например, постоянного, равного 100, совпадение сохраняется.

Таким образом, 3-й метод - синтез по методу преследования ведущей точки – является лучшим и обеспечивает движение по программной траектории и при действии возмущений.

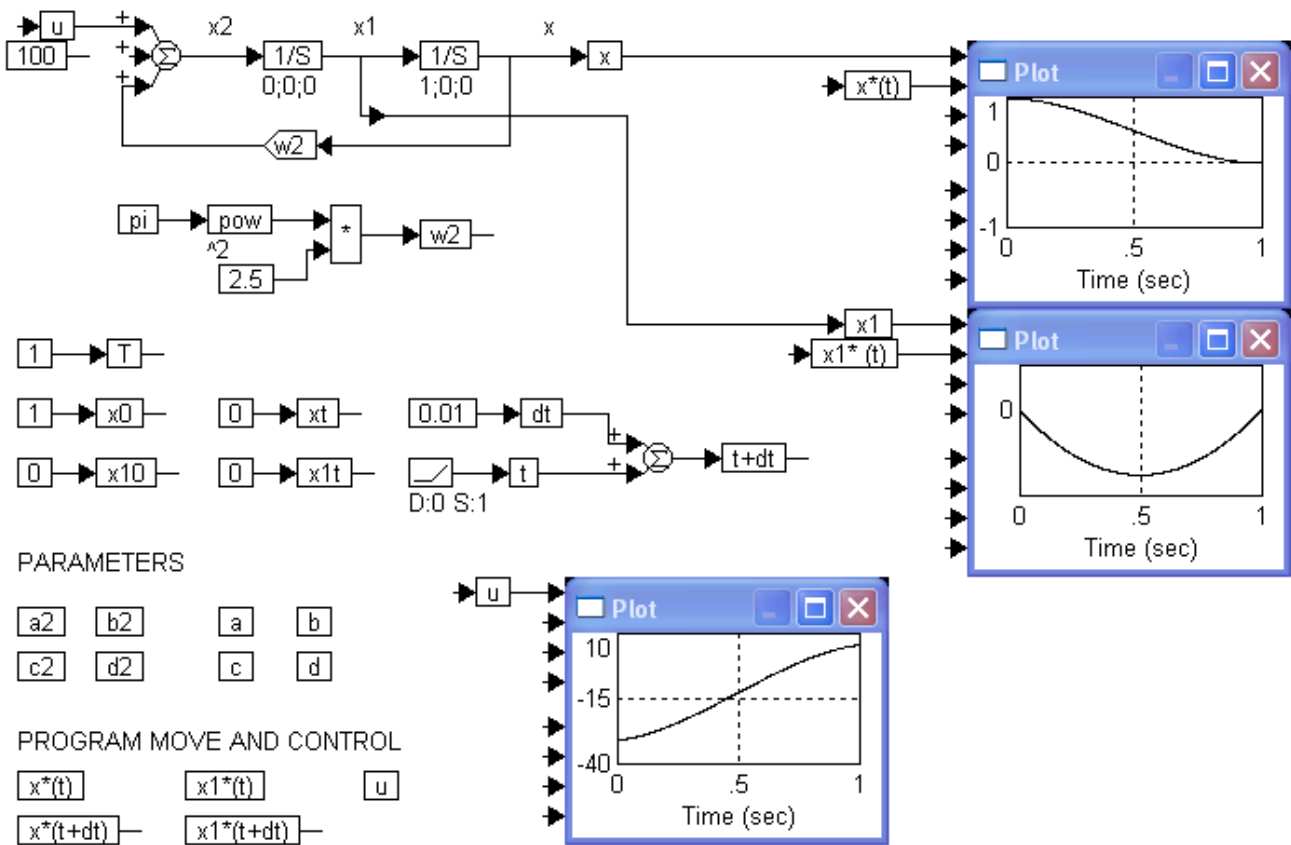


Рисунок 4.5

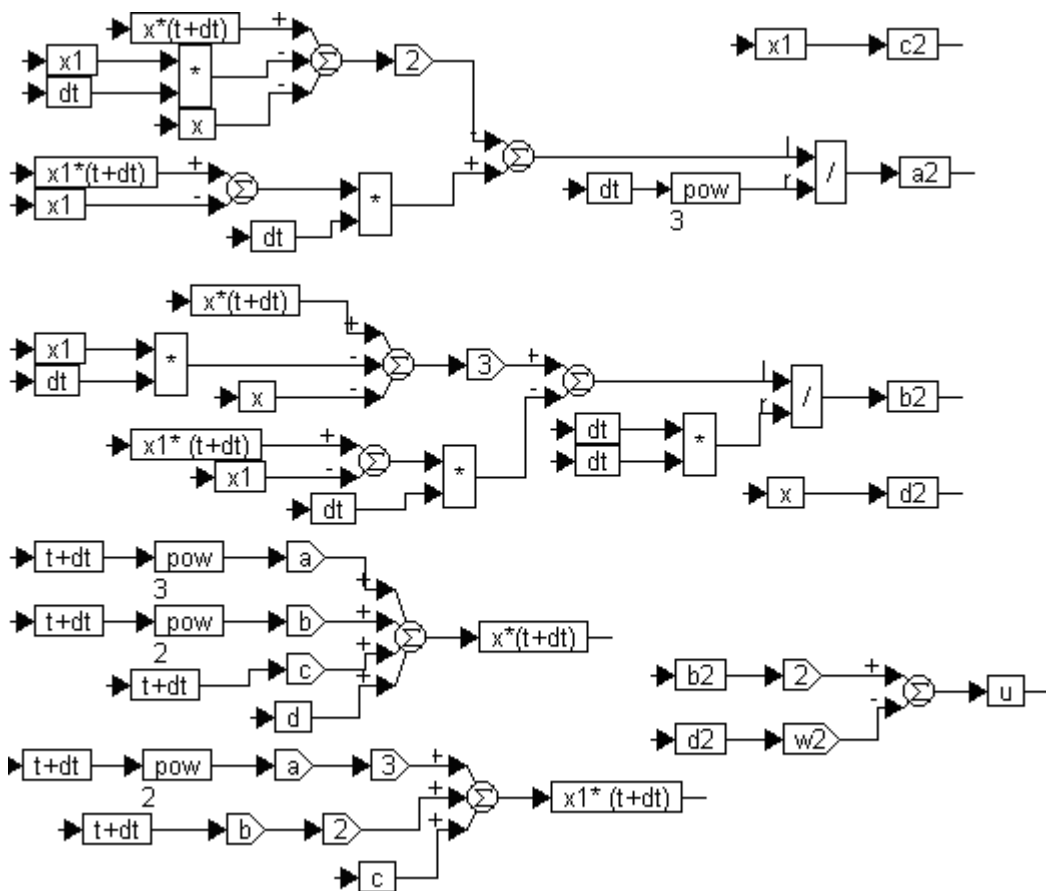


Рисунок 4.6

Индивидуальное задание по лабораторной работе 4

Построить замкнутую систему управления объектом

$$\ddot{x} = ax + bu,$$

переводящую вектор состояния системы из начальной точки $(x(0) = x_0, \dot{x}(0) = \dot{x}_0)$ в назначенную конечную $(x(T) = x_T, \dot{x}(T) = \dot{x}_T)$ за фиксированное время T . Все исходные данные взять из таблицы 4.1.

Таблица 4.1

№ варианта	Уравнения объекта	Начальная точка	Конечная точка	Время T , с
1	$\ddot{x} = \dot{x} + u$	$x(0) = 1, \dot{x}(0) = 1$	$x(T) = \dot{x}(T) = 0$	1
2	$\ddot{x} = x + u$	$x(0) = 1, \dot{x}(0) = -1$	$x(T) = \dot{x}(T) = 0$	1
3	$\dot{x}_1 = x_1 + x_2$ $\dot{x}_2 = u$	$x_1(0) = 4, x_2(0) = 0$	$x_1(T) = x_2(T) = 0$	2
4	$\dot{x}_1 = -x_2 + u$ $\dot{x}_2 = (x_1 - x_2)/2$	$x_1(0) = 1, x_2(0) = 0$	$x_1(T) = x_2(T) = 0$	1
5	$\dot{x}_1 = u$ $\dot{x}_2 = x_1$	$x_1(0) = 0, x_2(0) = 1$	$x_1(T) = 1, x_2(T) = 0$	1
6	$\ddot{x} - \dot{x} + x = u$	$x(0) = 0, \dot{x}(0) = 0$	$x(T) = 1, \dot{x}(T) = 0$	1
7	$\ddot{x} + \dot{x} = u$	$x(0) = 2, \dot{x}(0) = 0$	$x(T) = 0, \dot{x}(T) = -1$	1
8	$\dot{x}_1 - x_2 = 0$ $\dot{x}_2 + x_1 = u$	$x_1(0) = 1, x_2(0) = 1$	$x_1(T) = x_2(T) = 0$	1
9	$\ddot{x} + \dot{x} + x = u$	$x(0) = 1, \dot{x}(0) = 0$	$x(T) = \dot{x}(T) = 0$	1
10	$\dot{x}_1 = 0.5x_2$ $\dot{x}_2 = x_2 + u$	$x_1(0) = 1, x_2(0) = 0$	$x_1(T) = x_2(T) = 0$	1

Контрольные вопросы к лабораторной работе 4

1. Сформулировать задачу терминального управления.
2. Описать метод программного управления, указать его недостатки.
3. Привести расчет параметров программной траектории.
4. Как создается Compound block в пакете VisSim?
5. В чем состоит синтез по конечному состоянию? Указать его недостатки.
