

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра автоматического управления

М.А. Антипова, М.К. Хаджинов

АВТОМАТИЗИРОВАННЫЙ РАСЧЁТ СИСТЕМ УПРАВЛЕНИЯ

Лабораторный практикум
для студентов специальностей

53 01 03 «Автоматическое управление в технических системах» и
53 01 07 «Информационные технологии и управление в технических системах»
всех форм обучения

Минск 2003

УДК 681.518 (075.8)
ББК 32.965 я 73
А 72

Рецензент:
зав. кафедрой теоретических основ электротехники БГУИР,
д-р техн. наук, проф. Л.Ю. Шилин

Антипова М.А.

А 72 Автоматизированный расчёт систем управления: Лабораторный практикум для студ. спец. 53 01 03 «Автоматическое управление в технических системах» и 53 01 07 «Информационные технологии и управление в технических системах» всех форм обучения / М.А. Антипова, М.К. Хаджинов. – Мн.: БГУИР, 2003. – 40 с.

ISBN 985-444-522-4.

В практикуме приведены описание и порядок выполнения пяти лабораторных работ по курсу «Автоматизированный расчёт систем управления».

В каждой лабораторной работе изложены краткие теоретические сведения по рассматриваемой теме, представлены варианты заданий, предложены методики их выполнения, сформулированы контрольные вопросы. Лабораторные работы проиллюстрированы примерами программ на Matlab.

УДК 681.518 (075.8)
ББК 32.965 я 73

ISBN 985-444-522-4

© Антипова М.А., Хаджинов М.К., 2003
© БГУИР, 2003

СОДЕРЖАНИЕ

1. Лабораторная работа № 1. Программирование в среде MATLAB. Изучение характеристик типовых звеньев ТАУ
2. Лабораторная работа № 2. Построение математических моделей систем автоматического управления
3. Лабораторная работа № 3. Влияние обратной связи на характеристики звеньев
4. Лабораторная работа № 4. Применение последовательной коррекции для достижения необходимых запасов устойчивости
5. Лабораторная работа № 5. Синтез параллельной коррекции по требованиям качества переходного процесса

Литература

Приложение

ПРОГРАММИРОВАНИЕ В СРЕДЕ MATLAB. ИЗУЧЕНИЕ ХАРАКТЕРИСТИК ТИПОВЫХ ЗВЕНЬЕВ ТАУ

Цель работы: научиться составлять программы на языке Matlab в виде m-файлов двух видов (m-функции и script-файла), а также распознавать типовые звенья ТАУ по их переходным и логарифмическим частотным характеристикам.

Краткие теоретические сведения

Matlab является интерактивной системой для выполнения инженерных и научных расчетов, которая ориентирована на работу с массивами данных. Matlab – это одновременно и операционная среда и язык программирования. Пользователь может сам написать специализированные функции и программы, которые оформляются в виде m-файлов. Накопление родственных функций приводит к созданию пакетов прикладных программ, каждый из которых решает определенную задачу или проблему. Наиболее широко Matlab используется в следующих областях: математика и вычисления, разработка алгоритмов, вычислительный эксперимент, имитационное моделирование, анализ данных, исследование и визуализация результатов, научная и инженерная графика, разработка приложений, включая графический интерфейс пользователя.

Операционная среда. Matlab – это множество интерфейсов, которые поддерживают связь этой системы с внешним миром через командное окно, инструментальную панель, подсистемы просмотра рабочей области и путей доступа, редактор (отладчик) m-файлов, специальные меню.

Командное окно позволяет использовать Matlab как мощный научный калькулятор, который отображает символы набранных с клавиатуры команд, результаты их выполнения, текст исполняемой программы, а также информацию об ошибках выполнения программы.

В командной строке после знака приглашения (>>) можно выполнять любые операции с действительными или комплексными числами (простейшие арифметические действия, элементарные и специальные математические функции). Результат вычислений будет представлен в предварительно установленном формате.

Управление путями доступа. Рассмотрим способы включения рабочего каталога, например каталога *work*, в список путей доступа.

В Matlab для поиска m-файлов используют стандартный механизм выбора путей доступа. В процессе сеанса работы можно вывести на терминал или внести изменения в список путей доступа, используя ввод в командную строку следующих функций:

- 1) Addpath/home/lib – добавляет новый каталог в список путей доступа;
- 2) Rmpath/home/lib – удаляет путь из списка;
- 3) Path – выводит на экран список путей доступа;
- 4) Path(s) – заменяет существующий список списком s.

Кроме работы из командной строки существует средство просмотра путей доступа Path Browser, которое позволяет просматривать, модифицировать пути доступа и видеть списки всех файлов системы Matlab. Чтобы открыть средство просмотра, следует использовать опцию Set Path из меню File или кнопку инструментальной панели Path Browser. Все изменения, которые вносятся в список путей доступа, действуют только в течение сеанса работы. Чтобы внести их в файл pathdef.m для постоянного использования, необходимо воспользоваться командой Save Path из меню File. В системе Matlab используется понятие текущего каталога, который определен в файле запуска, для его вывода на экран предназначена команда **cd**. Для изменения текущего каталога следует использовать команду **cd <новый путь доступа>**.

Создание m-файлов в среде Matlab. В языке Matlab имеются программы двух типов с расширением .m:

- script-файлы (файлы-сценарии или управляющие программы);
- файл-функции (процедуры).

При помощи script-файлов оформляются основные программы, управляющие от начала до конца организацией всего вычислительного процесса. Как файл-функции оформляются отдельные процедуры или функции, т.е. такие части программы, которые рассчитаны на неоднократное использование script-файлами или другими процедурами при изменяемых значениях входных параметров и не могут быть выполнены без предварительного задания значений переменных, которые называют входными.

Главным внешним отличием текстов этих двух видов файлов является то, что файл-функции имеют первую строку вида:

function [перечень выходных величин] = имя процедуры (перечень входных величин).

Например, функция, вычисляющая значения квадратичного полинома, будет выглядеть следующим образом:

```
function [y]=example(x,a,b,c)
% Функция, вычисляющая значения квадратичного полинома
% Формат вызова: example(x,a,b,c)
y = a.*x.*x+b.*x+c;
```

В script-файлах такой строки нет, но принципиальное отличие заключается в совершенно разном восприятии системой имен переменных в этих файлах. В файлах-функциях все имена переменных внутри файла и в заголовке воспринимаются как локальные, а переменные script-файлов образуют так

называемое рабочее пространство и сохраняют свой смысл и значения в течение всего сеанса работы с системой.

Текст программ. Перечислим основные особенности записи текста программы на языке Matlab:

- каждый оператор записывается в отдельной строке и вводится в программу при нажатии клавиши Enter; несколько операторов в одной строке разделяются символами “ ; ” или “ , ”; длинный оператор можно записать в несколько строк, используя знак переноса – три точки (...);

- если оператор не заканчивается символом “ ; ”, то результат его действия при выполнении программы будет выведен в командное окно;

- строка программы или её часть, начинающаяся с символа “ % ”, не выполняется, она воспринимается системой как комментарий;

- строки комментария, предшествующие первому выполняемому оператору, воспринимаются как описание программы и выводятся в командное окно по команде **help** <имя файла>;

- операторы начала и окончания текста программы отсутствуют, т.е. начало и конец программы никак не маркируются;

- переменные не описываются и не объявляются, любое новое имя воспринимается системой как имя матрицы, размер которой устанавливается при предварительном вводе значений ее элементов;

- имена переменных могут содержать лишь буквы латинского алфавита или цифры и должны начинаться с буквы. Общее число символов – не более 19. В именах переменных могут использоваться как прописные, так и строчные буквы с учетом того, что система Matlab их различает.

Создание программы на языке Matlab осуществляется при помощи текстового редактора, который вызывается автоматически при открытии m-файлов командами New, Open из меню File. Запуск m-файлов на выполнение производится из командной строки по имени файла.

Получение справочной информации (команды работы с Help).

Получить информацию о функциях Matlab можно следующими способами:

- 1) командой **help**;
- 2) командой **lookfor**;
- 3) меню help;
- 4) просмотром и выводом на печать страниц документации с помощью средства Adobe's Acrobat;
- 5) с помощью других справочных команд.

Список каталогов выводит на экран команда **help** без аргументов. Список команд каталога выводится на экран командой **help** <имя каталога>. Основной и наиболее быстрый способ выяснить синтаксис и особенности применения m-функции – это использовать команду **help**<имя m-функции>.

Команда **lookfor** позволяет выполнить поиск m-функции по ключевому слову. Она выводит первую строку комментария m-функции, если в ней встретилось ключевое слово, или все строки комментария при использовании опции – **all**.

Меню Help позволяет активизировать следующие окна:

- 1) Help Window – окно справки;
- 2) Help Tips – окно справки для получения подсказки;
- 3) Help Desk(HTML) – доступ к справочным системам на жестком диске;
- 4) Examples and Demos – окно демонстрационной подсистемы;
- 5) About Matlab – информация об установленной версии;
- 6) Join Matlab Access – подписка на пользование в сети Internet системой Matlab.

Adobe's Acrobat позволяет просматривать текст в формате печатной страницы, с полным набором шрифтов графики и изображений и выводить его на печать.

К справочным командам относятся:

- **what** – вывод списка файлов;
- **which** – справка о пути доступа к функции или файлу;
- **demo, info, ver, helpinfo**.

Задание к лабораторной работе

1. С помощью предложенных возможностей работы со справочной системой Matlab ознакомиться с использованием следующих операторов: **function, tf, step, bode, menu, case, switch, break, if, while, for, end**.

2. Создать m-файлы двух видов: script-файл и файл-функцию. Script-файл может содержать текст демонстрационного примера или собственной простой программы на языке Matlab.

3. Файл-функция должна отражать возможность работы с входными и выходными переменными, выбор её является произвольным. Рекомендуется набрать текст рассмотренного выше примера вычисления квадратичного полинома, разобраться с ним, а затем написать свою аналогичную функцию.

