

Федеральное агентство по образованию  
Государственное образовательное учреждение  
высшего профессионального образования  
“Ивановская государственная текстильная академия”  
(ИГТА)

Кафедра автоматики и радиоэлектроники

**Исследование линейных систем автоматического  
управления с помощью приложения Control System Toolbox  
вычислительной системы MATLAB (версия 6.5)**

*Методические указания к выполнению лабораторных работ  
по теории автоматического управления*

Иваново 2005

Методические указания предназначены для студентов, обучающихся по направлению 550200 Автоматизация и управление, и имеют своей целью помочь им в овладении современными методами исследования систем автоматического управления. Методические указания содержат описание 3 лабораторных работ, в которых рассматриваются вопросы анализа и проектирования линейных непрерывных систем и линейных дискретных систем. Выполнение лабораторных работ построено на использовании вычислительной системы MATLAB и ее приложений Simulink, Control System Toolbox, Symbolic Math и других. Указания могут быть полезны студентам других специальностей, изучающим современные методы анализа и синтеза систем автоматического управления.

Составитель      д-р техн. наук, доц. Н.А. Кулида

Научный редактор канд. техн. наук В.И. Забелин

Научный редактор В.И. Забелин

Редактор      В.В. Зимнякова

Корректор      Е.В. Минаева

---

Лицензия ИД №06309 от 19.11.2001. Подписано в печать 18.11.2005.

Формат 1/16 60×84. Бумага писчая. Плоская печать.

Усл. печ. л. 1,86. Уч.-изд. л. 1,78. Тираж 40 экз. Заказ №\_\_\_\_\_

---

Редакционно-издательский отдел  
Ивановской государственной текстильной академии  
Отдел оперативной полиграфии  
153000 г. Иваново, пр. Ф. Энгельса, 21

## Лабораторная работа № 1

### Синтез систем автоматического управления в приложении Control System Toolbox методом корневых годографов

#### Краткие сведения из теории

Известно, что если модель линейной системы автоматического управления (САУ) задается в виде структурной схемы, состоящей из типовых звеньев, то связь между входом и выходом системы определяется передаточной функцией  $W(s)$ , которую можно представить в виде:

$$W(s) = \frac{B(s)}{A(s)}, \quad (1)$$

где  $s$  – комплексная переменная,  $B(s)$  – полином степени  $m$ ;  $A(s)$  – полином степени  $n$ .

Для физически реализуемых САУ  $m \leq n$ . Коэффициенты указанных полиномов действительные числа. Полюсами передаточной функции называются корни полинома – знаменателя  $A(s)$ , а нулями – корни полинома числителя  $B(s)$ . Применение метода корневого годографа базируется на зависимости свойств линейной САУ от полюсов и нулей ее передаточной функции. Полином  $A(s)$  называется также характеристическим многочленом передаточной функции  $W(s)$ . Положение полюсов  $W(s)$  на комплексной плоскости определяет устойчивость САУ, а в совокупности с нулями – вид импульсной переходной  $w(t)$  и переходной  $h(t)$  функций.

Метод корневого годографа позволяет находить полюса и нули передаточной функции замкнутой системы, располагая полюсами и нулями разомкнутой системы при изменении коэффициента усиления разомкнутой системы  $K$ . Передаточную функцию разомкнутой системы  $W_p(s)$  представим в виде:

$$W_p(s) = \frac{ka \prod_{j=1}^m (s - s_j^0)}{\prod_{i=1}^n (s - s_i^*)}, \quad (2)$$

где  $s_j^0$  – нули передаточной функции  $W_p(s)$ , ( $j = \overline{1, m}$ );  $s_i^*$  – полюса передаточной функции  $W_p(s)$ , ( $i = \overline{1, n}$ ),  $n$  и  $m$  – порядки знаменателя и числителя;  $k$  – коэффициент усиления разомкнутой системы;  $a$  – коэффициент, получающийся при разложении числителя и знаменателя передаточной функции  $W_p(s)$  разомкнутой системы.

Передаточная функция разомкнутой системы, как правило, задается в виде отношения произведений передаточных функций стандартных (типовых)

звеньев, при описании которых используются выражения трех видов:

$$Ts ; \quad (3)$$

$$Ts + 1; \quad (4)$$

$$T^2 s^2 + 2T\xi s + 1. \quad (5)$$

Здесь  $T$  – постоянная времени,  $\xi$  – коэффициент затухания.

Если выражения (3–5) стоят в знаменателе передаточных функций звеньев, то они называются соответственно интегрирующим, апериодическим, колебательным ( $0 < \xi < 1$ ). Если выражения (3–5) стоят в числителе передаточной функции (1), то звенья называются соответственно дифференцирующим, форсирующим первого порядка и форсирующим второго порядка. Для перехода от стандартной формы записи к формуле (2) необходимо вычислить полюса и нули соответствующих типовых звеньев.

Нули и полюса для передаточных функций, использующих выражения (3–5), находятся из формул:

$$s^{*(0)} = 0, \quad (6)$$

$$s^{*(0)} = -\frac{1}{T}, \quad (7)$$

$$s_{1,2}^{*(0)} = -\frac{\xi}{T} \pm \sqrt{\left(\frac{\xi}{T}\right)^2 - \frac{1}{T^2}} \quad (8)$$

или

$$s_{1,2}^{*(0)} = -\frac{1}{T}(\xi \pm i \cos \varphi), \quad (9)$$

где  $\varphi = \arcsin \xi$ .

Коэффициент  $a$  вычисляется по формуле

$$a = \frac{\prod_{j=1}^m T_j^0}{\prod_{i=1}^m T_i^0}. \quad (10)$$

Для звеньев, использующих выражение (5), соответствующая постоянная времени входит в выражение (10) в квадрате.

При замыкании системы с передаточной функцией  $W_p(s)$  единичной обратной связью передаточная функция замкнутой системы  $W_z(s)$  принимает вид:

$$W_3(s) = \frac{W_p(s)}{1 \pm W_p(s)}, \quad (11)$$

где знак "+" соответствует отрицательной обратной связи; знак "-" – положительной обратной связи.

Из (11) следует, что нули передаточной функции замкнутой системы равны нулям передаточной функции разомкнутой системы. Для определения полюсов замкнутой системы необходимо решить уравнение

$$W_p(s) = -1. \quad (12)$$

Так как  $W_p(s)$  является функцией комплексного переменного  $s$ , то уравнение (12) распадается на два уравнения:

– уравнение модулей:

$$|W_p(s)| = 1, \quad (13)$$

– уравнение аргументов:

$$\left. \begin{aligned} \arg W_p(s) &= \pm (2\nu + 1)\pi, \quad \nu = 0, 1, 2, \dots \\ \arg W_p(s) &= \pm 2\pi, \quad \nu = 0, 1, 2, \dots \end{aligned} \right\} \quad (14)$$

соответственно для отрицательной и положительной обратных связей.

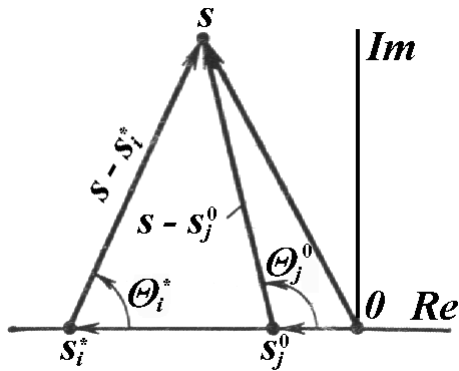


Рис. 1. Геометрическая интерпретация корневых годографов на плоскости корней

Уравнения (14) имеют наглядный геометрический смысл (рис. 1). Если точка  $s$  является полюсом замкнутой системы, то проводя в точку  $s$  вектора из всех нулей  $W_p(s)$  (аргументы этих векторов обозначены  $\theta_j^0$ ) и вектора из всех полюсов  $W_p(s)$  (аргументы обозначены  $\theta_i^*$ ), первое уравнение (14) можно записать в следующем виде:

$$\sum_{j=1}^m \theta_j^0 - \sum_{i=1}^n \theta_i^* = \pm (2\nu + 1)\pi, \quad \nu = 0, 1, 2, \dots, \quad (15)$$

а второе в виде:

$$\sum_{j=1}^m \theta_j^0 - \sum_{i=1}^n \theta_i^* = \pm 2\nu\pi, \quad \nu = 0, 1, 2, \dots \quad (16)$$

Углы  $\theta$  отсчитываются от положительного направления действительной оси. Знак угла "+" соответствует повороту против часовой стрелки, знак угла "-" соответствует повороту по часовой стрелке.

Траектории, описываемые на комплексной плоскости корнями характеристического уравнения замкнутой системы при плавном изменении одного из ее

параметров от 0 до  $\infty$ , называют корневым годографом.

Как следует из (15) и (16), конфигурация корневого годографа не зависит от коэффициента усиления  $k$ , но каждому конкретному значению  $k$  однозначно соответствуют точки на корневом годографе.

Для определения этого соответствия достаточно воспользоваться уравнением (13) в следующей интерпретации:

$$\frac{ka \prod_{j=1}^m l_j^0}{\prod_{i=1}^m l_i^*}, \quad (17)$$

где  $l_j^0$  – модуль (длина) вектора, проведенного из  $j$ -нуля в точку  $s$  корневого годографа;  $l_i^*$  – модуль вектора, проведенного из  $i$ -полюса в ту же точку  $s$ .

### **Свойства корневых годографов (случай отрицательной обратной связи)**

1. Ветви корневого годографа непрерывны и расположены на комплексной плоскости симметрично относительно действительной оси.