6. Как рассчитывается управление при синтезе по конечному состоянию?
7. Как находятся коэффициенты траектории при данном синтезе?
8. В чем состоит синтез по методу преследования ведущей точки? Преимущества метода.
9. Как рассчитывается управление и коэффициенты траектории при синтезе по методу преследования ведущей точки?

ЛАБОРАТОРНАЯ РАБОТА 5 ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ ДЕТЕРМИНИРОВАННЫМИ И СТОХАСТИЧЕСКИМИ СИСТЕМАМИ

Целью данной работы является синтез оптимального по быстродействию управления для детерминированных и стохастических систем и исследование влияния возмущения.

5.1. Детерминированная система с управлением по полному вектору состояния

Пусть движение материальной системы описывается уравнениями

$$\dot{x}_1 = x_2 \tag{5.1}$$

$$\dot{x}_2 = u,$$

а измерением является $y = x_1$,

или в виде векторно-матричной модели [3,4]:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{5.2}$$

где $x(t) = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, $A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, $C = (1 \ 0)$, $D = 0$.

Задано начальное и конечное состояние системы

$$x_1(0) = 2, \quad x_2(0) = 3, \quad x_1(T) = 0, \quad x_2(T) = 0.$$

При ограниченном по модулю управлении $|u| \leq 1$ требуется обеспечить оптимальное быстродействие, т.е. минимальное время перехода T . Для этого оптимальное управление синтезируется в виде [5]

$$u^0(x_1, x_2) = \begin{cases} -1, & \text{если } \{x_1, x_2\} \in \varepsilon^- \\ +1, & \text{если } \{x_1, x_2\} \in \varepsilon^+, \end{cases} \tag{5.3}$$

где $\varepsilon^- : \{x_1, x_2\} \begin{cases} x_1 \geq -x_2^2/2, & \text{при } x_2 > 0 \\ x_1 > x_2^2/2, & \text{при } x_2 \leq 0 \end{cases}$ $\varepsilon^+ : \{x_1, x_2\} \begin{cases} x_1 < -x_2^2/2, & \text{при } x_2 > 0 \\ x_1 \leq x_2^2/2, & \text{при } x_2 \leq 0 \end{cases}$

Схема моделирования движения системы (5.1) представлена на рис. 5.1, а схемы Compond Block-ов управления u^0 и остановки *stop* изображены на рис. 5.2 и рис. 5.3 соответственно.