4. Записать в tf-форме передаточные функции следующих звеньев:

- интегрирующего $W(s) = \frac{K}{s}$;
- дифференцирующего $W(s) = Ks$;
- усилительного $W(s) = K$;
- апериодического 1-го порядка $W(s) = \frac{K}{Ts + 1}$;

- апериодического 2-го порядка

$$W(s) = \frac{K}{(T_1s + 1)(T_2s + 1)};$$

- колебательного

$$W(s) = \frac{K}{T^2s^2 + 2\xi Ts + 1};$$

- консервативного

$$W(s) = \frac{K}{Ts^2 + 1};$$

- интегрирующего с запаздыванием

$$W(s) = \frac{K}{s(Ts + 1)};$$

- форсирующего

$$W(s) = K(Ts + 1).$$

Параметры звеньев K , T , ξ записываются в буквенном представлении в начале программы, чтобы иметь возможность их быстрого изменения. Значения параметров выбираются из следующих диапазонов:

$$K = 5 \dots 100; \quad T = 0,01 \dots 10 \text{ с}; \quad \xi = 0,05 \dots 1.$$

5. Используя операторы управления вычислительным процессом (условные, циклические, операторы выбора), оформить лабораторную работу в виде стандартного меню (**menu**), которое позволяло бы выбирать одно из элементарных звеньев ТАУ и строить его переходные, логарифмические и импульсные характеристики. Также реализовать возможность с помощью меню подключать ранее созданные script-файл и файл-функцию.

6. Проанализировать, как влияет изменение параметров звена (выбор звена определяется преподавателем) на вид переходной и частотной характеристик. Для этого необходимо наблюдать, как изменяется вид характеристик при различных K , T , ξ , выбирая различные значения параметров в цикле, как это делается в представленной ниже программе.

7. Построить графики зависимости длительности переходного процесса $t_{n,n}$ от изменения коэффициента усиления системы K и постоянной времени T .

8. Записать формулы для расчета ЛАЧХ и ФЧХ, построить асимптотические характеристики.

9. Разделить исследованные звенья на дифференцирующие, позиционные и интегрирующие.

10. Проверить связи между переходной $h(t)$, частотной $L(\omega)$ характеристиками и передаточной функцией $W(s)$ одного и того же объекта, сформулированные в теореме о предельных значениях. Согласно этой теореме,

$$\lim_{t \rightarrow \infty} h(t) = \lim_{w \rightarrow 0} L(w) = \lim_{s \rightarrow 0} W(s); \quad (1)$$

$$\lim_{t \rightarrow 0} h(t) = \lim_{w \rightarrow \infty} L(w) = \lim_{s \rightarrow \infty} W(s). \quad (2)$$

Для выполнения задания № 6 воспользуйтесь в качестве базовой следующей программой:

```
% Программа ar2d моделирует колебательное звено с изменяющимся
% коэффициентом демпфирования,
% выводит циклы переходных и частотных характеристик,
% преобразует непрерывную модель в дискретную, причем изменяется
% Ts – период дискретизации.
% Для исследования непрерывной модели необходимо закрыть строку
% формирования признака для преобразования в дискретный объект.
clear,close all; % Очистить память. Закрыть все окна.
k=25; T=2; c='s'; % Параметры и признак непрерывного объекта.
c='z'; % Признак для преобразования в дискретный объект.
txt1=['Колебат.зв. W(' ,c,') = (']; % Часть заголовка.
for i=-6:0; % Цикл по переменной i.
m=2^(i/2); % m – изменяемый в цикле множитель для W(s).
ch=[k]; % Числитель передаточной функции W(s).
zn=[T^2 2*T*m 1]; % Знаменатель передаточной функции W(s).
sk=tf(ch,zn); % Передаточная функция W(s).
if c=='z'; % Действия, если задан признак преобразования в
% дискретный объект.
Ts=m*T*2; % Вычисляется период дискретизации Ts.
sk=c2d(sk,Ts); % Объект sk преобразуется в дискретный.
[ch,zn]=tfdata(sk,'v'); % Числитель и знаменатель дискретного
% объекта.
end % Конец преобразования в дискретный объект.
txt2=[txt1,poly2str(ch,c),')/(' ,poly2str(zn,c),')']; % Заголовок.
figure(1); % Открыть окно № 1.
hold on;%Открыть режим сохранения графиков в окне № 1.
step(sk,T*9); % Добавить переходную функцию объекта sk.
title([txt2]); % Вставить заголовок.
hold off; % Закрыть режим сохранения графиков в окне № 1.
pause(0), % Сделать паузу, чтобы нарисовать график.
figure(2); % Открыть окно 2.
hold on; % Открыть режим сохранения прежних графиков в окне 2.
bode(sk); % Добавить частотные характеристики объекта sk.
title([txt2]); % Вставить заголовок.
hold off; % Закрыть режим сохранения прежних графиков в окне 2.
pause, % Сделать паузу, чтобы нарисовать график.
end; % Продолжать до окончания цикла по переменной i.
zoom on; % Включить режим масштабирования графика в окне 2.
```

Контрольные вопросы

1. Чем отличаются друг от друга переменные script-файла и m-функции в использовании рабочего пространства системы?
2. Какими способами можно воспользоваться для получения справочной информации?

3. Объяснить физический смысл переходной характеристики, передаточной функции, амплитудно- и фазо-частотной характеристик.
4. Что такое коэффициент затухания, как он влияет на вид переходной характеристики колебательного звена?
5. Какие звенья являются частным случаем колебательного звена и при каких значениях параметров?
6. Что такое физическая реализуемость звеньев? Сформулируйте условие физической реализуемости звена для его передаточной функции.
7. Как реализуются и используются дифференцирующие звенья на практике?
8. Как определить передаточную функцию звена по виду его ЛАЧХ?
9. Как строятся ЛАЧХ элементарных звеньев по передаточным функциям?
10. Как различаются ЛАЧХ непрерывных и дискретных звеньев?
11. При разделении звеньев на дифференцирующие, позиционные и интегрирующие начальная или конечная части характеристик step , impulse определяет тип звена?
12. На низких или высоких частотах характеристик bode , nyquist определяется тип звена?

ПОСТРОЕНИЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Цель работы: научиться создавать математические модели непрерывных и дискретных систем управления с постоянными параметрами в различных формах представления и записывать их с помощью пакета прикладных программ (ППП) Control System Toolbox.

Краткие теоретические сведения

Формы представления математических моделей систем управления. Модели динамических систем могут быть эмпирическими, связывающими непосредственно выходные и входные переменные, а также моделями в пространстве состояний, использующими переменные состояния как промежуточные между входом и выходом. Переменные состояния обычно отражают внутренние процессы, для описания которых применяются законы физики, механики и других естественных наук. В ППП Control System Toolbox для описания моделей динамических систем введен новый класс объектов – LTI-объекты, т.е. линейные объекты с постоянными параметрами. Control System Toolbox обеспечивает создание структур данных для каждой из моделей, называемых соответственно tf-, zpk- или ss-подклассами класса LTI-объектов. Эти три подкласса могут быть описаны одним типом данных – массивом ячеек, что позволяет манипулировать линейными системами как единым объектом, а не наборами данных в виде векторов или матриц.

Для пользователя zpk-объекты наиболее удобны и наглядны, но на ЭВМ ss-объекты просчитываются точнее. Поэтому в MATLAB принята нижеследующая иерархия объектов LTI-класса. Операции, в которых в качестве операндов используются объекты двух или более подклассов, будут иметь результатом:

объект подкласса ss, если по крайней мере один операнд принадлежит подклассу ss;

объект подкласса zpk, если отсутствуют операнды подкласса ss и по крайней мере один из операндов принадлежит подклассу zpk;

объект подкласса tf, если все операнды относятся к подклассу tf.

Tf-объект. Одномерная передаточная функция $h(s) = \text{num}(s)/\text{den}(s)$ задается многочленом числителя num и многочленом знаменателя den. В системе Matlab многочлены представляются как векторы-строки, составленные из коэффициентов многочлена в порядке убывания степеней переменной. Например, многочлену s^2+3s+5 соответствует вектор [1 3 5]. Если заданы векторы num и den, соответствующие многочленам числителя и знаменателя, то функция **h = tf(num, den)** создает LTI-модель одномерной системы в виде

передаточной функции $h(s) = n(s)/d(s)$. Переменная h является объектом подкласса tf , содержащим данные о числителе и знаменателе передаточной функции.

Zpk-объект. Модели одномерных систем подкласса zpk задаются выражением

$$W(s) = K \frac{(s - z_1)(s - z_2) \dots (s - z_m)}{(s - p_1)(s - p_2) \dots (s - p_n)}, \quad (3)$$

где z_1, z_2, \dots, z_m – нули системы;

p_1, p_2, \dots, p_n – полюсы системы;

K – обобщенный коэффициент передачи.

Функция, предназначенная для формирования таких моделей, имеет вид $h = zpk(z, p, K)$, где z и p – векторы из нулей и полюсов, а K – обобщенный коэффициент передачи. Она создает объект h подкласса zpk .

Ss-объект. Для описания динамических систем в пространстве состояний применяются модели подкласса ss , которые основаны на линейных дифференциальных или разностных уравнениях.

Модель непрерывной системы в подклассе ss имеет вид

$$\begin{cases} dx/dt = Ax + Bu, \\ y = Cx + Du, \end{cases} \quad (4)$$

где x – вектор состояния;

u – вектор входа;

y – вектор выхода.

Для формирования моделей в подклассе ss предназначена функция $sys = ss(A, B, C, D)$. В результате ее применения получаем описание ss -объекта в виде четверки матриц $\{A, B, C, D\}$, которые должны иметь согласованные размеры. Если n – число переменных состояния, p – число входов и m – число выходов, то четверка матриц должна иметь следующие размеры: $\{A_{n \times n}, B_{n \times p}, C_{m \times n}, D_{m \times p}\}$.