2. Число ветвей корневого годографа равно порядку системы  $n$ . Ветви начинаются в  $n$  полюсах разомкнутой системы при  $k = 0$ . При возрастании  $k$  от 0 до  $\infty$  полюса замкнутой системы двигаются по ветвям корневого годографа.

3. Отрезки действительной оси, по которым перемещаются действительные полюса замкнутой системы, являются действительными ветвями корневого годографа. Эти ветви находятся в тех частях действительной оси, справа от которых расположено нечетное общее число действительных полюсов и нулей разомкнутой системы.

4.  $m$  ветвей корневого годографа при возрастании  $k$  от 0 до  $\infty$  заканчиваются в  $m$  нулях  $W_p(s)$ , а  $(n - m)$  ветвей при  $k$ , стремящемся к  $\infty$ , удаляются от полюсов вдоль асимптот.

5. Асимптоты в виде звезды из  $(n - m)$  полупрямых выходят из точки с координатой

$$\sigma_a = \frac{\sum_{j=1}^m s_j^0 - \sum_{i=1}^n s_i^*}{n - m}$$

на действительной оси под углами

$$\theta_a = \frac{2\nu + 1}{n - m} \pi, \quad (\nu = \overline{0, n - m - 1})$$

к действительной оси.

6. Угол выхода  $\theta_i^*$  ветви корневого годографа из полюса  $s_i^*$  определяется из уравнения (15), примененного к данному полюсу. Аналогично определяется угол входа ветви корневого годографа в нуль  $s_j^0$ .

7. При расположении ветвей корневого годографа в левой полуплоскости  $s$  САУ устойчива. При пересечении ветвей корневого годографа мнимой оси слева направо САУ становится неустойчивой. Пусть при  $k = k^{кр}$  пересечение корневого годографа с мнимой осью произойдет в некоторой точке  $i\omega^{кр}$ . Назовем это значение коэффициента усиления критическим  $k^{кр}$ , а величину  $\omega^{кр}$  критической угловой частотой, на которой система становится неустойчивой.

Метод корневого годографа позволяет выбрать коэффициент усиления САУ, подобрать расположение полюсов и нулей передаточной функции корректирующих звеньев, определить параметры доминирующих полюсов САУ (ближайших к началу координат плоскости  $s$ ).

### **Использование приложения Control System Toolbox при проектировании систем автоматического управления методом корневых годографов**

Приложение Control System Toolbox вычислительной системы MATLAB предназначено для моделирования, анализа и проектирования непрерывных и дискретных систем автоматического управления. Здесь реализованы методы, основанные на использовании передаточных функций и моделей для переменных состояния. С помощью функций приложения можно выполнять анализ одномерных и многомерных объектов автоматического управления, получать временные и частотные характеристики, разрабатывать системы автоматического управления и др. Основными вычислительными объектами в Control System Toolbox являются:

- родительский объект (класс) LTI (Linear Time-Invariant System) – линейные стационарные системы;
- дочерние объекты (подклассы класса LTI) с четырьмя формами моделей:
  - a) в виде передаточных функций (*tf*-форма);
  - b) заданием нулей, полюсов и коэффициента передачи (*zpk*-форма);
  - c) в виде систем дифференциальных уравнений для переменных состояния (*ss*-форма);
  - d) в *frd*-форме – в виде набора (вектора) частот  $\omega_k$  и соответствующих значений комплексного коэффициента передачи  $W(j\omega_k)$ .

В состав приложения входит более 100 функций, которые можно объединить в следующие группы: 1) вызова графического интерфейса; 2) создания линейных стационарных систем; 3) извлечения данных; 4) получения информации об отдельных характеристиках модели; 5) преобразования моделей; 6) «ариф-

метических» операций с моделями; 7) моделей динамики; 8) временной задержки; 9) моделей для переменных состояния; 10) отклика во времени; 11) частотного отклика; 12) композиции систем; 13) традиционного проектирования систем; 14) аналитического конструирования регуляторов; 15) решения матричных уравнений; 16) демонстрационных программ.

### Функции вызова графического интерфейса

К рассматриваемой группе относятся функции **ltiview** и **rltool**.

С помощью функции **ltiview** открывается интерактивный обозреватель, известный из Simulink. Функция **rltool** открывает графический интерфейс, позволяющий проектировать корректирующее звено в замкнутой одномерной системе управления методом корневого годографа. Функция записывается в следующих формах:

**rltool**

**rltool(sys)**

**rltool(sys.comp)**

**rltool(sys.comp.LocationFlag, FeedbackSign)**

Здесь:

- **sys** – имя модели одномерного объекта;
- **comp** – имя (идентификатор) корректирующего звена-компенсатора;
- **LocationFlag** – переменная, задающая позицию компенсатора в системе:

1 – в прямом тракте системы, 2 – в цепи обратной связи;

- **FeedbackSign** – тип обратной связи (-1 – отрицательная обратная связь, 1 – положительная обратная связь).

Выполнение рассматриваемой функции без аргументов приводит к появлению основного окна интерфейса (рис. 2).

Работу с интерфейсом рассмотрим на примере. Из командного окна MATLAB выполним загрузку в рабочую область файла с набором LTI-моделей. Для этого в окне **Рабочая область** MATLAB необходимо загрузить файл данных **ltiexamples**, который находится в папке **toolbox** (**toolbox\control\control\ltiexamples.mat**).

Затем в меню открывшегося ранее окна (рис. 2) интерфейса выберем команду **File > Import** (**Файл > Импорт**). Выполнение данной команды меню приведет к открытию диалогового окна загрузки моделей и задания структуры системы (рис. 3).

В верхнем правом углу окна приведена структурная схема замкнутой системы. В ней **F**, **G**, **H**, **C** – звенья, модели которых можно загружать (с помощью переключателя **Import From**) из рабочей среды MATLAB, из mat-файла или из среды Simulink. Звено, обозначенное буквой **C**, – это компенсирующее динамическое звено, структуру и параметры которого нужно определить. В исходной схеме данное звено включено в прямой тракт замкнутой системы. Нажатием кнопки **Other** (Другая) можно перейти к альтернативной структуре, в



которой компенсатор введен в цепь обратной связи.

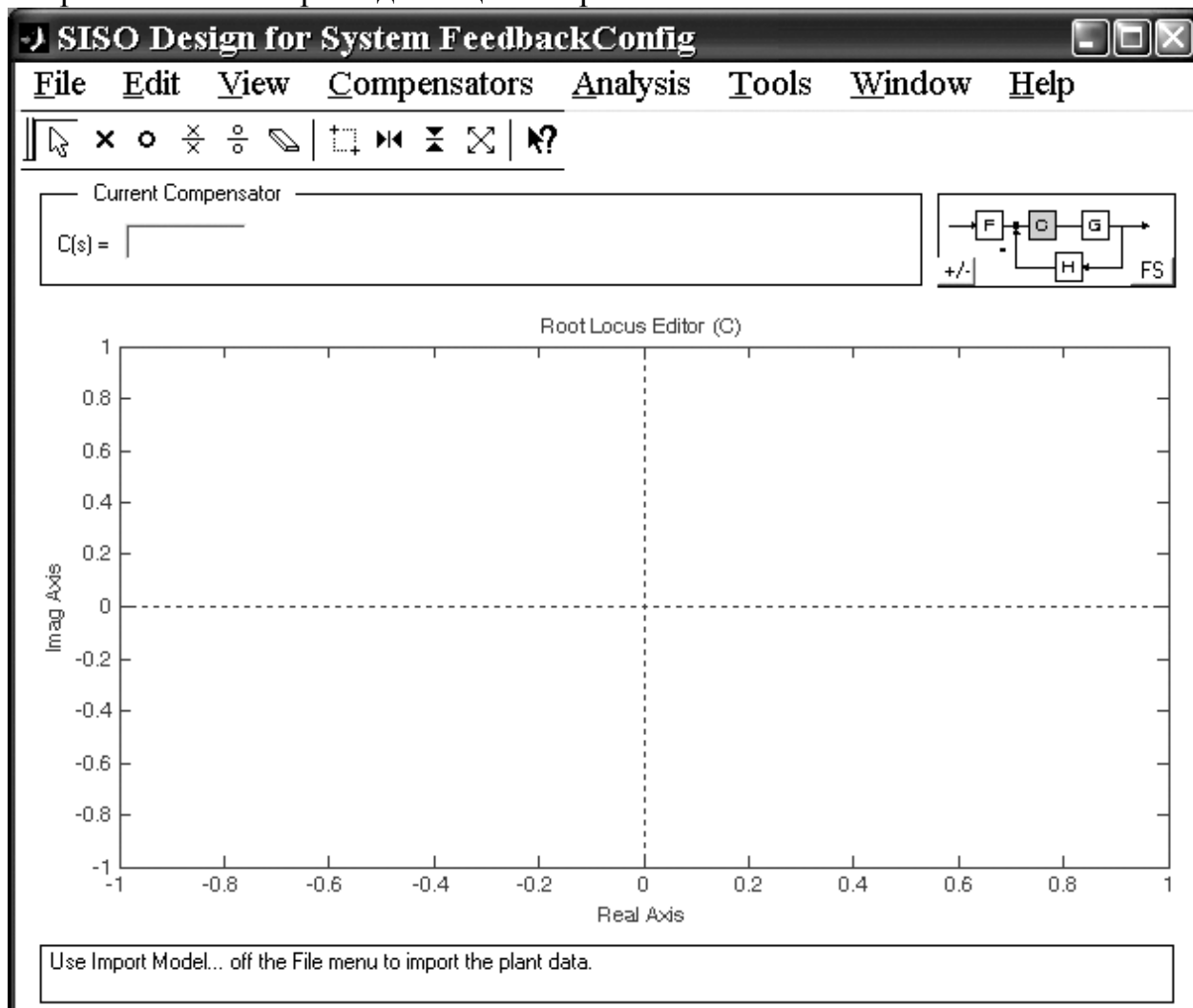


Рис. 2. Окно, открывающееся при использовании функции *rltool*

Сохранив исходную структуру, выберем модели для загрузки. В качестве модели звена **G** примем модель с именем **G**. Данный выбор проведем, выбрав в списке модель **G** и нажав кнопку со стрелкой, указывающей на звено **G** (в правой нижней части окна). Модели звеньев **F** и **H** изменять не будем (по умолчанию это пропорциональные звенья с единичным коэффициентом усиления). Зададим теперь какое-либо имя создаваемой системе, например SAUsys (в поле Name), подтвердим указанные действия нажатием кнопки ОК. Окно загрузки модели при этом закроется, и в основном окне интерфейса отобразится распределение нулей (кружочки) и полюсов (квадратики) замкнутой системы при изменении коэффициента усиления компенсатора **C** от указанного (заданного) значения до  $\infty$  (рис. 4).