Управление u^0 задается согласно (5.3) в виде (рис. 5.2)

$$u^0 = -\left[\text{sign}(x_1 + \text{sign} x_2 \cdot 0.5x_2^2) \right].$$

Блок остановки *stop* (рис. 5.3) предназначен для прекращения вычислений, если расстояние от точки $\{x_1, x_2\}$ на фазовой плоскости до начала координат становится меньше заданной точности $dr = 0.01$.

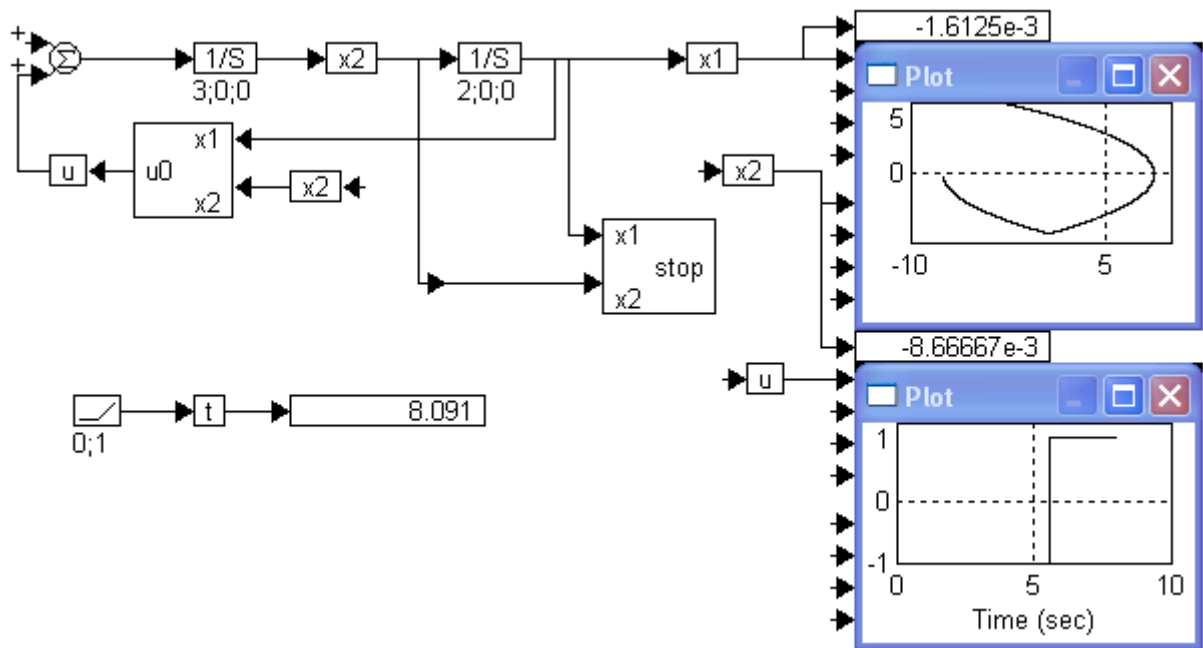


Рисунок 5.1

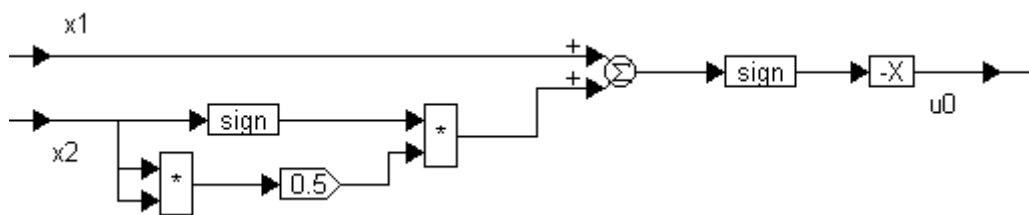


Рисунок 5.2

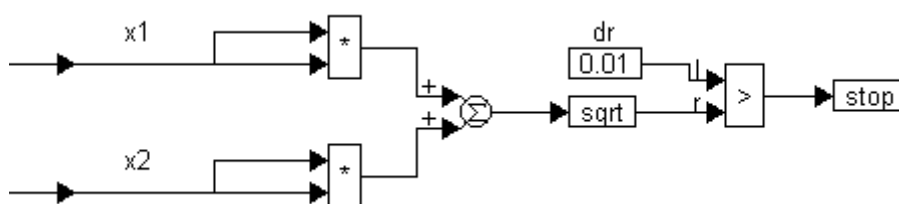


Рисунок 5.3

Запустив процесс моделирования (рис. 5.1, RK4, 0.001), получим, что процесс заканчивается за 8,091 с.

На рис. 5.4 изображена схема моделирования системы (5.1) при наличии случайного возмущения с нормальным законом распределения. Видно, что управление u_0 (рис. 5.2) обеспечивает переход системы в конечное состояние за время 7,99 с, многократно переключаясь от -1 до +1 и наоборот.