Для моделей с нулевой матрицей D можно использовать присваивание $D = 0$ как краткую форму записи нулевой матрицы соответствующих размеров.

Дискретные LTI-объекты. Переход от непрерывной модели Sn к дискретной Sd и обратно с периодом дискретизации Ts производится операторами $c2d$ и $d2c$:

$$Sd = c2d(Sn, Ts); \quad (5)$$

$$Sn = d2c(Sd, Ts). \quad (6)$$

Если к входным аргументам функций `tf`, `zpk`, `ss`, соответствующим непрерывной модели, добавить период дискретности T_s , то сформируется дискретная модель с совершенно нелепыми параметрами, т.е. переустановка значения периода дискретности с нулевого на ненулевое значение, характерное для дискретных систем, не влечет за собой осмысленное построение дискретной модели.

Принято, что период дискретности для непрерывной системы равен нулю. Значение $T_s = -1$ соответствует случаю, когда период дискретности не специфицирован.

Извлечение числовых параметров LTI-моделей. Описанные выше функции `tf`, `zpk` и `ss` формируют информацию об используемой модели и периоде дискретности в единый LTI-объект. Извлечь эти данные из описания существующего lti-объекта позволяют следующие команды:

```
[num, den, Ts] = tfdata(sys, 'v');  
[z, p, k, Ts] = zpkdata(sys, 'v');  
[a, b, c, d, Ts] = ssdata(sys).
```

Выходные аргументы `num`, `den` оператора `tfdata` и `z`, `p`, `k` оператора `zpkdata` всегда являются массивами ячеек. Они имеют число строк, равное числу выходов, и число столбцов, равное числу входов, а их элементы h_{ij} определяют передаточную функцию к i -му выходу от j -го входа.

LTI-объекты включают данные о модели, периоде дискретности, а также могут содержать дополнительную информацию, такую как имена входов и выходов, примечания об истории модели. Различают родовые свойства, которые являются общими для всех трех подклассов объектов (`Input Name`, `Notes`, `Output Name`, `Ts`, `Td`), и специфические свойства, которые относятся только к одному подклассу моделей. Каждое свойство задается парой аргументов: свойство (`Property Name`), значение (`Property Value`). Значения свойств системы можно определить с помощью команды **get**.

Существуют три способа установки значений свойств объекта:

- при создании LTI-объектов с помощью команд **tf**, **zpk**, **ss**;
- изменение значений свойств существующей LTI-модели командой **set**;
- присваивание значений элементам структуры.

Узнать информацию о количестве входов и выходов системы можно с помощью функции **size** (имя системы).

Задание к лабораторной работе

1. Выбрать в Приложении, согласно варианту, значения перечисленных параметров двигателей постоянного тока и записать их в программу:

- а) P_H , Вт – мощность двигателя;
- б) w_H , рад/с – номинальная угловая скорость вращения ротора двигателя;
- в) U_H , В – номинальное напряжение, приложенное к цепи якоря;
- г) I_H , А – номинальный ток в цепи якоря;

- д) $M_H, \text{Н}\cdot\text{м}$ – момент нагрузки на валу двигателя;
- е) $J_{\text{ДВ}} \cdot 10^{-4}, \text{кг}\cdot\text{м}^2$ – момент инерции двигателя;
- ж) $R_{\text{я}}, \text{Ом}$ – активное сопротивление в цепи якоря;
- з) $L_{\text{я}}, \text{мГн}$ – индуктивность в цепи якоря.

2. Записать программу расчёта $T_{\text{я}}, T_{\text{М}}, C_{\text{Е}}, C_{\text{М}}$ по приведенным ниже формулам:

$$T_{\text{я}} = L_{\text{я}} / R_{\text{я}} - \text{постоянная времени якорной цепи};$$

$$T_{\text{М}} = (R_{\text{я}} J) / (C_{\text{Е}} C_{\text{М}}) - \text{электромеханическая постоянная двигателя};$$

$$C_{\text{Е}} = (U_{\text{Н}} - I_{\text{Н}} R_{\text{я}}) / \omega_{\text{Н}} - \text{коэффициент ЭДС двигателя};$$

$$C_{\text{М}} = M_{\text{Н}} / I_{\text{Н}} - \text{коэффициент момента двигателя}.$$

3. Построить схему моделирования двигателя на основе системы дифференциальных уравнений, записанных по закону Ома для якорной цепи и закону Ньютона для вращательного движения:

$$\begin{cases} U = RI + L \dot{I} + C_{\text{Е}} \omega \\ J \dot{\omega} = C_{\text{М}} I - M_{\text{Н}} \end{cases} \quad (7)$$

4. Записать по схеме моделирования уравнения в пространстве состояний:

$$\begin{cases} \dot{X}(t) = AX(t) + BU(t), \\ Y(t) = CX(t) + DW(t), \end{cases} \quad (8)$$

где $X(t)$ – вектор состояния,

$U(t)$ – вектор входа,

$Y(t)$ – вектор выхода,

$W(t)$ – вектор возмущающих воздействий,

$A (n \times n)$ – матрица состояния, n – количество переменных состояния,

$B (n \times m)$ – матрица входа, m – количество входов,

$C (r \times n)$ – матрица выхода, r – количество выходов,

$D (l \times n)$ – матрица возмущающих воздействий, где l – количество возмущающих воздействий в системе.

Для того чтобы правильно составить схему моделирования, выбираем две переменные состояния X_1 и X_2 , которые обозначаем:

$$X_1 = I; \quad (9)$$

$$X_2 = \omega. \quad (10)$$

Входным воздействием, согласно схеме, является напряжение $U(t) = U$, а в качестве выходных выбираем:

а) $Y_1(t) = U_{\text{ш}}$, т.е. выход по току, который снимается на $R_{\text{ш}} = 0,01R_{\text{я}}$;

б) $Y_2(t) = U_{\text{ТГ}}$, т.е. выход по скорости, которая измеряется с помощью тахогенератора с $K_{\text{ТГ}} = 5$.

Момент нагрузки двигателя $M_{\text{н}}$ рассматривается в качестве внешнего возмущающего воздействия $W(t)$.

По составленной таким образом схеме моделирования записываем уравнения состояния для X_1 и X_2 и по уравнениям формируем матрицы A , B , C , D .

5. Сформировать модель двигателя в ss-форме и записать ее в программу.

6. По схеме моделирования можно также записать передаточные функции системы по току и по скорости. Для этого используют формулу записи передаточной функции замкнутой системы:

$$W_{3C} = \frac{1/W_{OC}}{1 + 1/(W_{\text{ПЦ}}W_{OC})}, \quad (11)$$

где W_{3C} – передаточная функция замкнутой системы,

W_{OC} – передаточная функция звеньев в обратной связи,

$W_{\text{ПЦ}}$ – передаточная функция звеньев прямой цепи,

$W_{\text{ПЦ}}W_{OC}$ – передаточная функция разомкнутой системы.

Полученные передаточные функции необходимо преобразовать к стандартному виду записи колебательного звена с постоянными времени $T_{\text{я}}$ и $T_{\text{м}}$ и записать в программу в виде tf-моделей передаточных функций двигателя по току и по скорости W_{UI} и $W_{U\&}$

7. Общая передаточная функция двигателя получается в результате вертикальной конкатенации, т.е. преобразования ранее полученных передаточных функций к следующему виду:

$$W_{\text{ДВ}} = [W_{UI}; W_{U\&}]. \quad (12)$$

8. Представить модель двигателя в zpk-форме путем преобразования из ss-модели.

9. Получить дискретные модели двигателя путем преобразования их из непрерывных (s2d) с периодами дискретизации $T_1 = 0,01$ с и $T_2 = 0,002$ с.

10. Обозначить все входы (InputName) и выходы (OutputName) модели, её название в примечании (Notes) и сведения об авторе (UserData) с помощью оператора set или путем явного указания, используя имена необходимых свойств.

11. Оформить результаты работы в виде пользовательского меню, позволяющего осуществлять вывод в командное окно любой из созданных

моделей (в ss-, zpk-, tf-формах), с просмотром их родовых и специфических свойств, а также с построением переходных процессов (step) и логарифмических частотных характеристик (bode, margin). Для zpk-формы представить карту нулей и полюсов – pzmap.

12. Сравнить полученные характеристики и сделать выводы.

Контрольные вопросы

1. Какие преимущества дает принадлежность всех моделей объектов к одному LTI классу?

2. Какие специфические и родовые свойства tf-, zpk- и ss-моделей вы можете назвать?

3. Какие способы установки свойств объектов вы знаете?

4. Как называются и как формируются матрицы A, B, C, D для ss-моделей?

5. В каком месте плоскости pzmap искать признаки типа звена?

6. Каковы признаки типа звена в zpk-модели?

7. Пояснить на примере составления уравнений в пространстве состояний для исследуемого двигателя, каким образом формируются матрицы A, B, C, D по схеме моделирования.

8. Как получить дискретную модель объекта с неспецифицированным периодом дискретизации? Как изменить период дискретизации?

9. Какова иерархия объектов LTI-класса? Почему?

10. Как получить передаточную функцию двигателя по току и по скорости, используя полученную схему моделирования?

11. Как создается схема моделирования на основе дифференциальных уравнений, описывающих динамику системы?

12. Объясните физический смысл переходных характеристик двигателя по току и по скорости.

13. Как влияют на характеристики двигателя постоянные времени T_j и T_M ? Как они определяются?

ВЛИЯНИЕ ОБРАТНОЙ СВЯЗИ НА ХАРАКТЕРИСТИКИ ЗВЕНЬЕВ

Цель работы: изучить механизм влияния обратных связей; основные виды корректирующих обратных связей; научиться оценивать их влияние на динамику замкнутой цепи; осуществлять расчет корректирующей обратной связи в простейших случаях.