Система с таким расположением полюсов не обладает хорошим качеством переходного процесса, поскольку они расположены слишком близко к границе устойчивости (к мнимой оси комплексной плоскости). В этом легко убедиться, просмотрев график переходного процесса замкнутой системы. Для этого нужно в

окне интерфейса метода выбрать в меню **Analysis** команду **Other Loop Responses**, что приведет к открытию окна интерактивного обозревателя (рис. 5).

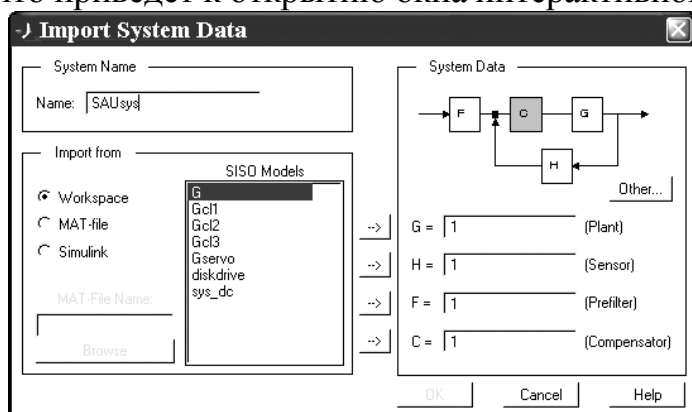


Рис. 3. Окно загрузки модели и задания структуры системы

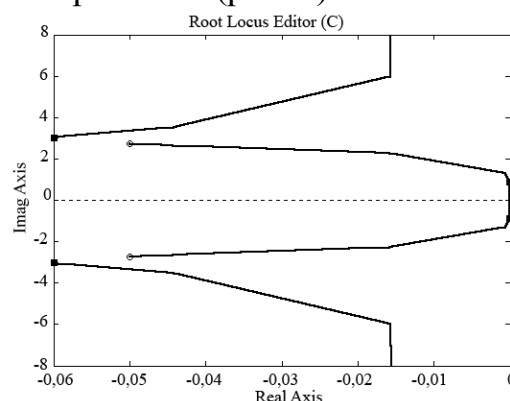


Рис. 4. Нули и полюсы замкнутой системы

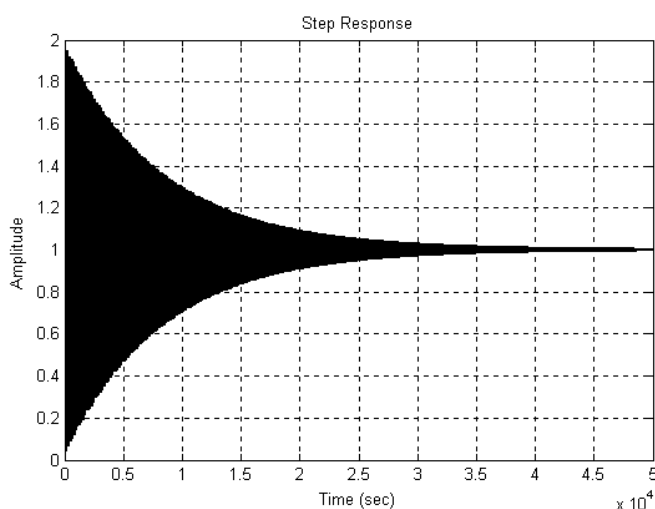


Рис. 5. Переходный процесс в нескорректированной системе

Как видно из графика переходного процесса, исходная система близка к границе устойчивости. Время затухания переходного процесса составляет около  $3,5 \cdot 10^4$  с. Если ввести корректирующее звено пропорционального типа, то существенно улучшить динамические свойства системы не удастся (при заданной структуре компенсатора нули и полюсы можно передвигать непосредственно на графике при помощи мыши; при этом соответственно изменяется значение

коэффициента усиления компенсатора в поле C). Поэтому применим компенсатор инерционно-форсирующего типа, добавив один полюс и один нуль с помощью кнопок с обозначением крестика и кружочка, расположенных над графическим окном слева, и разместив при этом (с помощью мыши) нули и полюсы, например, так, как это показано на рис. 6.

График переходного процесса (рис. 7) при такой коррекции существенно улучшается. Если такой вид переходного процесса нас устраивает, на этом процедуру синтеза можно и закончить. Заметим, что передаточная функция компенсатора в данном случае имеет вид  $K(s) = \frac{0,319(5s + 1)}{1,5s + 1}$ .

Интерфейс метода корневых годографов обладает многочисленными сервисными функциями. Если, например, дважды щелкнуть на блоке **G** структурной схемы, то откроется окно с информацией о моделях **F**, **H**, **G** (рис. 8).

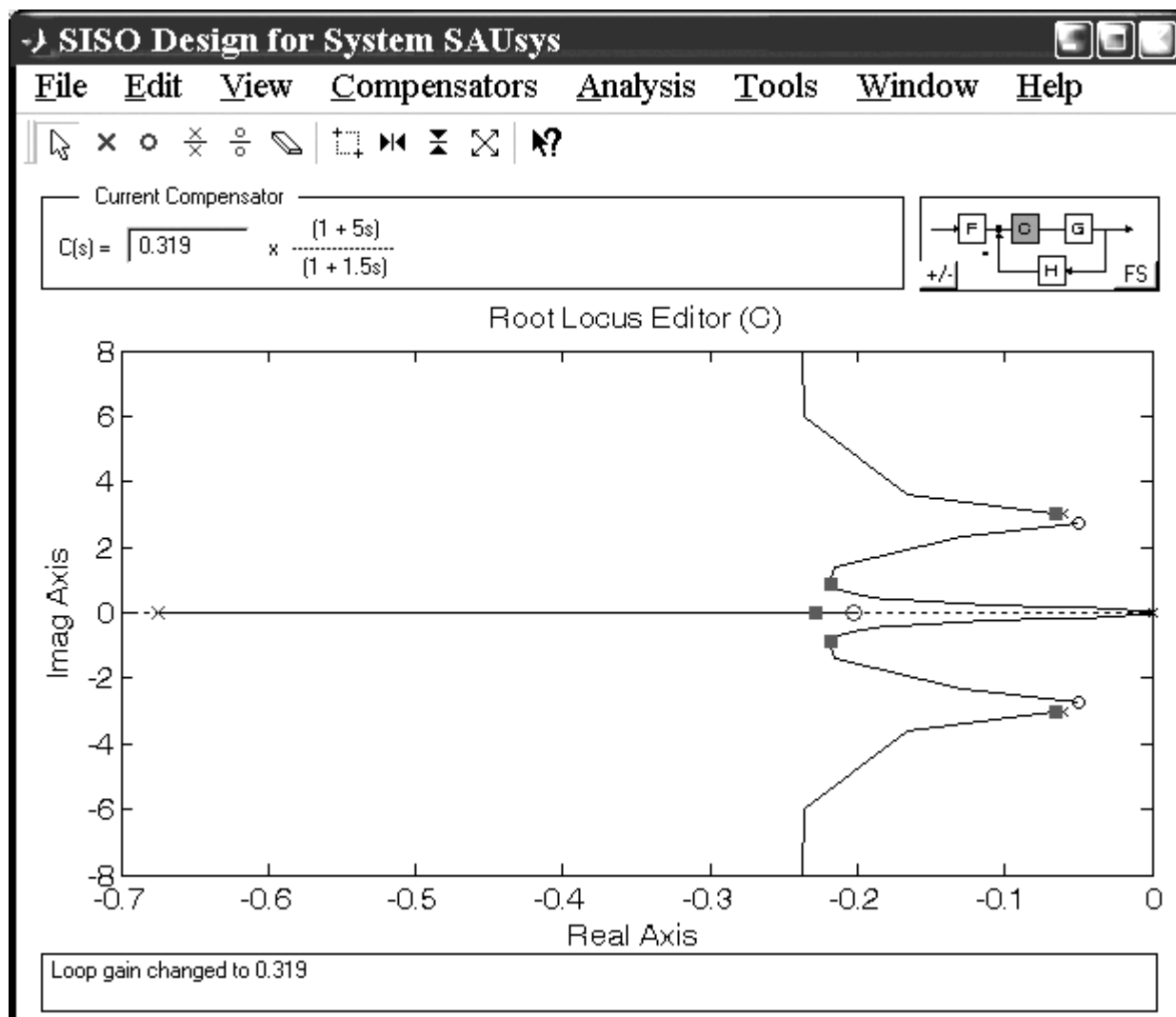


Рис. 6. Расположение нулей и полюсов после введения инерционно-форсирующего звена