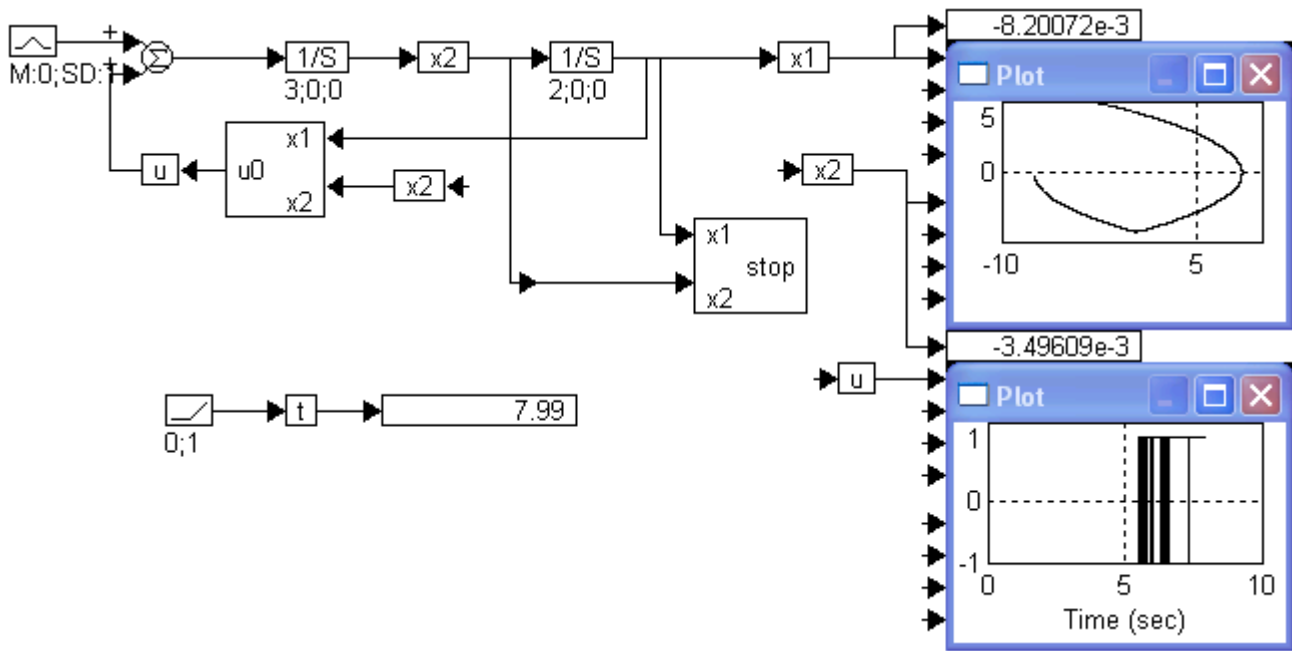


Рисунок 5.4

5.2. Детерминированная система с измерением неполного вектора состояния

Если не все переменные состояния системы могут быть измерены, то они подлежат оценке в результате наблюдения за поведением объекта (5.2). В процессе оценки используется вся доступная информация, т.е. управление u , измерение $y = Cx$ и матрицы системы A, B . Вектор состояния $x(t)$ в любой момент времени неизвестен, в частности, неизвестны начальные условия $x(0)$. Алгоритм для оценки состояния системы называется *наблюдателем*.

Уравнения наблюдателя представляются в виде [4, 5]:

$$\dot{\hat{x}}(t) = (A - KC)\hat{x}(t) + Bu(t) + Ky(t), \quad (5.4)$$

где $\hat{x}(t)$ – оценка состояния системы $x(t)$, а матрица коэффициентов усиления наблюдателя K должна быть выбрана таким образом, чтобы наблюдатель был устойчивым даже при неустойчивом объекте и чтобы качество процесса восстановления (нахождения оценки) было приемлемым.

Характеристическое уравнение наблюдателя (5.4) имеет вид:

$$\det[A - KC - sI] = 0 \quad (5.5)$$

или

$$\begin{vmatrix} -k_1 - s & 1 \\ -k_2 & -s \end{vmatrix} = s^2 + k_1s + k_2 = (s - s_1)(s - s_2) = 0 \quad (5.6)$$

Задав значения корней s_1 и s_2 , можно рассчитать коэффициенты k_1 и k_2 :

$$k_1 = -(s_1 + s_2), \quad k_2 = s_1s_2$$

Рассмотрены два варианта: 1) $s_1 = s_2 = -0.1$; $k_1 = 0.2$; $k_2 = 0.01$;

2) $s_1 = s_2 = -10$; $k_1 = 20$; $k_2 = 100$.

На рис. 5.5 представлена схема моделирования системы 5.1 с наблюдателем для варианта 1, на рис. 5.6 - для варианта 2, а на рис. 5.7 – схема

Compound Block-а наблюдателя Observer для варианта 2.