Краткие теоретические сведения

Основные виды корректирующих обратных связей. На ЛАЧХ разомкнутого контура с передаточной функцией $W_{\text{ПЦ}}W_{\text{ОС}}$ можно выделить полосу пропускания контура, где $W_{\text{ПЦ}}W_{\text{ОС}} > 1$, и, согласно формуле

$$W_{\text{ЗС}} = \frac{1/W_{\text{ОС}}}{1 + 1/(W_{\text{ПЦ}}W_{\text{ОС}})}, \quad (13)$$

модель замкнутого контура совпадает с $1/W_{\text{ОС}}$, т.е. с обратной величиной обратной связи. За пределами полосы пропускания обратная связь фактически не работает и модель замкнутого контура совпадает с $W_{\text{ПЦ}}$.

ЛАЧХ замкнутого контура можно получить, выбрав на каждой частоте наименьшую из ЛАЧХ передаточных функций $W_{\text{ПЦ}}$ и $1/W_{\text{ОС}}$. В точках пересечения ЛАЧХ возможны резонансные пики и даже потеря устойчивости, так как это точки частот среза разомкнутого контура. Устойчивость контура гарантируется, если в точках пересечения разница наклонов ЛАЧХ равна единице. Если в точках пересечения разница наклонов ЛАЧХ больше двух, гарантируется неустойчивость контура.

Точно судить об устойчивости контура можно по разнице фазовых характеристик $W_{\text{ПЦ}}$ и $1/W_{\text{ОС}}$, которая не должна превышать 180° .

Эффективность обратной связи принято оценивать её глубиной. Хотя точного определения глубины не существует, обычно её связывают с величиной уменьшения коэффициента передачи под действием обратной связи, т.е. глубина обратной связи определяется уровнем ЛАЧХ разомкнутого контура.

Основные виды корректирующих обратных связей следующие:

а) жесткая обратная связь

$$W_{\text{ОС}} = k_{\text{ОС}}; \quad (14)$$

б) инерционная жесткая обратная связь

$$W_{OC} = \frac{k_{OC}}{T_{OC}s + 1}; \quad (15)$$

в) гибкая обратная связь

$$W_{OC} = k_{OC}s; \quad (16)$$

г) инерционная гибкая обратная связь

$$W_{OC} = \frac{k_{OC}s}{T_{OC}s + 1}. \quad (17)$$

Возможны и более сложные передаточные функции корректирующих обратных связей.

Положительная жесткая обратная связь. Пусть она охватывает апериодическое звено, т.е.

$$W_0 = \frac{k}{Ts + 1}, \quad W_{OC} = k_{OC}. \quad (18)$$

Тогда общая передаточная функция будет

$$W(s) = \frac{W_0(s)}{1 - W_{OC}(s)W_0(s)} = \frac{k}{Ts + 1 - kk_{OC}} \quad (19)$$

или

$$W(s) = \frac{k_1}{T_1s + 1}, \quad (20)$$

где

$$k_1 = \frac{k}{1 - kk_{OC}}, \quad T_1 = \frac{T}{1 - kk_{OC}}.$$

Следовательно, положительная обратная связь может служить для увеличения коэффициента усиления. Но надо иметь в виду, что одновременно с этим увеличивается и постоянная времени, т.е. инерционность звена, а при $kk_{OC} > 1$ звено становится неустойчивым.

Отрицательная жесткая обратная связь. При охвате этой связью апериодического звена получаем

$$W(s) = \frac{k}{Ts + 1 + kk_{OC}} = \frac{k_1}{T_1s + 1}, \quad (21)$$

где

$$k_1 = \frac{k}{1 + kk_{OC}}, \quad T_1 = \frac{T}{1 + kk_{OC}}.$$

Следовательно, отрицательная жесткая обратная связь уменьшает инерционность звена. Тем самым она улучшает качество переходного процесса в системе и может оказывать стабилизирующее действие, т.е. превращать неустойчивую замкнутую систему в устойчивую (подобно введению производной). Уменьшение же коэффициента усиления k_1 при этом может быть скомпенсировано за счет других звеньев системы.

При охвате интегрирующего звена отрицательной жесткой обратной связью, т.е. при

$$W_0(s) = \frac{k}{s}, \quad W_{OC} = k_{OC}, \quad (22)$$

получаем

$$W(s) = \frac{W_0(s)}{1 + W_{OC}(s)W_0(s)} = \frac{k}{s + kk_{OC}} = \frac{k_1}{T_1s + 1}, \quad (23)$$

где

$$k_1 = \frac{1}{k_{OC}}, \quad T_1 = \frac{1}{kk_{OC}}.$$

Видно, что под действием жесткой обратной связи теряется интегрирующее свойство звена и оно превращается в апериодическое с коэффициентом усиления, который целиком определяется обратной величиной обратной связи. Постоянная времени T_1 будет мала при большом коэффициенте усиления звена k .

Указанный способ введения обратной связи применяется для точного управления, например в приводных устройствах, чтобы сделать угол поворота выходного вала пропорциональным управляющему сигналу (напряжению).

Инерционная жесткая обратная связь. При охвате этой связью интегрирующего звена

$$W(s) = \frac{k}{s}, \quad W_{OC} = \frac{k_{OC}}{T_{OC}s + 1} \quad (24)$$

приходим к следующему выражению:

$$W(s) = \frac{k(T_{OC}s+1)}{T_{OC}s^2 + s + kk_{OC}} = \frac{k_1(T_{OC}s+1)}{T_1^2s^2 + T_2s+1}, \quad (25)$$

где

$$k_1 = \frac{1}{k_{OC}},$$
$$T_1^2 = \frac{T_{OC}}{kk_{OC}},$$
$$T_2 = \frac{1}{kk_{OC}}.$$

Следовательно, в данном случае интегрирующее звено превращается в звено второго порядка с введением производной. При этом коэффициент усиления k_1 и интенсивность введения производной T_{OC} целиком определяются обратной связью, а первичный коэффициент усиления звена k влияет на новые постоянные времени T_1 и T_2 , которые будут тем меньше, чем больше k . Поэтому при большом k охват интегрирующего звена инерционной жесткой обратной связью эквивалентен усилительному звену с введением производной. При этом

$$W(s) \approx \frac{T_{OC}s+1}{k_{OC}}, \quad (26)$$

т.е. обратной величине обратной связи.

Отсюда вытекает и хорошее влияние инерционной обратной связи на качество переходного процесса в системе в целом.

Гибкая обратная связь. При охвате ею колебательного звена, т.е.

$$W_0(s) = \frac{k}{T^2s^2 + 2\chi Ts + 1}, \quad W_{OC}(s) = k_{OC}s, \quad (27)$$

имеем

$$W(s) = \frac{k}{T^2s^2 + 2\chi_1 Ts + 1}, \quad (28)$$

где

$$2\chi_1 Ts = 2\chi Ts + kk_{OC}s; \quad \chi_1 = \chi + \frac{kk_{OC}}{2T}.$$

Как видно, в этом случае увеличивается демпфирование колебательного звена (ибо $\xi_1 > \xi$), причем не меняется коэффициент усиления. Процесс становится менее колебательным и может превратиться в аperiodический (если $\xi_1 \geq 1$).

Заметим, что охватывать аperiodическое звено гибкой обратной связью нет смысла, так как это только увеличит его инерционность (постоянную времени).

При охвате инерционного интегрирующего звена гибкой обратной связью, т.е.

$$W_0(s) = \frac{k}{s(Ts+1)}, \quad W_{OC} = k_{OC}s, \quad (29)$$

имеем

$$W(s) = \frac{k}{s(Ts+1) + kk_{OC}s} = \frac{k_1}{s(T_1s+1)}, \quad (30)$$

где

$$k_1 = \frac{k}{1 + kk_{OC}},$$

$$T_1 = \frac{T}{1 + kk_{OC}}.$$

Как видно, сохраняется тот же тип интегрирующего звена, но с уменьшенной инерционностью.

Инерционная гибкая обратная связь. При охвате ею инерционного интегрирующего звена, т.е.

$$W_0(s) = \frac{k}{s(Ts+1)}, \quad (31)$$

$$W_{OC}(s) = \frac{k_{OC}s}{T_{OC}s+1}, \quad (32)$$

имеем

$$W(s) = \frac{k(T_{OC}s+1)}{s(TT_{OC}s^2 + (T_1 + T_{OC})s + 1 + kk_{OC})} = \frac{k_1(T_{OC}s+1)}{s(T_2^2s^2 + T_1s + 1)}, \quad (33)$$

где

$$k_1 = \frac{k}{1 + kk_{OC}},$$

$$T_2^2 = \frac{TT_{OC}}{1 + kk_{OC}},$$

$$T_1 = \frac{T + T_{OC}}{1 + kk_{OC}}.$$

Здесь при сохранении интегрирующего свойства звена получается эффект введения производной, т.е. интегрирующее звено становится изодромным, а новые постоянные времени T_1 и T_2 , характеризующие инерционность звена, могут быть малыми за счет большого первичного коэффициента усиления k . В последнем случае имеем

$$k_1 \approx \frac{1}{k_{OC}}, \quad (34)$$

$$W(s) \approx \frac{T_{OC}s + 1}{k_{OC}s}, \quad (35)$$

т.е. обратную величину обратной связи.

Можно отметить, что вообще инерционное запаздывание в обратной связи (в отличие от такового в прямой цепи) целесообразно использовать для улучшения качества переходных процессов, получая эффект, аналогичный введению производной в прямой цепи. Однако при этом следует побеспокоиться об устойчивости контура, например, поместив корректирующее звено в прямую цепь. Общим свойством является также и то, что жесткие обратные связи аннулируют интегрирующее свойство звена (т.е. аннулируют астатизм системы, если в ней нет интегрирования в другом месте цепи звеньев), а гибкие обратные связи не устраняют астатизм.