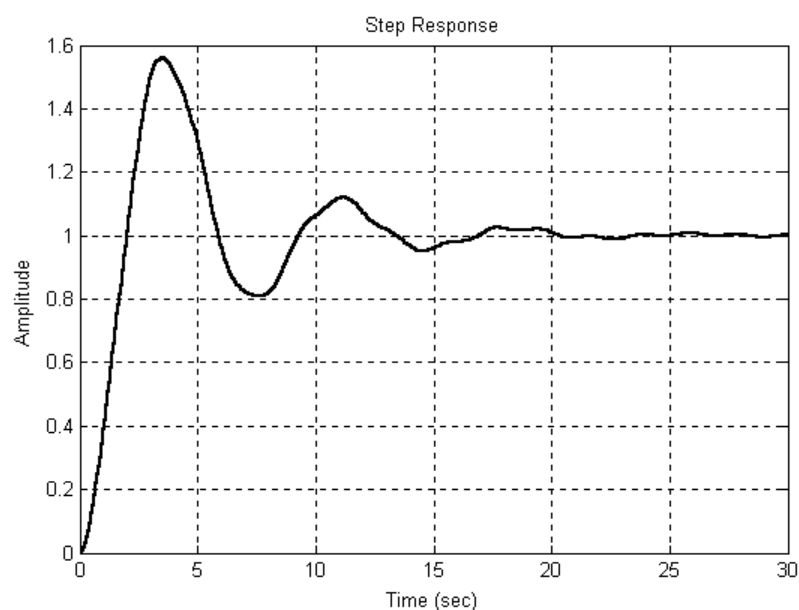


Рис. 7. Переходный процесс в скорректированной системе

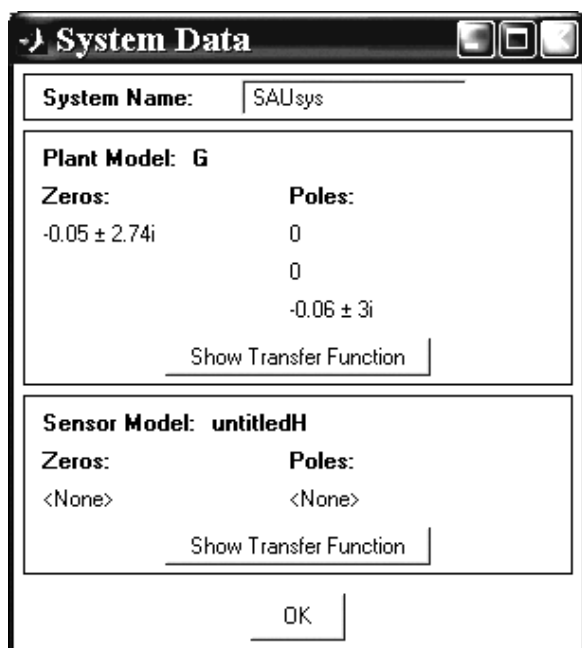


Рис. 8. Информационное окно о параметрах моделей блоков *G*, *H*

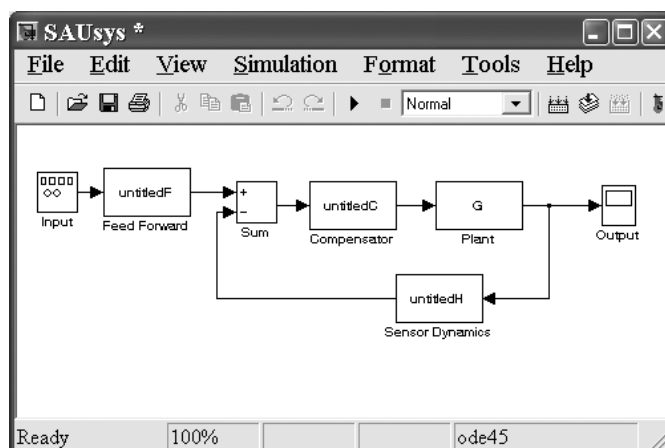


Рис. 9. Структурная схема исследуемой системы в среде Simulink

Довольно много возможностей предоставляет пользователю меню интерфейса. Например, используя команду меню **Утилиты / Рисовать Simulink диаграмму**, непосредственно перейдем к моделированию функционирования системы в среде **Simulink** (рис. 9).

### Указания к выполнению лабораторной работы

В работе исследуются системы автоматического управления, параметры которых заданы в табл. 4.1 методических указаний [1]. Для системы, состоящей из функционально необходимых элементов, с помощью метода корневых годографов необходимо определить параметры последовательного корректирующего устройства, выбрав в качестве критерия оптимального синтеза близость частотных характеристик скорректированной системы частотным характеристикам системы с желаемой ЛАЧХ. По передаточной функции последовательного корректирующего устройства определить передаточную функцию корректирующего устройства в виде звена отрицательной обратной связи, осуществив переход от последовательной коррекции к коррекции с помощью звена обратной связи. Выполнить сравнение передаточных функций звеньев, полученных двумя способами (частотным методом и методом корневых годографов), и сделать соответствующие выводы.

Формируя *S*-модель рассматриваемой системы и выполняя ее анализ с помощью интерактивного обозревателя, осуществим загрузку модели в рабочую область **MATLAB** с помощью команды **File → Export → Export to Workspace** (из меню интерактивного обозревателя). В режиме командной строки **MATLAB** командой **rltool** откроем окно графического интерфейса проектиро-

вания замкнутой одномерной системы автоматического управления методом корневого годографа. Далее выполним загрузку модели системы (в рассматриваемом случае модель имеет имя **System\_funkc\_1**). Система с расположением корней и полюсов, показанным на рис. 10, не будет обладать требуемыми показателями качества переходного процесса, что и подтверждается переходным процессом на рис. 11.

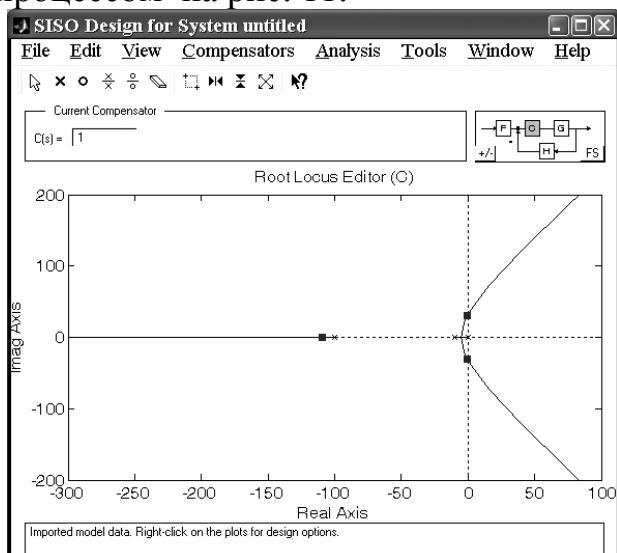


Рис. 10. Расположение нулей и полюсов системы, состоящей из функционально необходимых элементов

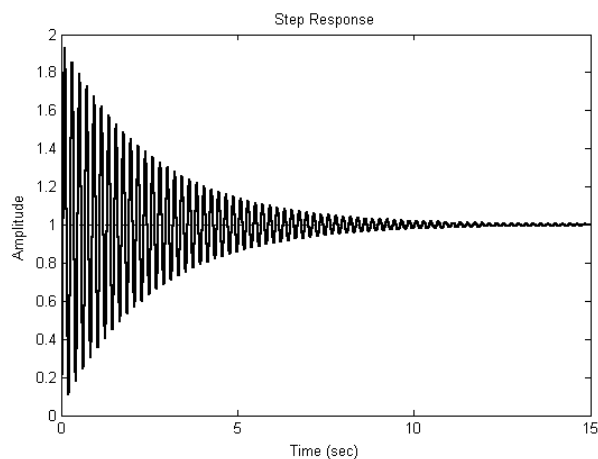


Рис. 11. Переходный процесс в системе, состоящей из функционально необходимых элементов

Если построить компенсатор с применением двух одинаковых звеньев инерционно-форсирующего типа (рис. 12), то переходный процесс в системе имеет вид, показанный на рис. 13.