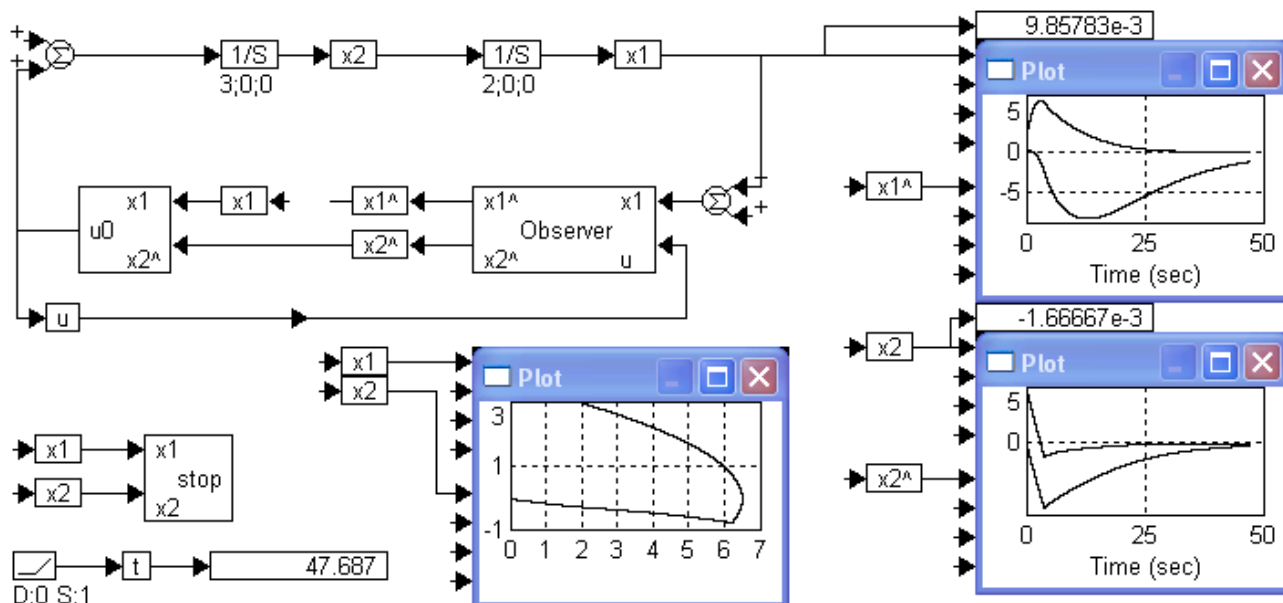


Рисунок 5.5

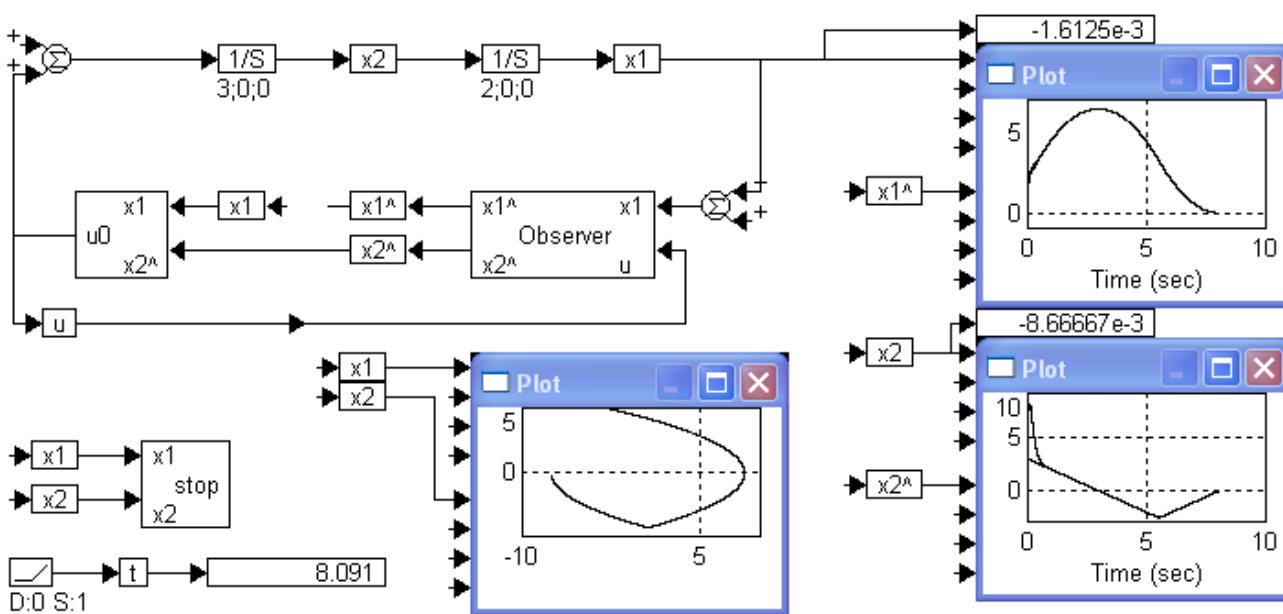


Рисунок 5.6

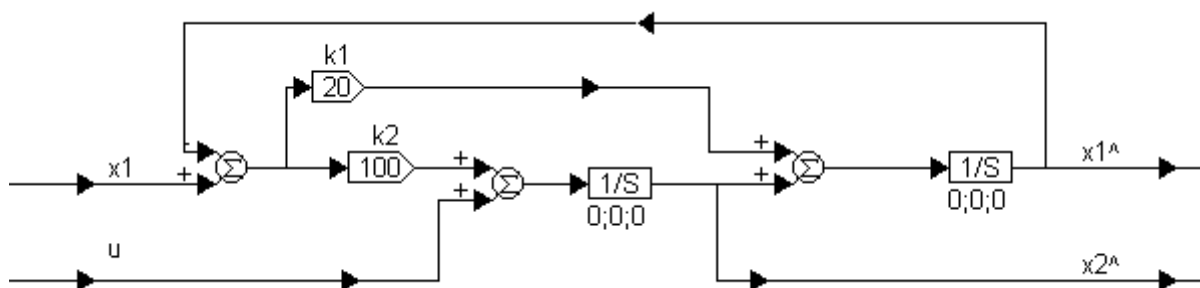


Рисунок 5.7

По результатам моделирования можно сделать вывод, что при больших корнях характеристического уравнения наблюдателя (вариант 2) выходные величины наблюдателя и объекта становятся близкими за 1 с и процесс управления заканчивается за 8,091 с (как и для системы с управлением по полному вектору состояния – рис. 5.1). При малых корнях характеристического уравнения наблюдателя (вариант 1) процесс затягивается и выходные величины наблюдателя и объекта становятся равными за 47,687 с.

На рис. 5.8 и рис. 5.9 представлены системы (рис. 5.5 и рис. 5.6) при действии на вход и выход системы возмущений с нормальным распределением. Для варианта 1 процесс заканчивается за 68,556 с, для варианта 2 – за 19,845 с. Графики изменения переменных идентичны.

Таким образом, использование детерминированной модели в условиях случайных возмущений не вполне эффективно.