Задание к лабораторной работе

1. Охватить единичной обратной связью интегрирующее, аperiodическое, интегрирующее с запаздыванием и колебательное звенья, посмотреть их переходные и частотные характеристики. Для звеньев прямой цепи выбираем коэффициент усиления $K = 10 \dots 100$, постоянные времени T – произвольно, коэффициент затухания ξ для колебательного звена $0,1 \dots 0,9$.

2. Охватить те же звенья каждым из видов параллельной коррекции, построить переходные и частотные характеристики, провести сравнительный

анализ полученных результатов. Для цепи обратной связи выбрать коэффициент усиления $K_{OC} = 0,1 \dots 1$, постоянную времени $T_{OC} = 0,01 \dots 1$.

3. Проанализировать влияние обратной связи на точность, качество и устойчивость исследуемых систем на примере одного из видов корректирующих устройств, указанных преподавателем.

4. Написать программу в виде меню со следующими возможностями: выбор передаточной функции в прямой цепи, выбор передаточной функции коррекции в цепи обратной связи, построение переходных и логарифмических частотных характеристик для исходной и скорректированной систем в одном окне, выход.

5. Построить графики изменения коэффициента передачи $dsgain(sys)$, перерегулирования $\max(\text{step}(sys))$ и частоты среза W_C в зависимости от изменения параметров звеньев в цепи обратной связи.

Для выполнения лабораторной работы воспользуйтесь в качестве базовой следующей программой:

```
% ar2d_ос4.Эта программа строит колебательное звено с инерционной
% гибкой обратной связью и циклы характеристик с параметром m.
% Строится дискретная модель, Ts - период дискретизации.
% Для исследования непрерывной модели закрыть строку формирования
% признака преобразования в дискретный объект.
clear, close all;
k=25; T=2; ch=[k]; zn=[T^2 2*T*.02 1]; % Параметры объекта.
so=tf(ch,zn); c='s'; % Непрерывный объект W(s) без ОС.
%c='z'; % Признак для преобразования в дискретный объект.
txt1=['Колебательное звено с инерционной гибкой обратной связью'];
txt2=['W(' ,c, ')=( ');
if c=='z'; % Если задан признак преобразования в дискретный
% объект, вычисляется период дискретизации Ts
Ts=T/3;
so=c2d(so,Ts); % Объект sk преобразуется в дискретный sk.
[ch,zn]=tfdata(sk,'v'); % Числитель и знаменатель дискретного
% объекта.
end % Конец преобразования в дискретный объект.
w={.1/T 10/T}; tk=10*T; % Диапазон частот и времени.
figure(1); step(so,tk); pause(0); % Объект W(s) без ОС.
figure(2); bode(so,w); % ЛАЧХ объекта W(s).
rez1=[]; disp('Разместите окна №1 и №2 '); pause
for i=-6:0; % Цикл по переменной i.
m=2^(i/2); % Изменяемый множитель m.
Koc=tf([m/8 0],[T/10 1]); % Обратная связь непрерывная.
if c=='z'; % Признак преобразования в дискретный объект.
Koc=c2d(Koc,Ts); % Преобразование ОС в дискретный объект.
end
disp(' Koc ='),Koc % Обратная связь.
[wc,ph]=wc_ph(so*Koc); % Частота среза и фаза разомкнутого контура.
rez1=[rez1,[m,wc,ph+180]']; % Регистрация m, wc, запаса по фазе.
```

```

sk=feedback(so,Кос); % Замкнутый контур.
[ch,zn]=tfdata(sk,'v'); % Замкнутый контур, числитель и знаменатель.
txt3=[txt2,poly2str(ch,c),' / (' ,poly2str(zn,c),' )'];
% Запись tf контура.
figure(1); hold on; % Открыть окно 1 и режим сохранения графиков.
step(sk,tk);% Добавить график ПФ объекта sk.
title([txt3]);% Нанести заголовок txt3.
hold off; % Выключить режим сохранения графиков в окне 1.
pause(0), % Сделать паузу, чтобы нарисовать график.
figure(2); hold on; % Открыть окно № 2 и режим сохранения графиков.
bode(sk,w); % Добавить графики частотных характеристик объекта sk.
title([txt1]); % Нанести заголовок txt1.
hold off; % Выключить режим сохранения графиков в окне 2.
pause; % Сделать паузу до нажатия клавиши.
end; % Окончание цикла по переменной i.
hold on; bode(tf(1/Кос),w);
% Добавить обратную характеристику ОС.
title('1/Кос отсекает часть ЛАЧХ объекта'); % Нанести заголовок.
hold off; % Выключить режим сохранения графиков в окне 2
zoom on; pause; % Включить режим масштабирования графиков в окне
% №2, в новом окне нарисовать два графика.
figure; plotyy(rez1(1,:),rez1(3,:),rez1(1,:),rez1(2,:)),grid
xlabel('Дополнительный коэффициент m');
title('Запас по фазе и частота среза разомкнутого контура - so*Кос.');
```

```

function [wc,ph]=wc_ph(sys,er); % Вычисляет частоту среза и фазу.
% wc_ph - частота среза на ниспадающем отрезке и фаза на частоте
% среза.
% [wc,ph]=wc_ph(sys,er), здесь sys - имя SISO lti-объекта
% er - относительная ошибка вычислений wc. wc - частота среза,
% ph - фаза на частоте среза объекта sys.
ni = nargin;
if ni==1, er=.001; end
w1=[];
[m,f,w]=bode(sys);
for i= length(w):-1:1;
if m(i)>=1 & m(i+1)<1; w1={w(i),w(i+1)}; break; end
end
if isempty(w1); disp('Частоты среза не обнаружено'); return; end
while (w1{2}-w1{1})>w1{2}*er;
[m,f,w]=bode(sys,w1);
for i=1: length(w);
if m(i)>=1 & m(i+1)<1; w1={w(i),w(i+1)}; wc=w(i);ph=f(i); break;
end
end % for
end % while

```

Контрольные вопросы

1. Каков общий механизм влияния отрицательных обратных связей?
2. Как приблизительно оценить изменение ЛАЧХ охваченного объекта от применения произвольной обратной связи?

3. Каково влияние жёсткой и гибкой обратных связей на интегрирующие звенья?
4. Каково влияние жёсткой и гибкой обратных связей на позиционные звенья?
5. Каково влияние жёсткой и гибкой обратных связей на дифференцирующие звенья?
6. Какой результат следует ожидать от интегрирующей обратной связи?
7. Изменения параметров самого объекта или обратной связи сильнее скажутся на характеристиках замкнутого объекта?
8. Как количественно оценить стабилизирующее воздействие обратной связи на изменение параметров охватываемого объекта управления?
9. Как связаны между собой глубина и стабилизирующее воздействие обратной связи?

Библиотека БГУИР

ПРИМЕНЕНИЕ ПОСЛЕДОВАТЕЛЬНОЙ КОРРЕКЦИИ ДЛЯ ДОСТИЖЕНИЯ НЕОБХОДИМЫХ ЗАПАСОВ УСТОЙЧИВОСТИ

Цель работы: изучить и освоить на практике методику расчёта последовательной коррекции в тех случаях, когда необходимо спроектировать систему с определенными запасами устойчивости по фазе; научиться оценивать влияние параметров корректирующей обратной связи на динамику замкнутой системы.

Краткие теоретические сведения

При автоматизированном расчете систем управления достаточно распространённой является задача подбора последовательной коррекции в условиях ограничения на мощность управления. На практике это ограничение представляется в виде ЛАЧХ объекта управления, причем в области высоких и средних частот используется пассивная коррекция (без усиления сигналов управления).

Отметим наиболее важные моменты, которые необходимо учитывать при выборе параметров корректирующей обратной связи для достижения необходимых запасов устойчивости:

1. Для сохранения быстродействия системы частоту среза скорректированного разомкнутого контура нежелательно далеко удалять от частоты среза объекта.

2. Для придания скорректированному контуру необходимых запасов устойчивости по фазе следует использовать корректирующие звенья с положительным наклоном ЛАЧХ в предполагаемой области частоты среза скорректированного контура. При этом коррекция фазовой характеристики звеном 1-го порядка не достигает уровня $+90^\circ$ и зависит от протяжённости участка подъёма ЛАЧХ.

3. Наклон ЛАЧХ объекта управления на частоте среза можно оценить по фазовой характеристике. Необходимое количество вышеописанных корректирующих звеньев 1-го порядка будет на единицу меньше.

4. Максимум коррекции фазовой характеристики совпадает с серединой участка подъёма ЛАЧХ. Поэтому для достижения максимума запаса устойчивости необходимо совместить середину участка подъёма ЛАЧХ с частотой среза ω_{co} скорректированного разомкнутого контура.

Совмещение характеристик производится по следующей методике. По фазе на частоте среза разомкнутого объекта (ω_o) ω_{cn} оцениваем наклон ЛАЧХ. Пусть он равен $-k$, а наклон ЛАЧХ контура скорректированного равен -1 . Тогда число корректирующих звеньев равно $(k - 1)$, а передаточная функция разомкнутого контура управления умножается на передаточную функцию

последовательной коррекции с выбранным количеством корректирующих звеньев $(k-1)$ и принимает вид $W_0 \cdot s k^{k-1}$.

Протяжённость участка коррекции от $\omega_{сн}$ до p_1 в k раз меньше, чем от W_c до p_1 . В таком случае, если $p_1 = \omega_{сo} n_1^{1/2}$, то $p_1 = -\omega_{сн} n_1^{1/(2*k)}$; $z_1 = p_1/n_1$.

5. Нули и полюса объекта управления, совпадающие с участком подъёма ЛАЧХ коррекции, желательно компенсировать полюсами и нулями коррекции.

Таким образом, выбрав необходимое количество включённых последовательно корректирующих звеньев 1-го порядка и варьируя протяжённость участка подъёма ЛАЧХ коррекции, постоянно совмещая его середину с частотой среза разомкнутого контура, можем спроектировать достаточно экономную последовательную коррекцию, обеспечивающую желаемый запас по фазе.