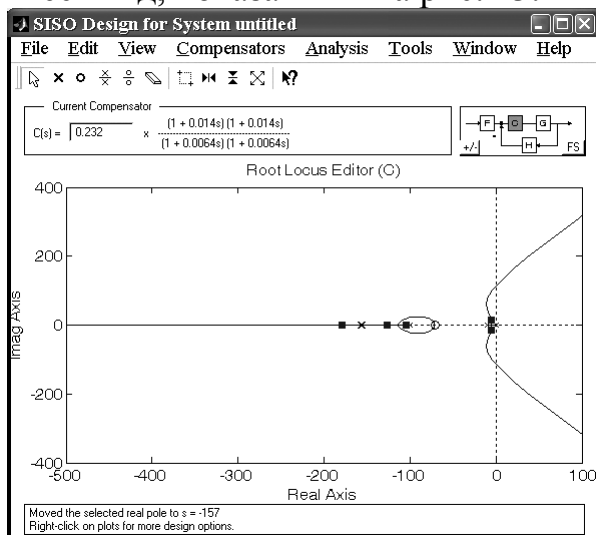


Рис. 12. Параметры последовательного корректирующего устройства (компенсатора)

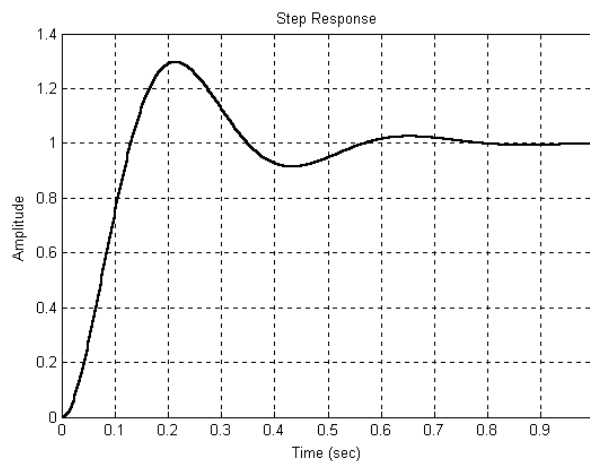


Рис. 13. Переходный процесс в скорректированной системе

## Лабораторная работа № 2

### Моделирование линейных стационарных систем автоматического управления в MATLAB (приложение Control System Toolbox)

#### Создание моделей стационарных систем

Для создания моделей линейных стационарных систем в приложении Control System Toolbox используется несколько функций:

- передаточные функции (*tf*-форма);
- нули, полюса и коэффициент передачи (*zpk*-форма);
- матрица пространства состояний (*ss*-форма);
- векторы частот и соответствующих значений комплексного коэффициента передачи (*frd*-форма).

При задании модели одномерной (SISO) системы в форме передаточной функции возможны следующие варианты:

```
» w = tf([1 2 3],[0.01 0.1 1])
```

Transfer function:

$$\frac{s^2 + 2s + 3}{0.01s^2 + 0.1s + 1}$$

где [1 2 3] и [0.01 0.1 1] – векторы-строки коэффициентов полиномов числителя и знаменателя, расположенных в порядке убывания степеней  $s$ ;

```
» w = tf([1 2 3],[0.01 0.1 1],0.05)
```

Transfer function:

$$\frac{z^2 + 2z + 3}{0.01z^2 + 0.1z + 1}$$

Sampling time: 0.05,

где 0,05 – период квантования, когда создаются дискретные модели.

Если создаются модели многомерных (MIMO) систем, то в отличие от рассмотренных одномерных систем аргументы передаточных функций содержат не вектора-строки, а массивы ячеек из векторов-строк. Создадим матрицу передаточных функций объекта, имеющего один вход и два выхода:

$$W(s) = \begin{bmatrix} \frac{0,5}{s^2 + 5s + 1} \\ \frac{1,5}{2s + 1} \end{bmatrix}.$$

```
» w=tf({0.5;1.5},{[1 5 1];[2 1]})
```

Transfer function from input to output...

0.5

#1: -----

$$s^2 + 5s + 1$$

$$\begin{array}{r} 1.5 \\ \#2: \text{-----} \\ 2 s + 1 \end{array}$$

Функция *zpk* создает модель в виде передаточной функции с указанием ее нулей и полюсов. Если в качестве аргумента функции *zpk* использовать передаточную функцию в других формах, то она преобразует произвольную LTI-модель в модель *zpk*. Например, для созданной выше матрицы передаточных функций модель системы в форме *zpk*:

```
» zpk(w)
Zero/pole/gain from input to output...
0.5
#1: -----
(s+4.791) (s+0.2087)
0.75
#2: -----
(s+0.5)
```

Функция *ss* формирует модель системы в форме пространства состояний. В качестве аргументов здесь выступают матрицы *a*, *b*, *c* и *d* переменных состояния. Если в качестве аргумента выступает передаточная функция, заданная в *tf*- или *zpk*-форме, то выполняется ее преобразование в *ss*-форму:

```
» w=tf([1 2 3],[4 5 6])
```

Transfer function:

$$s^2 + 2 s + 3$$

$$\text{-----}$$

$$4 s^2 + 5 s + 6$$

```
» ss(w)
```

a =

		x1	x2
x1		-1.25	-1.5
x2		1	0

b =

		u1
x1		1
x2		0

c =

		x1	x2
y1		0.1875	0.375

d =

		u1
y1		0.25

Continuous-time system.

Функция *frd* создает частотную модель объекта в *frd*-форме. В общем случае частотная модель создается командой:

```
w=frd(response, frequency, 'Units', units)
```

В качестве аргументов в функции выступают: *response* – вектор-строка (в случае ММО систем – массив) значений комплексного коэффициента передачи; *frequency* – вектор-строка частот  $\omega_k$  и *units* – строковая переменная, задающая размерность частоты (по умолчанию 'rad/c', другое возможное значение 'Hz').

### Преобразование моделей

К функциям преобразования моделей относятся следующие функции, для краткости описания указанные только именами.

- *tf* – преобразует LTI-модель в *tf*-форму;
- *zpk* – преобразует LTI-модель в *zpk*-форму;
- *ss* – преобразует LTI-модель в *ss*-форму;
- *frd* – преобразует LTI-модель в *frd*-форму;
- *chgunits* – изменяет размерность частоты в *frd*-модели;
- *c2d* – преобразует непрерывную модель в дискретную;
- *d2c* – преобразует дискретную модель в непрерывную;
- *d2d* – изменяет интервал дискретизации в дискретной модели.

### «Арифметические» операции с моделями

Функции этой группы позволяют создавать и изменять структуры динамических систем. Внешне они подобны арифметическим операторам, но операндами в данном случае являются LTI-модели.

- *plus* (*minus*) или + (-) – сложение (вычитание) LTI-моделей (параллельное соединение);
- *mtimes* или \* – умножение LTI-моделей (последовательное соединение);
- \ – левое деление ( $\text{sys1} \backslash \text{sys2}$  равносильно  $\text{inv}(\text{sys1}) * \text{sys2}$ );
- / – правое деление ( $\text{sys1} / \text{sys2}$  равносильно  $\text{sys1} * \text{inv}(\text{sys2})$ );
- ^ – возведение LTI-модели в степень (последовательное соединение нескольких одинаковых LTI-моделей);
- ' – операция *pertransposition*, означающая замену матрицы системы  $\mathbf{A}(p)$  на матрицу  $[\mathbf{A}(-p)]^T$ ; для дискретных моделей – замена матрицы  $\mathbf{A}(z)$  на матрицу  $\mathbf{A}(z^{-1})^T$ ;



- `'` – транспонирование модели;
- `[..]` – горизонтальное/вертикальное объединение LTI-моделей, т.е. такое их объединение, когда входы становятся общими, а выходы остаются независимыми. Для такого объединения необходимо, чтобы число входов объединяемых моделей было одинаковым, тогда число входов в результирующей модели останется таким же, как и в каждой из объединяемых моделей, а число выходов будет равно сумме выходов объединяемых моделей;
- `inv` – обращение LTI-модели (замена входов на выходы и наоборот);
- `feedback` – соединение двух звеньев, при котором второе звено составляет цепь отрицательной обратной связи для первого звена.

Рассмотрим в качестве примера создание модели линейной стационарной системы с передаточной функцией в разомкнутом состоянии

$W(s) = \frac{100(s+1)^2}{s(5s+1)^2(0.1s+1)}$ , охваченной единичной отрицательной обратной связью.

```

» W1=tf([0 100],[1 0]);
» W2=tf([1 1],[5 1]);
» W3=tf([1 1],[5 1]);
» W4=tf([0 1],[0.1 1]);
» W5=W1*W2*W3*W4;
» Woc=1;
» sys=feedback(W5, Woc)
Transfer function:
          100 s^2 + 200 s + 100
-----
2.5 s^4 + 26 s^3 + 110.1 s^2 + 201 s + 100

```

Эту же систему более просто можно создать с помощью приложения Simulink (рис. 14).

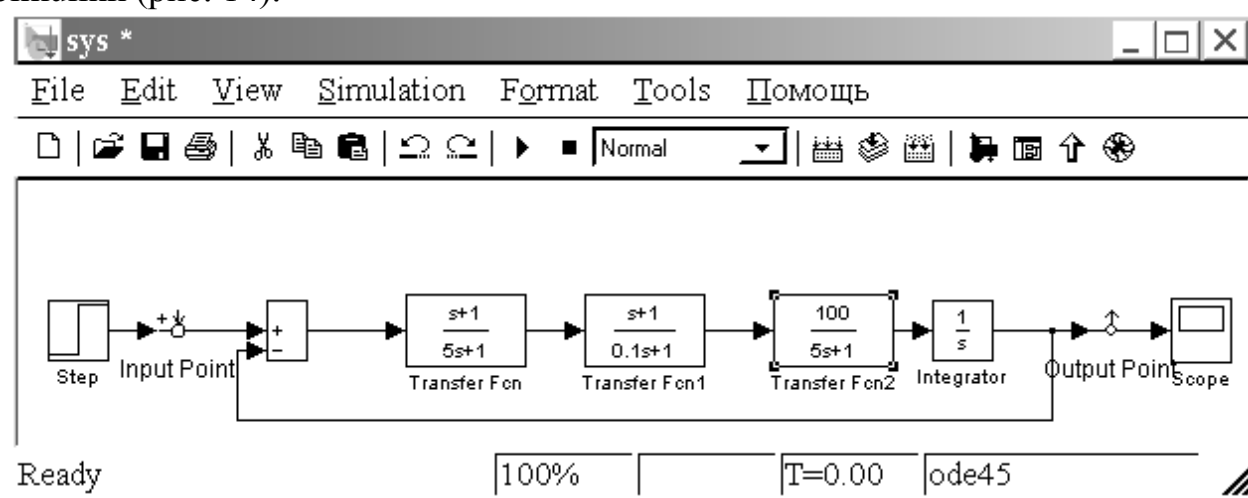


Рис. 14. S-модель системы автоматического управления

Если теперь выполнить анализ системы с помощью интерактивного обозревателя *ltiview*, который можно вызвать командой **ltiview** из главного окна MATLAB, или с помощью команды **Linear Analysis** меню **Tools** созданной S-модели. После выполнения команды на экране появляется окно **LTI Viewer**, из меню **File** которого с помощью команды **Export** необходимо осуществить загрузку модели в рабочее пространство. Далее с помощью функции `tf(sys_1)`, где `sys_1` имя системы, под которым она загружена в рабочее пространство, из командного окна MATLAB можно найти нужную передаточную функцию системы:

```
» tf(sys_1)
Transfer function from input "Input Point" to output
"Output Point":
          40 s^2 + 80 s + 40
-----
s^4 + 10.4 s^3 + 44.04 s^2 + 80.4 s + 40
```

Полученное выражение полностью совпадает с найденной ранее передаточной функцией замкнутой системы (после сокращения числителя и знаменателя на 2,5).