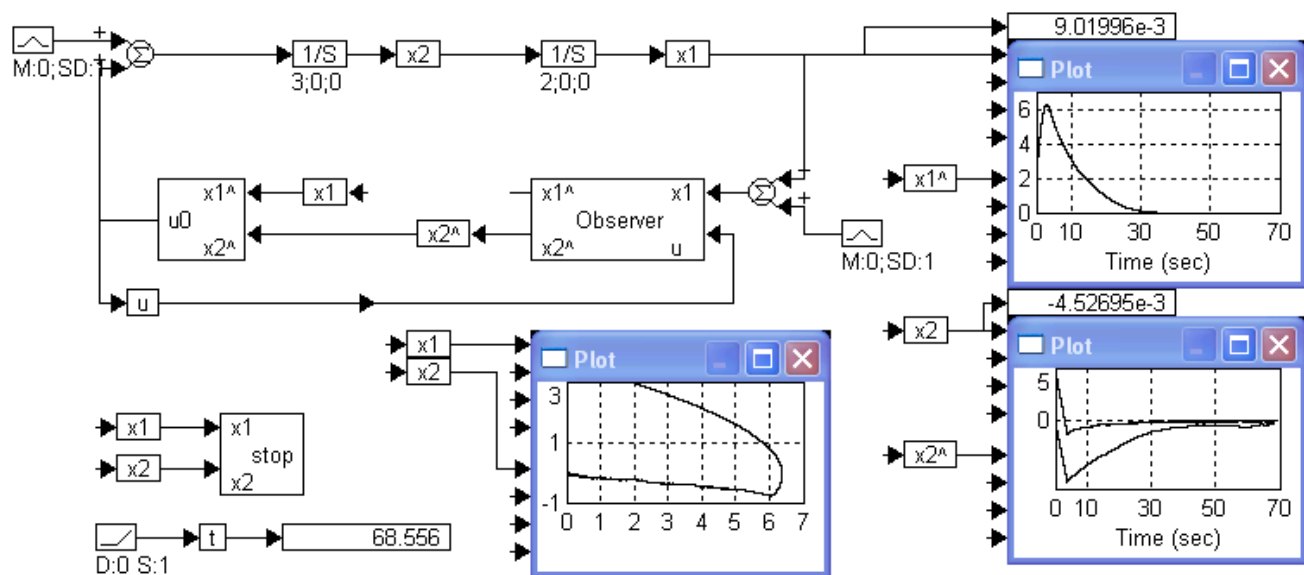


Рисунок 5.8

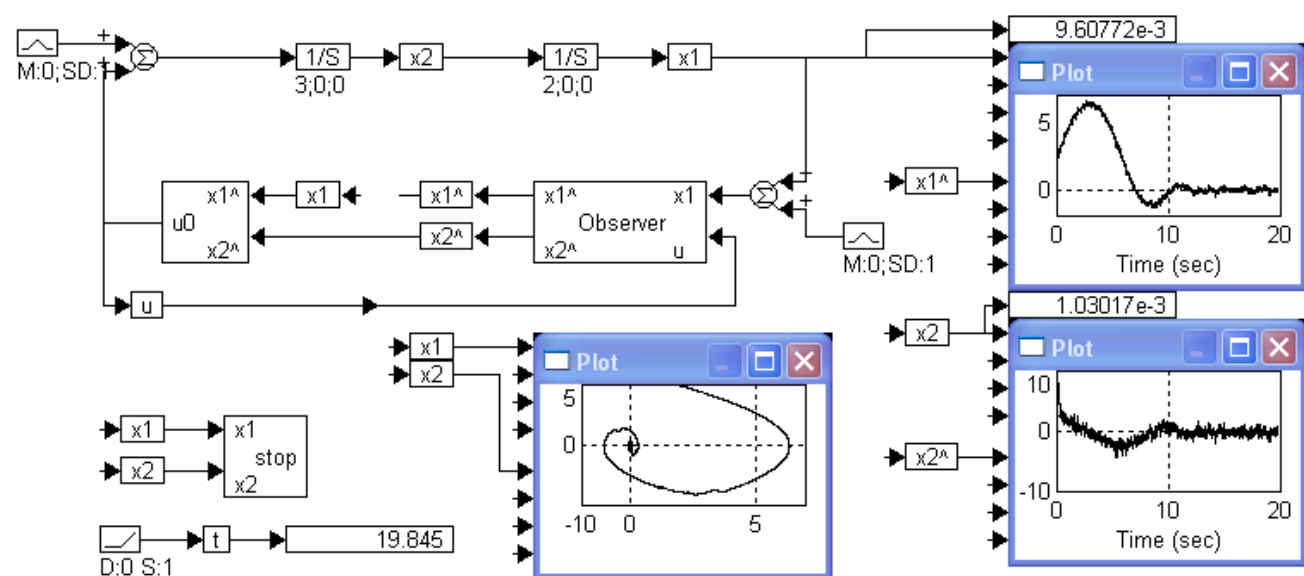


Рисунок 5.9

5.3. Стохастическая система с измерением неполного вектора состояния

Необходимо синтезировать управление, обеспечивающее оптимальное протекание процесса с учетом случайных возмущений и помех.

Движение системы описывается уравнениями

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u + v, \end{aligned} \quad (5.7)$$

где $v = \{0; g\}$ – входной белый шум или возмущение, g – скаляр;

$x(0) = \{m_{x_0}, p_0\}$, p_0 – начальное значение корреляционной матрицы 2×2 ;

$x(T) = (0; 0)$; управление – $|u| \leq 1$; измеренная выходная величина

$z = x_1 + N$, где $N = \{0; q\}$ – выходной шум или помеха измерения.

Задача состоит в том, чтобы перейти из начального состояния в конечное за минимальное время T . Решение ищется в форме синтеза, т.е.

$$u^0(\hat{x}_1, \hat{x}_2) = -[\text{sign}(\hat{x}_1 + \text{sign} \hat{x}_2 \cdot 0.5\hat{x}_2^2)]. \quad (5.8)$$

Для получения \hat{x}_1 и \hat{x}_2 необходимо использовать фильтр Калмана [6]:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu^0 + K(z - C\hat{x}) \\ K &= PC^T Q^{-1} \end{aligned} \quad (5.9)$$

$$\dot{P} = AP + PA^T - PC^T Q^{-1} CP + G$$

и найти \hat{x}_1 и \hat{x}_2 для управления u^0 из уравнений

$$\begin{aligned} \dot{\hat{x}}_1 &= \hat{x}_2 + k_1(z - \hat{x}_1) \\ \dot{\hat{x}}_2 &= u^0 + k_2(z - \hat{x}_1), \end{aligned} \quad (5.10)$$

где $k_1 = \frac{p_{11}}{q}$, $k_2 = \frac{p_{12}}{q}$.

На рис. 5.10 представлена схема моделирования стохастической системы.