Алгоритм проектирования последовательной коррекции на Matlab может состоять из следующих этапов:

1. Ввод в окно редактора исходных данных о параметрах модели объекта управления и ввод желаемого запаса устойчивости по фазе.

2. Определение необходимого числа корректирующих звеньев 1-го порядка:

2.1. Вычисление частоты среза $\omega_{сo}$ объекта и фазы ρ_0 на частоте среза.

2.2. Оценка наклона ЛАЧХ по фазовой характеристике на частоте среза $\omega_{сo}$ и определение необходимого числа включённых последовательно корректирующих звеньев 1-го порядка.

3. Организация цикла изменения протяжённости участка подъёма ЛАЧХ коррекции, внутри которого выполняются пункты 4 – 10.

4. Вычисление полюса и нуля коррекции, которые приблизительно совмещают середину участка подъёма ЛАЧХ с частотой среза разомкнутого контура.

5. Формирование коррекции из необходимого числа включённых последовательно корректирующих звеньев 1-го порядка.

6. Компенсация полюса объекта нулями коррекции.

7. Компенсация нуля объекта полюсами коррекции.

8. Проверка и совмещение $\omega_{сo}$ с максимумом фазовой характеристики коррекции.

9. Вычисление запаса устойчивости по фазе для скорректированной системы.

10. Вывод в одно окно ЛАЧХ разомкнутого контура и переходные характеристики устойчивого замкнутого контура.

11. Выход из цикла при достаточном запасе устойчивости по фазе.

12. Представление результатов проектирования в виде графиков и передаточных функций.

Задание к лабораторной работе

1. Изучить предложенный алгоритм проектирования последовательной коррекции по его описанию и программной реализации.

2. Просмотреть работу программы для нескольких вариантов объектов управления разного порядка астатизма.

3. Проследить процесс компенсации нулей и полюсов объекта в коррекции.

4. Выяснить зависимость глубины коррекции от порядка объекта и желаемого запаса по фазе.

Для выполнения лабораторной работы воспользуйтесь в качестве базовой следующей программой:

```
% Программа cor_posk проектирует последовательную коррекцию с
% компенсацией изломов ЛАЧХ объекта и обеспечивает желаемый запас
% по фазе. Используются обозначения: sys - передаточная функция
% объекта управления, pm - желаемый запас устойчивости по фазе,
% sk - передаточная функция звеньев последовательной коррекции. Sk
% состоит из нескольких включаемых последовательно элементов sk1.
% sk1 - звено коррекции 1-го порядка с наклонами ЛАЧХ (0, +1, 0),
% т.е. с изломом вверх (нулём z1) и изломом вниз (полюсом p1).
% sr - передаточная функция разомкнутого контура управления.
% Эффект коррекции зависит от  $n1=p1/z1$  - длины участка с наклоном +1.
clear; close all; % Очистить память. Закрыть все окна.

% 1. Вводим и корректируем исходные данные.
sys=zpk([], [0 0 -1 -25 -6], 4200) % Вводим модель объекта управления.
pm=45 % Требуемый запас устойчивости по фазе.

% 2. Определяем необходимое число корректирующих звеньев 1-го порядка.
[wco, pho]=wc_ph(sys); % Частота среза и фаза на частоте среза.
k=ceil(-pho/90) % Оценка сверху наклона ЛАЧХ на частоте среза wco.
if k<=1; k=k+4; end % Корректировка сдвига фазовой характеристики.
[z0, po, ko]=zpkdata(sys, 'v'); % Выделяем нули и полюса объекта.
figure(1); bode(sys); pause(0) % ЛАЧХ разомкнутого объекта Wo.
hold on % Открыть режим сохранения предыдущих графиков в окне № 1

% 3. Цикл подбора последовательной коррекции (k-1)-го порядка.
rez=[]; % Для регистрации результатов.
for i=4:24; % Цикл изменения протяжённости участка подъема ЛАЧХ коррекции.
n1=2^(i/4); % Задание протяжённости участка подъема ЛАЧХ коррекции.

% 4. Вычисление полюса и нуля коррекции, приблизительно
% совмещающих середину участка подъема ЛАЧХ с частотой среза
% разомкнутого контура.
p1=-wco*n1^(1/(2*(k))); z1=p1/n1; % Вычисление полюса и нуля коррекции.
wcp=(p1*z1)^(1/2); % wcp - максимума коррекции фазовой
% характеристики.
```

```

sk1=zpk(z1, p1, 1); % sk1 - элемент коррекции.

% 5. Формирование коррекции из необходимого числа элементов.
sk=sk1; k1=k-1; while k1>1; sk=sk*sk1; k1=k1-1; end % sk - коррекция.

% 6. Учёт изломов ЛАЧХ объекта в расчёте коррекции.
[zk,pk,kk]=zpkdata(sk,'v');% Выделяем нули и полюса коррекции.
% В частотном диапазоне коррекции (-z1 -p1) добавляем к нулям
% коррекции полюса объекта, выбрасываем лишние нули z1.
for j=1:length(po), if abs(po(j))<-p1 & abs(po(j))>-z1;
zk=[zk;po(j)]; zk(1)=[]; end; end % Добавляем к полюсам коррекции
% нули объекта, добавляем нули z1.
for j=1:length(zo), if abs(zo(j))<-p1 & abs(zo(j))>-z1; pk=[pk;
zo(j)]; zk=[z1;zk]; end; end
sk=zpk(zk,pk,kk); % Коррекция с компенсацией нулей и полюсов объекта.

% 7. Совмещаем wc с максимумом коррекции фазовой характеристики.
sr=minreal(sys*sk); % Контур управления разомкнутый
[wc,ph]=wc_ph(sr); % Частота среза и фаза на частоте среза контура.
kwc=wc/wcsr; % Проверка совмещения wc с максимумом фазовой коррекции.
if kwc<.9 | kwc>1.3, sk=sk/kwc; sr=minreal(sys*sk);
[wc,ph]=wc_ph(sr); kwc=wc/wcsr; end % Совмещения wc с максимумом
% фазовой коррекции.

% 8. Вычисляем запас устойчивости по фазе.
ph_z=ph+180;% Запас устойчивости по фазе.
rez=[rez,[n1;ph_z;wc;kwc]]; % Регистрация результатов.
txt0=('Протяжённость подъёма ЛАЧХ в ');
txt1=[txt0,num2str(n1), 'Запас по фазе = ',num2str(ph_z)];
% Заголовок.

% 9. Регистрируем ЛАЧХ разомкнутого контура.
bode(sr);% Добавляем ЛАЧХ разомкнутого контура управления.
title(txt1); pause(0), % Заголовок txt1.

% 10. Переходные характеристики устойчивого замкнутого контура.
if ph_z>15; hold off; % Если запас есть, закрыть сохранение окна № 1.
sz=feedback(sr,1); sz=c1_zp(minreal(sz));
% Замкнуть устойчивый контур.
figure(2); hold on; % Открыть окно № 2 и режим сохранения
графиков.
step(sz,9/wc); % Добавить переходную характеристику замкнутого контура
title('контур управления замкнутый'); pause(0), hold off;
% Закрыть окно № 2.
figure(1); hold on; % Открыть окно № 1 и режим сохранения графиков.
end % Конец от if ph_z > 15;

% 11. Выходим из цикла при достаточном запасе устойчивости по фазе.
if ph_z >=pm; break; end % Проверка запаса по фазе и выход из цикла.
end % Завершение для for i=4:24;
hold off; zoom on; % Включить масштабирование графиков в окне № 1.

```

```

% 12. Представляем результаты проектирования.
figure; bode(sys,sr,sk); title(['sys,sr,sk. ',txt1]); ); pause(0)
sz=feedback(sr,1); sz=(minreal(ss(sz)));
figure; step(sz); pause(0), title(' контур управления замкнутый');
figure; plotyy(rez(1,:),rez(2,:),rez(1,:),rez(3,:)), grid
xlabel('Протяжённость подъёма ЛАЧХ коррекции');
title('Запас по фазе и частота среза контура управления разомкнутого');
disp(' Объект управления'); zpk(sys),
disp(' Последовательное корректирующее устройство'); zpk(sk),
disp(' Контур управления разомкнутый'); zpk(sr),
disp(' Контур управления замкнутый'); zpk(sz)

```

Контрольные вопросы

1. Что такое пассивная коррекция?
2. Укажите места в структуре системы управления, удобные для последовательной коррекции.
3. Как связаны частота среза и быстродействие системы управления?
4. Как оценить порядок звеньев коррекции?
5. Как оценить глубину коррекции?
6. Изменяет ли порядок астатизма системы предложенный вариант последовательной коррекции?
7. Какова должна быть последовательная коррекция, чтобы порядок астатизма системы изменился?
8. Как соотносятся величина запаса устойчивости по фазе и точность системы с предложенным вариантом последовательной коррекции?

СИНТЕЗ ПАРАЛЛЕЛЬНОЙ КОРРЕКЦИИ ПО ТРЕБОВАНИЯМ КАЧЕСТВА ПЕРЕХОДНОГО ПРОЦЕССА

Цель работы: изучить методику расчёта звеньев параллельной коррекции, удовлетворяющей заданным требованиям к переходной характеристике системы.

Краткие теоретические сведения

В системах управления кроме главной (единичной) обратной связи часто используют дополнительные (корректирующие) обратные связи, охватывающие обычно часть прямой цепи. Изменяя динамические свойства охваченных звеньев, корректирующие обратные связи дополнительно стабилизируют их характеристики. Изменение параметров охваченных звеньев компенсируется корректирующей обратной связью, пока не происходит потеря устойчивости корректирующего контура. Обратная связь вычисляется по формуле

$$W_{OC} = 1/W_{жс} - 1/W_O, \quad (36)$$

т.е. определяется как разница обратных передаточных функций желаемой ЛАЧХ и охваченного объекта. Коррекция с помощью обратной связи происходит в полосе пропускания корректирующего контура. За пределами полосы пропускания ЛАЧХ желаемая и охваченного объекта должны совпадать. Обратная связь лишь отрезает часть ЛАЧХ охваченного объекта, поэтому желаемая ЛАЧХ не может проходить выше ЛАЧХ охваченного объекта.