Если потребуется найти передаточную функцию по ошибке, то достаточно в S-модели назначить другой выход (рис. 2).

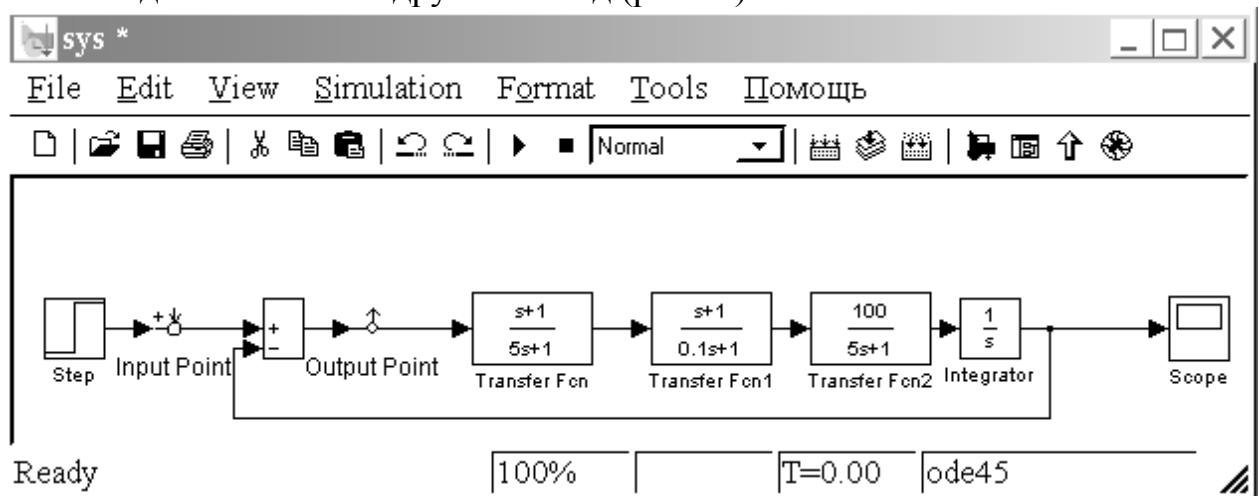


Рис. 15. S-модель системы для определения передаточной функции по ошибке

```
» tf(sys_2)

Transfer function from input "Input Point" to output
"Output Point":
s^4 + 10.4 s^3 + 4.04 s^2 + 0.4 s + 2.842e-015
-----
s^4 + 10.4 s^3 + 44.04 s^2 + 80.4 s + 40
```

Анализ показывает, что последнее слагаемое числителя пренебрежимо мало по сравнению с другими, более того, оно должно отсутствовать в выра-

жении передаточной функции системы по ошибке. Видимо, появление этого слагаемого следует отнести к недостаткам используемого приложения MATLAB.

Подобным образом очень легко “свертываются” структурные схемы сложных многоконтурных систем. Рассмотрим определение передаточной функции датчика натяжения нити с вращающимися нитепроводами [2]. При использовании подсистем и пренебрежении моментом сухого трения в опорах датчика его S-модель имеет вид (рис. 16).

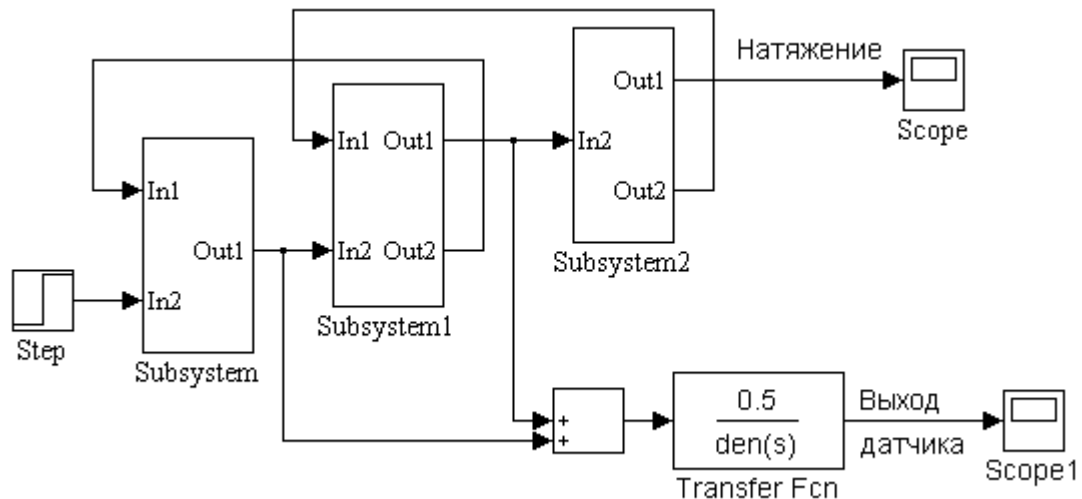


Рис. 16. S-модель датчика натяжения нити

Передаточная функция по каналу “натяжение-выход датчика” легко получается при использовании из командного окна MATLAB функции `tf(datchik_lin_analysis_1)`:

```
tf(datchik_lin_analysis_1)
```

Transfer function from input "Input Point" to output "Output Point":

```
4.636e004s^5+6.074e006s^4+1.941e008s^3+2.643e009s^2+
```

```
----->
s^8+146s^7+1.029e004s^6+4.97e005s^5+1.618e007s^4+
```

```
+1.66e010s+3.976e010
```

```
----->
+3.089e008s^3+3.265e009s^2+1.772e010s+3.859e010
```

С помощью полученной передаточной функции можно легко получить переходную и частотные характеристики датчика:

```
» step(tf(datchik_lin_analysis_1))
» bode(tf(datchik_lin_analysis_1))
```

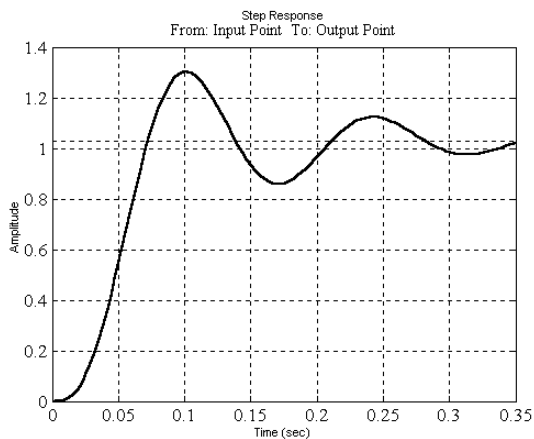


Рис. 17. Переходная характеристика датчика

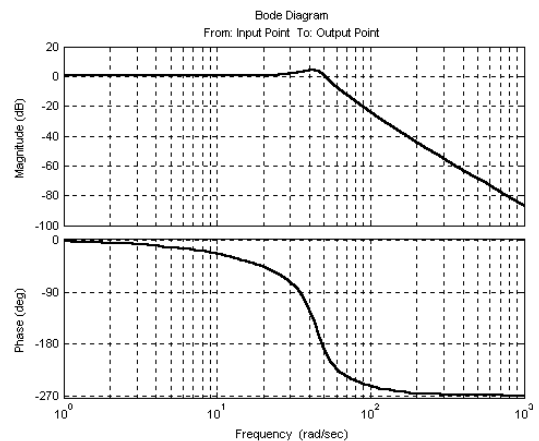


Рис. 18. Частотные характеристики датчика

Сравнение характеристик, показанных на рис. 17 и 18, с аналогичными характеристиками, полученными с помощью приложения Simulink [2], показывает их полное совпадение.

### Указания к выполнению лабораторной работы

Для скорректированных систем, исследованных в лабораторной работе № 1, создать с помощью *tf*-функции модель системы и определить передаточные функции по выходной величине и по ошибке. Далее определить указанные передаточные функции с помощью *S*-модели системы и сравнить полученные результаты.

Выполнить переход от *tf*-формы к *zpk*-форме и определить заданные передаточные функции в *zpk*-форме. Сравнить результаты, находя с помощью MATLAB корни полиномов числителя и знаменателя передаточной функции системы в *tf*-форме.

Найти аналитически указанные передаточные функции и сравнить их с полученными с помощью моделей.