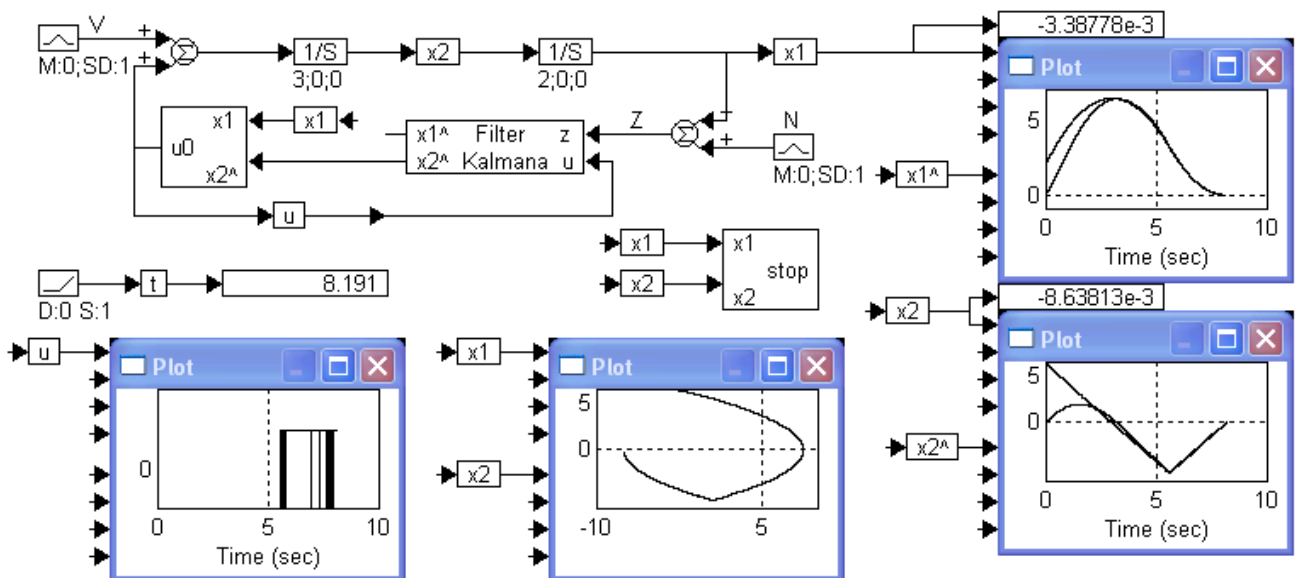


Рисунок 5.10

Схемы Compound Block-ов Filter Kalmana и Find приведены на рис. 5.11 и рис. 5.12 соответственно.

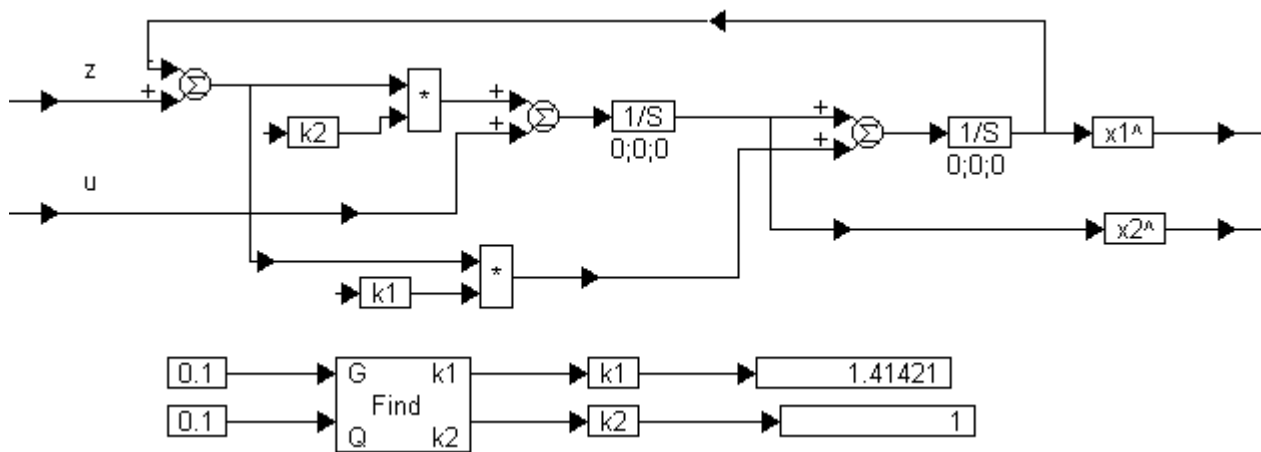


Рисунок 5.11

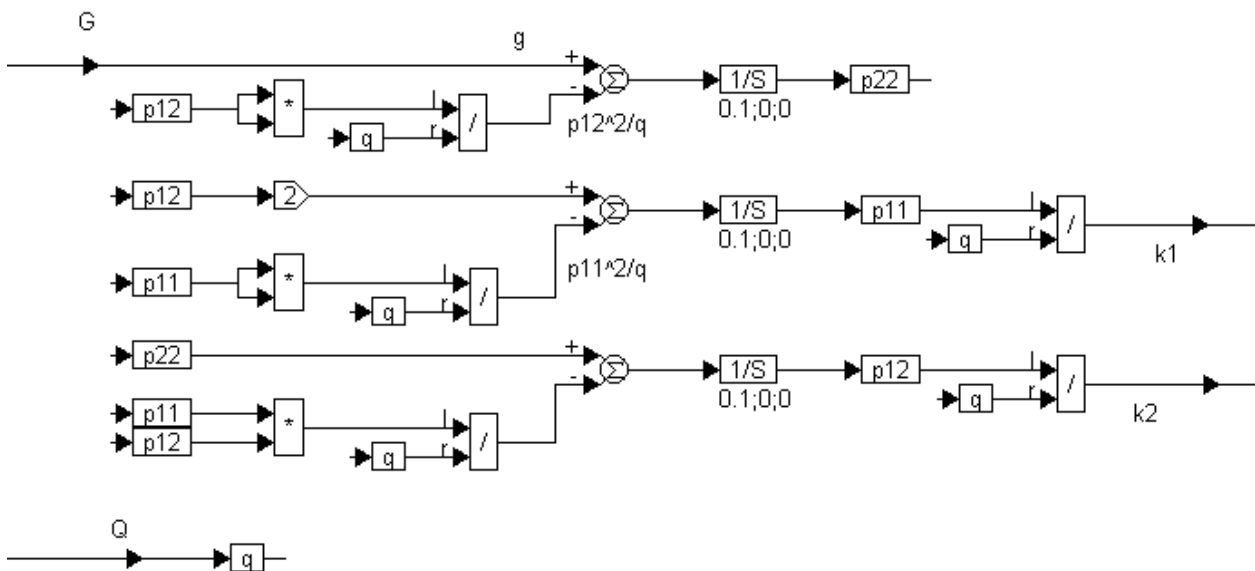


Рисунок 5.12

По результатам моделирования можно сделать вывод, что процесс заканчивается за 8,191 с.