ЛАЧХ типичной корректирующей обратной связи состоит из участков, обеспечивающих запасы устойчивости основного контура, запасы устойчивости корректирующего контура, физическую реализуемость коррекции. Перечисление участков соответствует увеличению частоты.

При изменении числа охваченных звеньев результат от параллельной коррекции сохраняется, если останутся неизменными характеристики разомкнутого корректирующего контура. Поэтому целесообразно вначале рассчитывать обратную связь, охватывающую всю прямую цепь. Затем пересчитывать её в местную обратную связь, добавляя в её состав неохваченные звенья, и тем самым сохранить свойства корректирующего контура.

Требования к переходной характеристике задаются допустимыми значениями её длительности и величины перерегулирования.

Обычно решение задачи начинают с построения желаемой модели разомкнутой системы, приблизительно удовлетворяющей требованиям к замкнутой системе. При компьютерном проектировании имеет смысл начинать с построения желаемой модели замкнутой системы, точно удовлетворяющей требованиям к переходной характеристике, а затем переходить к модели

разомкнутой системы. Для построения модели замкнутой системы удобно использовать нормированные полиномы: Латропа и Грехема, Баттерворда, с кратными корнями и т.д.

Проектирование корректирующей обратной связи целесообразно проводить последовательно, по этапам. Сначала необходимо рассчитать обратную связь, охватывающую целиком прямую цепь, не рассматривая вопрос о её физической реализации, затем пересчитать её для конкретного контура, охватывающего часть объекта управления. На следующем этапе следует обеспечить физическую реализуемость обратной связи (увеличить порядок знаменателя до порядка числителя) при сохранении устойчивости корректирующего контура. При последующей проверке запасов устойчивости корректирующего и основного контуров, а также переходной характеристики замкнутой системы убедиться в правильности результатов. В противном случае необходимо внести изменения в требования к системе или в управляющие параметры проектирования.

Таким образом, алгоритм проектирования параллельной коррекции может состоять из следующих этапов:

1. Ввод исходных данных: модели объекта и её части, не охватываемой корректирующей обратной связью, желаемой длительности переходной характеристики и допустимой величины перерегулирования. Задание дополнительных управляющих параметров проектирования, в нашем случае – доли запаса устойчивости корректирующего контура для придания физической реализуемости коррекции.

2. Построение желаемой модели:

- 2.1. Определение порядка модели по наклону ЛАЧХ объекта на высоких частотах.

- 2.2. Выбор типа нормированной модели замкнутой системы исходя из допустимого перерегулирования.

- 2.3. Масштабирование модели замкнутой системы под желаемую длительность переходной характеристики.

- 2.4. Размыкание масштабированной модели.

- 2.5. Расчёт дополнительного усилителя к объекту для совмещения ЛАЧХ объекта и модели на высоких частотах.

3. Расчёт корректирующей обратной связи:

- 3.1. Расчёт обратной связи, охватывающей всю прямую цепь.

- 3.2. Поправка ОС, если ЛАЧХ объекта ниже желаемой на низких частотах.

- 3.3. Пересчёт её путём добавления в её состав неохваченных звеньев.

- 3.4. Обеспечение физической реализуемости обратной связи путем добавления кратных полюсов, расходуя на них заданную долю запаса устойчивости корректирующего контура.

4. Проверка свойств корректирующего контура и замыкание его.

5. Формирование основного контура, проверка его свойств и замыкание.

6. Регистрация и вывод результатов проектирования.

Задание к лабораторной работе

1. Изучить предложенный алгоритм проектирования параллельной коррекции по его описанию и программной реализации.

2. Просмотреть работу программы для нескольких вариантов объектов управления разного порядка астатизма.

3. Проследить процесс создания модели желаемого контура управления.

4. Сравнить результаты придания физической реализуемости обратной связи при изменении охватываемой части объекта.

5. Выяснить зависимость глубины коррекции от порядка объекта и желаемой длительности переходной характеристики.

6. Для выполнения лабораторной работы воспользуйтесь в качестве базовой следующей программой:

```
% cor_os4 Коррекция в обратной связи, охватывающая часть объекта.
% Коррекция под требования к переходной характеристике.
% Задаются объект управления sol и so_n – не охватываемый обратной связью.
% Объект может быть неустойчивый, но без положительных нулей.
% Желаемая замкнутая модель использует нормированные полиномы,
% масштабируется и размыкается.
% Обратная связь вычисляется как разница обратных передаточных
% функций желаемой и объекта, затем дополняется полюсами для
% обеспечения физической реализации.
clear; close all;

% Исходные данные и требования к системе управления.
sol=zpk(tf([ 10 5 ],[1 3 14 20 3 ])) % Объект управления.
so_n=tf(1,[1 0]); % Часть объекта управления, не охватываемая обратной связью.
tnn=0.4; per=10; % Требования к переходной характеристике.

% Дополнительные управляющие параметры при проектировании.
zu=0.35 % Доля запаса устойчивости контура под физическую реализацию.

% Порядок модели замкнутой системы.
[z,p,k]=zpkdata(sol,'v'); % Извлекаем форму zpk объекта sol.
i=length(p)-length(z); % i – наклон ЛАЧХ объекта so на высоких
% частотах.

% Под требования к перерегулированию выбираем тип полиномов.
run poll ; % Полиномы Латропа и Грехема.
run pola; % Полиномы с монотонными переходными характеристиками.
if per>0; zn=poll(i).polin; tn=poll(i).tnn; % Полиномы Латропа и
% Грехема.
else; zn=pola(i).polin; tn=pola(i).tnn; end % Полиномы с
% монотонными характеристиками.
sz=tf(1,zn); % sz – модель замкнутой системы нормированная.

% Масштабирование во времени модели замкнутой системы.
mt=tn/tnn; % mt – коэффициент масштабирования во времени.
```

```

% tn - нормированная, tnn - желаемая длительность переходной
% характеристики
szm=mascht2(sz,mt); % Масштабирование во времени модели.

% Размыкание модели.
[sg,ko]=razm1(szm); % Sg - модель желаемой разомкнутой системы.

% Уравнение на высоких частотах ЛАЧХ объекта управления и
% желаемой ЛАЧХ.
[zg,pg,kg]=zpkdata(sg,'v'); % Извлекаем форму zpk объекта sg.
ky=kg/k; % Уравнение путем добавления к объекту звена с
% коэффициентом ky.
so=zpk(z,p,kg); % Фактически so= sol*ky.
figure(3); bode(sg,so); legend('Желаемая','объекта * ky',0);
title('sg, so*ky, ЛАЧХ объекта не ниже желаемой на высоких частотах?');
pause(0);

% Расчёт обратной связи soc1, охватывающей sol*ky.
soc1=1/sg-1/so; % Обратная связь (может быть физически не реализуемая).
soc1=kl_zp(soc1) % Очистка от ошибок округления.

% Поправка, если ЛАЧХ объекта ниже желаемой на низких частотах.
[z1,p1,k1]=zpkdata(soc1,'v'); % Нули и полюса обратной связи soc1.
for i=length(z1):-1:1; if (z1(i))>0; z1(i)=0; break; end; end
soc1=zpk(z1,p1,k1); % Устраняем положительный нуль обратной связи.
svk=minreal((soc1*(so))); % Корректирующий контур разомкнутый.
[wc,ph]=wc_ph(svk); pm=ph+180; % Частота среза и запас по фазе контура.
figure; margin(svk), legend('модель разомкнутого корректирующего контура')
pause(0) % Запасы устойчивости разомкнутого корректирующего контура.
figure; bode(sg,so,svk,1/soc1);
legend('Желаемая','Объект','Корректирующий контур',' - Лос',3);
title(['sg, so*ky, svk, 1/soc1; запас по фазе контура svk =
',num2str(pm)]);pause(0)

% Пересчёт обратной связи soc, не охватывающей so_n часть объекта so.
soc=minreal(soc1*so_n);

% Обеспечение физической реализуемости обратной связи soc.
[z,p,k]=zpkdata(soc,'v'); % Нули и полюса обратной связи soc.
n2=(length(z)-length(p)) % n2 - число дополнительных полюсов.
if n2>0;

% soc дополняем n2 полюсами величиной p2. Под дополнительные
% полюса выделяем часть запаса устойчивости (zu*pm)
% корректирующего контура.
pm_p2=zu*pm/n2; % pm_p2 - запас устойчивости под каждый полюс p2.
% Полюса p2 располагаем правее wc в 57.3/pm_p2 раз.
p2=-wc*(57.3/pm_p2)*ones(n2,1); % Дополнительные n2 полюса p2.
sk=minreal(zpk(z,[p;p2],k*(-p2(1))^n2)); % Обратная связь реализуемая.
else; sk=soc; % Обратная связь физически реализуема без дополнения.
end % if n2>0