Рассмотрим пример выполнения лабораторной работы. Пусть структурная схема системы имеет вид (рис. 19):

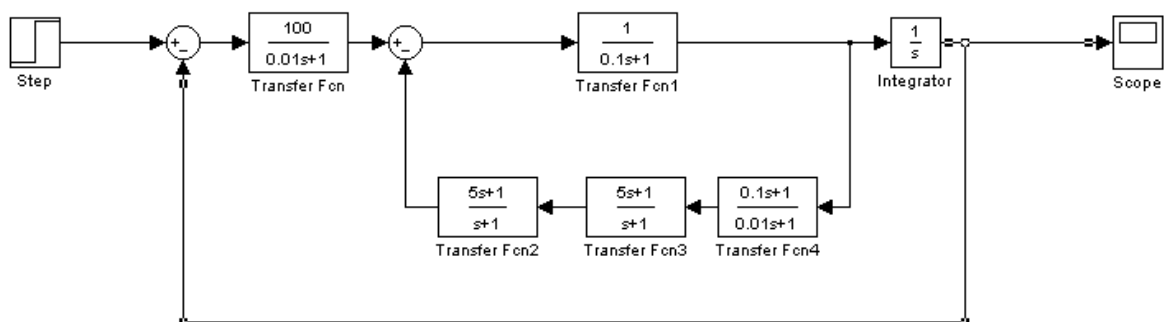


Рис. 19. *S*-модель системы автоматического управления

Модель в *tf*-форме имеет вид:

```

» W1=tf([0 1],[0.1 1]);
» W2=tf([5 1],[1 1]);
» W3=tf([5 1],[1 1]);
» W4=tf([0.1 1],[0.01 1]);
» W5=W2*W3*W4;
» W6=feedback(W1,W5);
» W7=tf([0 1],[0.01 1]);
» W8=tf([0 1],[1 0]);
» W9=100;
» W10=W7*W6*W8*W9;
» Woc=1;
» sys=feedback(W10,Woc)

```

Transfer function:

$$\frac{s^3 + 102s^2 + 201s + 100}{1e-005s^6 + 0.02712s^5 + 2.884s^4 + 28.34s^3 + 114.2s^2 + 203s + 100}$$

Получим эту же передаточную функцию с помощью S-модели:

```

» sys1=tf(sausys_1)

```

Transfer function from input "Input Point" to output "Output Point":

$$\frac{1e005s^3 + 1.02e007s^2 + 2.01e007s + 1e007}{s^6 + 2712s^5 + 2.884e005s^4 + 2.834e006s^3 + 1.142e007s^2 + 2.03e007s + 1e007}$$

Передаточные функции скорректированной системы по ошибке определим аналогично. Если воспользоваться представлением модели системы в *tf*-форме, то

```

» sys1=feedback(Woc,W10)

```

Transfer function:

$$\frac{1e-005s^6 + 0.02712s^5 + 2.884s^4 + 27.34s^3 + 12.23s^2 + 2s}{1e-005s^6 + 0.02712s^5 + 2.884s^4 + 28.34s^3 + 114.2s^2 + 203s + 100}$$

При использовании S-модели имеем:

```

» sys1=tf(sausys_2)

```

Transfer function from input "Input Point" to output "Output Point":

$$\frac{s^6 + 2712s^5 + 2.884e005s^4 + 2.734e006s^3 + 1.223e006s^2 + 2e005s}{s^6 + 2712s^5 + 2.884e005s^4 + 2.834e006s^3 + 1.142e007s^2 + 2.03e007s + 1e007}$$

Модель этой же системы в zpk-форме:

```

» zpk(W1)
Zero/pole/gain:
10
-----
(s+10)
» zpk(W2)
Zero/pole/gain:
5 (s+0.2)
-----
(s+1)
» zpk(W3)
Zero/pole/gain:
5 (s+0.2)
-----
(s+1)
» zpk(W4)
Zero/pole/gain:
10 (s+10)
-----
(s+100)
» zpk(W5)
Zero/pole/gain:
250 (s+10) (s+0.2)^2
-----
(s+100) (s+1)^2
» zpk(W6)
Zero/pole/gain:
10 (s+100) (s+1)^2
-----
(s+2602) (s+10) (s^2 + 0.4616s + 0.07688)
» zpk(W7)
Zero/pole/gain:
100
Static gain.
» zpk(W9)
Zero/pole/gain:
1
-
s
» zpk(W10)
Zero/pole/gain:
100000 (s+100) (s+1)^2
-----
s (s+2602) (s+100) (s+10) (s^2 + 0.4616s + 0.07688)

```

```

» zpk(Woc)
Zero/pole/gain:
1
Static gain.
» zpk(sys)
Zero/pole/gain:
100000 (s+100) (s+1) ^2
-----
(s+2602) (s+100) (s+3.731) (s+0.7629) (s^2+5.953s+13.5)
» zpk(sys1)
Zero/pole/gain from input "Input Point" to output
"Output Point":
s (s+2602) (s+100) (s+10) (s^2 + 0.4616s + 0.07688)
-----
(s+2602) (s+100) (s+3.731) (s+0.7629) (s^2+5.953s+13.5)
В полученных передаточных функциях sys – передаточная функция сис-
темы по выходной величине, а sys1 – по ошибке.
Для определения корней полиномов числителя и знаменателя передаточ-
ных функций воспользуемся функцией roots:
» x=[1 102 201 100]; d=roots(x)
d =
-100.0000
-1.0000
-1.0000
» x=[1e-005 0.0271 2.884 28.34 114.2 203 100];
d=roots(x)
d =
1.0e+003 *
-2.5995
-0.1001
-0.0030 + 0.0022i
-0.0030 - 0.0022i
-0.0037
-0.0008

```

Таким образом, передаточные функции, полученные в формах *tf* и *zpk*, совпадают.

### Лабораторная работа № 3

#### Моделирование линейных цифровых систем автоматического управления в MATLAB

Рассмотрим особенности моделирования цифровых систем автоматического управления на примере системы позиционирования магнитной головки жесткого диска ПЭВМ. На основании уравнения Ньютона дифференциальное уравнение вращательного движения магнитной головки имеет вид:

$$J \frac{d^2\varphi(t)}{dt^2} + C \frac{d\varphi(t)}{dt} + K\varphi(t) = K_i i(t), \quad (18)$$

где  $J$  - момент инерции рычага, на котором установлен блок магнитной головки;

$C$  - коэффициент вязкого трения в опоре рычага;

$K$  - коэффициент жесткости пружины;

$K_i$  - коэффициент передачи двигателя (по моменту);

$i(t)$  - ток якоря двигателя;

$\varphi(t)$  - угловое положение рычага головки.

Если принять за входную величину ток двигателя  $i(t)$ , а за выходную - угол поворота  $\varphi(t)$ , то, применяя к уравнению (1) преобразование Лапласа, получим

$$W(s) = \frac{K_i}{Js^2 + Cs + K}. \quad (19)$$

Создадим  $tf$ -модель при следующих исходных данных:  $J = 0,01$  кг·м<sup>2</sup>,  $K = 10$  Н·м/рад,  $C = 0,004$  Н·м/(рад/с),  $K_i = 0,05$  Н·м/А:

```
» J = 0.01; C = 0.004; K = 10; Ki = 0.05;
» num = Ki;
» den = [J C K];
» H = tf(num,den)
Transfer function:
      0.05
```

```
-----
0.01s^2 + 0.004s + 10
```

В рассматриваемой примере ставится задача проектирования цифровой системы позиционирования магнитной головки жесткого диска. Поскольку система управления имеет в своем составе аналого-цифровой преобразователь, то, переходя от непрерывной модели объекта к дискретной, получим:

```
» Ts = 0.005;
» Hd = c2d(H, Ts, 'zoh')
Transfer function:
6.233e-005z + 6.229e-005
-----
      z^2 - 1.973z + 0.998
Sampling time: 0.005
```



Для преобразования непрерывной модели в дискретную здесь использована функция `c2d`, у которой период квантования  $T_s = 0.005$  с, а для дискретизации выбран метод `'zoh'`. Проверку качества выполненной дискретизации непрерывного объекта можно выполнить, рассмотрев частотные характеристики непрерывной и дискретной моделей объектов (рис. 20):

```
» bode (H, '-', Hd, '--')
```

Если в дискретной системе автоматического позиционирования магнитной головки жесткого диска не использовать корректирующие устройства, то переходный процесс, определяемый функцией **step**, имеет вид (рис. 21):

```
» step (Hd)
```

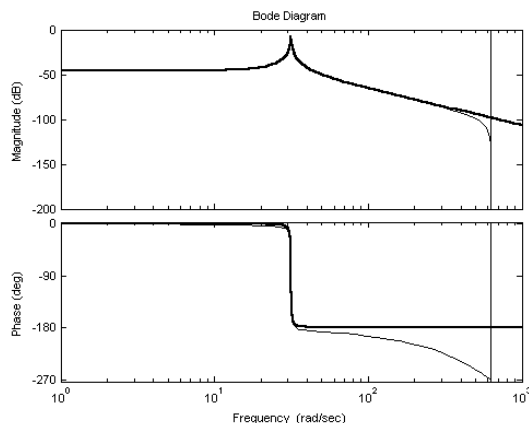


Рис. 20. Частотные характеристики объекта

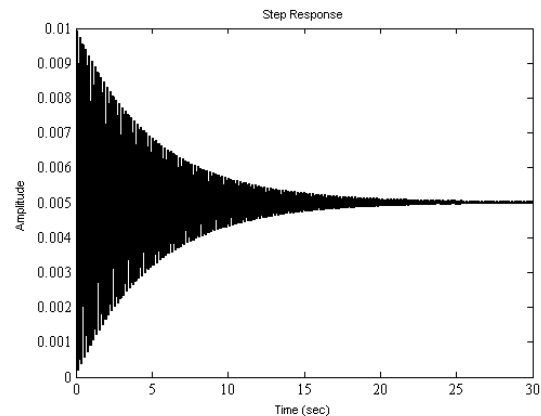


Рис. 21. Переходная характеристика объекта

Большая длительность переходного процесса системы и высокая колебательность свидетельствуют о ее слабом демпфировании. В дискретной системе близость полюсов передаточной функции к окружности единичного радиуса свидетельствует о неудовлетворительных показателях устойчивости системы. Получим с помощью функции **damp** собственные значения рассматриваемой системы и соответствующие им собственную частоту и коэффициент демпфирования:

```
» damp (Hd) ;
```