Сравнивая результаты моделирования системы с фильтром Калмана (рис. 5.10) и системы с наблюдающим устройством (рис. 5.9), можно сделать следующие выводы:

1. Использование детерминированной модели с наблюдателем состояния в условии измерения неполного вектора состояния и случайных возмущений не вполне эффективно;

2. Применение фильтра Калмана в этих условиях дает наилучшую оценку переменных и наименьшее время процесса управления.

Контрольные вопросы к лабораторным работам

1. В чем состоит метод управления по старшей производной?
2. Как формируется эталонная модель с заданными динамическими характеристиками?
3. Как вычисляется закон управления по невязке старшей производной и какова роль коэффициента усиления k в законе управления?
4. В чем состоит задача структурного синтеза управлением программным движением системы?
5. Как находится управление для системы отработки заданий, являющихся функциями времени?
6. Как строится управление для воспроизведения заданных траекторий?
7. Представить векторно-матричную модель системы в пространстве состояний.
8. Что такое оптимальное быстродействие и в каком виде задается оптимальное управление?
9. Назначение наблюдателя при управлении детерминированной системой с измерением неполного вектора состояния.
10. Влияние на переходной процесс значений корней характеристического уравнения наблюдателя.
11. Сформулировать задачу оптимального управления стохастической системой.
12. Привести основные уравнения фильтра Калмана.
13. Влияние на эффективность оценивания по фильтру Калмана матриц дисперсий входного (Q) и измерительного (R) шумов.
14. Что такое белый шум? Закон распределения?
15. Объяснить, как строится закон оптимального по быстродействию синтеза $u^0(x_1, x_2)$ в рассмотренной системе.

Список литературы

1. Крутько П.Д. Обратные задачи динамики в теории автоматического управления. Цикл лекций, учеб. пособ. для вузов/ П.Д. Крутько – М.: Машиностроение, 2004. – 576 с.
2. Дьяконов В.П. VisSim+Mathcad+MATLAB. Визуальное математическое моделирование/ В.П. Дьяконов – М.: СОЛОН-Пресс, 2004. – 384 с.
3. Филлипс Ч. Системы управления с обратной связью/ Ч. Филлипс, Р. Харбор – М.: Лаборатория Базовых знаний, 2001. – 615 с.
4. Дорф Р.. Современные системы управления. Пер. с англ./ Р. Дорф, Р. Бишоп – М.: Лаборатория Базовых Знаний, 2002. – 832 с.
5. Фельдбаум А.А. Основы теории оптимальных автоматических систем/ А.А. Фельдбаум – М.: Физматгиз, 1963. – 552 с.
6. Браммер К. Фильтр Калмана – Бьюси/ К. Браммер, Г. Зиффлинг – М.: Наука. Главная редакция физико-математической литературы, 1982. – 200 с.

Приложение 1

Краткое описание пакета прикладных программ VisSim

Пакет VisSim (Visio Simulation) предназначен для решения задач математического моделирования динамических объектов путем составления графической схемы соединений блоков, реализующих различные математические и логические операции.

Блоки, соответствующие различным операциям, собраны в меню Blocks в различных подразделах по группам. Для установки требуемых параметров блока (коэффициентов, начальных условий и др.) нажимают правую клавишу мыши (ПКМ). В открывшемся окне свойств (Properties) в необходимых полях записывают требуемые значения параметров, устанавливая, если нужно, флажки (галочки). Для изменения ориентации входа-выхода блока необходимо вначале выделить этот блок и нажатием клавиш Ctrl+Left (Left – левая клавиша управления курсором на клавиатуре) изменить ориентацию входа-выхода блока.

При соединении двух блоков при помощи связывающей линии, для того, чтобы она не заходила за другие блоки и было понятно, к какому блоку она относится, необходимо использовать промежуточный блок wirePositioner (Blocks, Annotation). Весьма полезным при создании схемы соединений может быть применение блока Variable (переменная), который вызывается из меню Blocks, Annotation. Для соединения блоков достаточно подключить к выходу первого блок Variable, дать ему имя, затем скопировать его и подключить ко входу следующего блока. Таким образом для соединения этих блоков связывающая линия оказывается ненужной. Для нанесения меток на схеме необходимо воспользоваться блоком label (Blocks, Annotation).

При создании сложных схем моделирования для большей наглядности используется CompoundBlock, позволяющий объединить несколько блоков в один укрупненный блок. Для создания CompoundBlock-а необходимо обвести прямоугольником объединяемые блоки, выбрать в меню EDIT, Create CompoundBlock, назначить имя CompoundBlock-а и нажать ОК. В схеме появится CompoundBlock с назначенным именем с указанием его связей с другими блоками. Добавление новых входов и выходов и их удаление осуществляется при помощи построителей $\xrightarrow{+}$ и $\xrightarrow{-}$ из инструментальной строки.

Результаты моделирования представляются в наглядной графической форме с помощью смотрового окна Plot (Blocks, SignalConsumer). В Plot Properties задается сетка (Grid Lines), минимальные и максимальные значения и фиксирование границ изменения переменных (Fixed Bounds), число делений по осям, вид линий и координаты любой точки графика при нажатии кнопки Read Coordinates.

Для моделирования и построения графиков изменения переменных используется меню Simulate, Simulation Properties, где устанавливается шаг, временной интервал и метод интегрирования, а затем зеленой кнопкой в инструментальной строке запускается процесс интегрирования.

Навчальне видання

В. Б. Успенський, Л.В. Шипулина

**МЕТОДИЧНІ ВКАЗІВКИ ДО ЛАБОРАТОРНИХ РОБІТ
ПО КУРСАМ: «ОБЕРНЕНІ ЗАДАЧІ ДИНАМІКИ» ТА
«МЕТОДИ СИНТЕЗУ ТА ОПТИМІЗАЦІЇ СИСТЕМ УПРАВЛІННЯ»**

Навчально-методичний посібник