```

```

% Корректирующий контур с реализуемой обратной связью.
figure; margin(minreal(sk*so/so_n)); % Запасы устойчивости контура.
legend('разомкнутый физически реализуемый корректирующий контур',4)
pause(0)
sck=minreal(feedback(minreal(so/so_n) ,sk)); % Замкнутый контур коррекции.

% Основной контур.
figure; bode(sg, sck*so_n, sk); title('sg, sck*so_n, sk');
legend('Желаемая','основной контур разомкнутый','Лос',3); pause(0)
sckz=minreal(feedback(minreal(sck*so_n),1)); % Основной контур замкнутый.
figure; step(sckz,szm); title('sckz, szm');
legend('Основной контур замкнутый ','модель контура замкнутого',4);
pause(0);
figure; pzmap(sckz); title('основной контур замкнутый '); pause(0);

% Вывод результатов проектирования.
disp('Требования к переходной характеристике замкнутой системы'); tnn,пер
disp('Дополнительные управляющие параметры при проектировании');
disp('Доля запаса устойчивости под физическую реализацию коррекции'); zu
disp(' Объект управления'); sol,
disp('Часть объекта управления, не охваченная обратной связью'); so_n,
disp('Дополнительный усилитель в прямой цепи'); ky,
disp('Корректирующая обратная связь физически реализуемая'); sk,
disp('Корректирующий контур замкнутый'); sck,
disp('Основной контур разомкнутый '); minreal(sck*so_n),
disp('Основной контур замкнутый '); sckz

function Smt=mascht2(S0,mt)
% mascht2 - Масштабирование моделей.
% Smt=mascht2(S0,mt); Объект Smt быстрее объекта S0 в mt раз.
ni = nargin; no = nargout;
error(nargchk(2,2,ni));
if ~isproper(S0), error('Модель неправильная.'); end
[a1,b1,c1,d1]=ssdata(ss(S0)); % Извлекаются параметры объекта S0 в ss-форме
mt=abs(mt);
a2=a1*mt; b2=b1*mt ;% Матрицы А и В умножаются на mt.
Smt=ss(a2,b2,c1,d1); % Вводятся в объект Smt.
txt0=('Исходный объект S0');
txt=(['Масштабированный во времени объект Smt;
mt= ',num2str(mt)]);
disp(txt0), zpk(S0) % Сравнение исходного S0 и масштабированного объектов.
disp(txt), zpk(Smt)
figure; bode(S0,Smt), title([txt0,txt]); pause(0)

function [sr,koc]=razml(sz)
% razml - размыкание объекта с астатизмом 1-го порядка.
%[sr,koc]=razml(sz), sz - замкнутый объект,
% sr - разомкнутый объект с астатизмом 1-го порядка.
% koc - коэффициент обратной связи.
ni = nargin; no = nargout;

```

```

error(nargchk(1,1,ni));
if ~Isis(sz), error('LTI system SYS must be SISO.');
```

end
[chazzans]=tfdata(tf(sz),'v');
% Размыкание объекта
koc=zns(length(znz))/chz(length(chz)); % koc коэффициент обратной
% связи.
chr=chz; % Числитель разомкнутого объекта.
znr=(znz-chr*koc); % Знаменатель разомкнутого объекта.
sr=minreal(zpk(tf(chr,znr))); % Разомкнутый объект с астатизмом
% 1-го порядка.
disp(' Замкнутый объект,');zpk(sz),
disp([' Коэффициент обратной связи Koc = ',num2str(koc)]);
disp('Разомкнутый объект с астатизмом 1-го порядка');zpk(sr)
figure; bode(sz,sr),
title(['Замкнутый объект, разомкнутый объект с Koc =
',num2str(koc)]);
pause(0);

function [scl]=cl_zp(sys,w)
% cl_zp - чистит lti-объект, устраняя изломы ЛАЧХ за пределами w.
% [scl]=cl_zp(sys,w); где: sys - исходный lti-объект;
% w - задаёт границы частот [wmin wmax], по умолчанию w=[1e-3,1e5].
ni = nargin; error(nargchk(1,2,ni));
if ni==1, w=[1e-3,1e5]; **end**
[z,p,k]=zpkdata(sys,'v');
for i=length(z):-1:1;
if abs(z(i))<w(1); z(i)=0; % Обнуляем малые нули.
elseif abs(z(i))>w(2); k=k*abs(z(i)); z(i)=[]; **end** % Устраняем
% большие.
end
for i=length(p):-1:1;
if abs(p(i))<w(1); p(i)=0; % Обнуляем малые полюса.
elseif abs(p(i))>w(2); k=k/abs(p(i)); p(i)=[]; **end** % Устраняем
% большие.
end
scl=minreal(zpk(z,p,k)); % Объект без изломов ЛАЧХ за пределами w.

% pola Полиномы с быстрыми монотонными переходными характеристиками
pola(8)=struct('polin',[1 5.5 15.43 27.7 34.7 30.25 17.8 6.32 1],'tnn',10.07,'nep',0);
pola(7)=struct('polin',[1 5.8 15.25 23.8 24.1 15.75 5.98 1],'tnn',9.67,'nep',0);
pola(6)=struct('polin',[1 4.07 8.75 11.4 9.4 4.61 1],'tnn',7.4,'nep',0);
pola(5)=struct('polin',[1 4.04 8.03 8.07 4.43 1],'tnn',5.9,'nep',0);
pola(4)=struct('polin',[1 3.37 4.87 3.48 1],'tnn',5.4,'nep',0);
pola(3)=struct('polin',[1 1.98 2.38 1],'tnn',4.3,'nep',0);
pola(2)=struct('polin',[1 1.78 1],'tnn',4,'nep',0);

% poll Полиномы Латропа и Грехема, самые быстрые переходные
% характеристики с малым перерегулированием
poll(2)=struct('polin',[1 1.4 1],'tnn',2.8,'nep',4.5);

```
poll(3)=struct('polin',[1 1.75 2.15 1],'tnn',3.6,'nep',3);  
poll(4)=struct('polin',[1 2.1 3.4 2.7 1],'tnn',4.7,'nep',10.2);  
poll(5)=struct('polin',[1 2.8 5 5.5 3.4 1],'tnn',5.2,'nep',2.5);  
poll(6)=struct('polin',[1 3.25 6.6 8.6 7.45 3.95 1],'tnn',5.4,'nep',3);  
poll(7)=struct('polin',[1 4.475 10.42 15.08 15.54 10.64 4.58 1],'tnn',7.7,'nep',6.4);  
poll(8)=struct('polin',[1 5.2 12.8 21.6 25.75 22.2 13.3 5.15 1],'tnn',10,'nep',12.8).
```

Контрольные вопросы

1. Перечислите преимущества параллельной коррекции по сравнению с последовательной.
2. Укажите в структуре системы управления места, удобные для подсоединения параллельной коррекции.
3. Как должны соотноситься ЛАЧХ желаемая и ЛАЧХ объекта управления, и какими мерами это достигается в программе?
4. Зачем и каким образом можно минимизировать полосу пропускания корректирующего контура?
5. Каково функциональное назначение отдельных частей ЛАЧХ корректирующей обратной связи?
6. Как выглядит типичная ЛАЧХ обратной связи, охватывающая всю прямую цепь системы управления?
7. Как изменяется обратная связь при переводе в неохватываемую часть объекта интегратора?
8. Как соотносятся частоты среза основного и корректирующего контуров системы управления?
9. Как соотносятся запасы устойчивости по фазе основного и корректирующего контуров системы управления?
10. Как оценить глубину параллельной коррекции?

ЛИТЕРАТУРА

1. Теория автоматического управления. Ч. 1, 2. / Под ред. А.А. Воронова. – М.: Высш. шк., 1986.
2. Попов Е.П. Теория линейных систем автоматического управления и регулирования. – М.: Наука, 1989.
3. Первозванский А.А. Курс теории автоматического управления. – М.: Наука, 1986.
4. Андриевский Б.Р., Фрадков А.Л. Избранные главы теории автоматического управления с примерами на языке MATLAB. – СПб.: Наука, 1999.
5. Справочник по электрическим машинам / Под общ. ред. И.П. Копылова и Б.К. Клокова. Т. 1, 2. – М.: Энергоатомиздат, 1988.
6. Медведев В., Потемкин В. Control System Toolbox. Matlab 5 для студентов. – М.: Диалог-МИФИ, 2000.
7. Лазарев Ю. Matlab 5.x. – Киев: ВНУ, 2000.
8. Потемкин В. Система Matlab 5 для студентов. – М.: Диалог-МИФИ, 1998.

ПРИЛОЖЕНИЕ

Паспортные данные двигателей постоянного тока [5]

№ п.п.	Марка двигателя	P_H , Вт	ω_H , рад/с	U_H , В	I_H , А	M_H , Н·м	$J_{ДВ} \cdot 10^{-4}$, кг·м ²	$R_{я}$, Ом	$L_{я}$, мГн	$T_{я}$, мс
1	СЛ-121	7,5	470	110	0,16	0,014	0,038	240	130	
2	СЛ-161	8,6	420	110	0	0,02	0,052	170	125	
3	СЛ-221	13	377	110	0,25	0,034	0,137	117	230	
4	СЛ-261	24	377	110	0,41	0,064	0,196	51	140	
5	СЛ-281	26	545	24	2,16	0,049	0,196	1,15	0,5	
6	СЛ-321	38	315	110	0,58	0,123	0,59	25,8	130	
7	СЛ-361	50	315	110	0,75	0,156	0,687	20,5	115	
8	СЛ-369	55	377	110	0,8	0,147	0,687	15,2	90	
9	СЛ-521	77	315	110	1,07	0,245	1,67	8,5	58	
10	МИГ-90А	90	315	27	4,5	0,236	0,2	1,1	–	5,1
11	МИГ-180А	180	315	27	8,9	0,573	0,39	0,56	–	4
12	ПГ-0,12	120	315	27	7	0,38	0,3	0,3	–	3,9

Учебное издание

Антипова Марина Александровна,
Хаджинов Михаил Касьянович

АВТОМАТИЗИРОВАННЫЙ РАСЧЁТ СИСТЕМ УПРАВЛЕНИЯ

Лабораторный практикум
для студентов специальностей

53 01 03 «Автоматическое управление в технических системах» и
53 01 07 «Информационные технологии и управление в технических системах»
всех форм обучения

Редактор Н.А. Бебель

Подписано в печать 22.07.2003.
Печать ризографическая.
Уч.-изд. л. 1,8.

Формат 60x84 1/16.
Гарнитура «Таймс».
Тираж 200 экз.

Бумага офсетная.
Усл. печ. л. 2,44.
Заказ 159.

Издатель и полиграфическое исполнение:
Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники».
Лицензия ЛП № 156 от 30.12.2002.
Лицензия ЛВ № 509 от 03.08.2001.
220013, Минск, П.Бровки, 6.