Eigenvalue	Magnitude	Equiv.Damping	Equiv. Freq. (rad/s)
9.87e-001+1.57e-001i	9.99e-001	6.32e-003	3.16e+001
9.87e-001-1.57e-001i	9.99e-001	6.32e-003	3.16e+001

Отсюда видно, что полюса расположены вблизи окружности единичного радиуса и система, вследствие этого, имеет слабое демпфирование. Таким образом, необходим дискретный регулятор, который увеличит демпфирование в системе. Решим поставленную задачу с помощью метода корневого годографа (МКГ) (см. лабораторную работу № 1). Для этого с помощью функции **rltool**

выполним загрузку основного окна интерфейса МКГ. Далее из меню **File** с помощью **Import Model** импортируем модель **Hd** в блок объекта **G**. В результате получим распределение (рис. 22) нулей и полюсов модели.

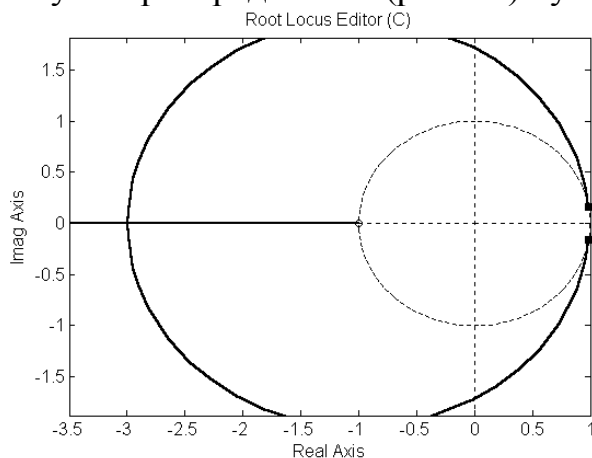


Рис. 22. Распределение нулей и полюсов модели объекта

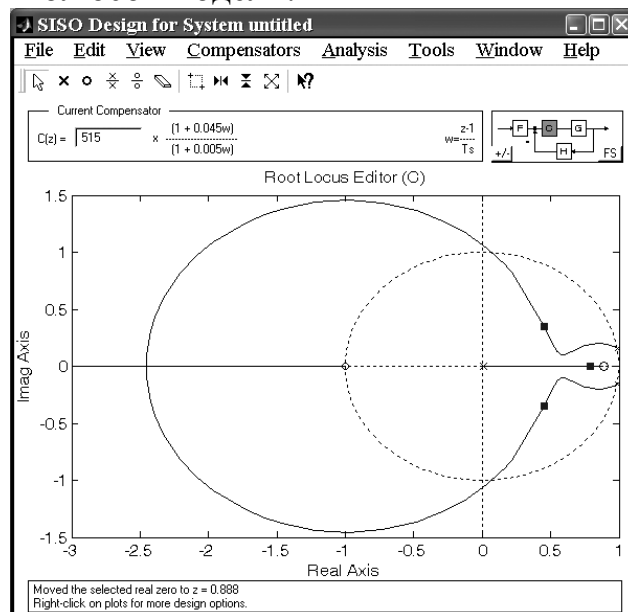


Рис. 23. Распределение нулей и полюсов модели системы управления

Если выбрать параметры компенсатора с передаточной функцией, показанной на рис. 23, то получим переходный процесс в системе, изображенный на рис. 24.

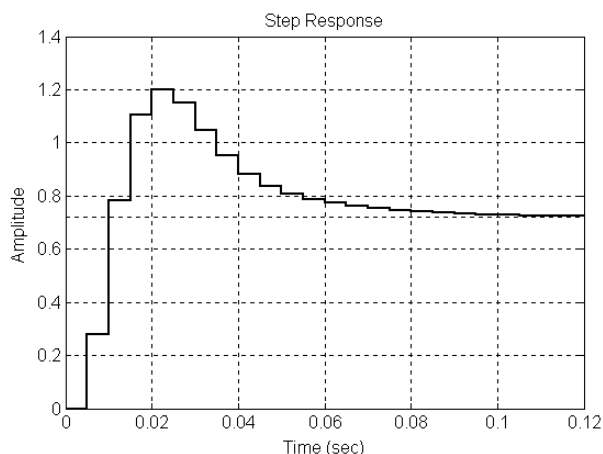


Рис. 24. Переходный процесс в системе с компенсатором

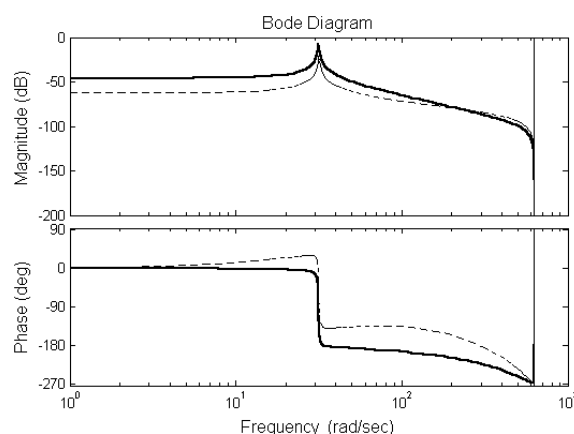


Рис. 25. Частотные характеристики исходной и скорректированной систем

Если параметры полученного переходного процесса не отвечают требованиям на проектирование, то, изменяя положение полюса и значение коэффициента передачи компенсатора, получают оптимальный переходный процесс. В частности, полученное быстродействие системы (около 0,1 с) не отвечает современным требованиям к управлению магнитными головками. Можно попытаться усложнить передаточную функцию компенсатора, добавив нули и полю-

са, хотя демпфирование магнитной головки низкое и добиться высокого быстродействия затруднительно.

Синтез дискретного корректирующего устройства рассматриваемой системы можно выполнить с помощью функций Control System Toolbox. В частности после определения дискретной передаточной функции объекта дискретная модель компенсатора может быть получена с помощью функции

```
» Komp=tf([1 -0.85],[1 0],Ts)%Формирование модели компенсатора
```

```
Transfer function:
```

```
z - 0.85
```

```
-----
```

```
z
```

```
Sampling time: 0.005
```

```
» Hp=Hd*Komp %Формирование разомкнутого контура
```

С помощью функции

```
» bode(Hd, '--', Hp, '-') )
```

можно выяснить, как модифицирует выбранный компенсатор частотные характеристики системы (рис. 25).

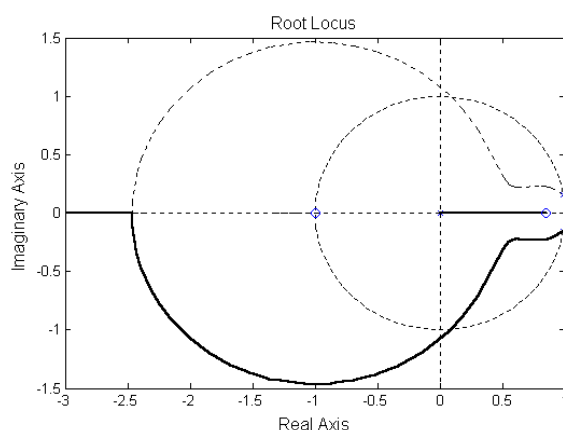


Рис. 26. Корневой годограф контура с компенсатором

Анализируя частотные характеристики системы, видим, что компенсатор сдвигает фазовую характеристику вверх (вводится опережение по фазе).

Корневой годограф контура с компенсатором (рис. 26) можно определить с помощью функции

```
» rlocus(Hp), zgrid
```

Далее используя функцию

```
» k,poles]=rlocfind(Hp),
```

задают оптимальное расположение полюсов:

```
Select a point in the graphics window
```

```
selected_point =
```

```
0.2995 - 0.1024i
```

```
k = 3.5241e+003
```

```
poles =
```

```
0.7081 + 0.2276i
```

```
0.7081 - 0.2276i
```

```
0.3373
```

Определим демпфирование замкнутой системы:

```
» ddamp(poles,Ts)
```

Eigenvalue	Magnitude	Equiv.Damping	Equiv. Freq. (rad/s)
7.08e-001+2.28e-001i	7.44e-001	6.90e-001	8.59e+001
7.08e-001-2.28e-001i	7.44e-001	6.90e-001	8.59e+001
3.37e-001	3.37e-001	1.00e+000	2.17e+002

Чтобы оценить результаты синтеза системы, сформируем замкнутую систему и построим для нее переходную функцию:

```

» Hsam=feedback(Hp,k)
Transfer function:
6.233e-005 z^2 + 9.308e-006 z - 5.294e-005
-----
z^3 - 1.753 z^2 + 1.031 z - 0.1866
Sampling time: 0.005
» step(Hsam)

```

Для определения показателей устойчивости системы воспользуемся функцией **margin**, предварительно учтя в передаточной функции разомкнутой системы найденной с помощью функции **rlocfind** значение  $k$ :

```

» olk=k*Hp;
» [Gm,Pm,Wcg,Wcp]=margin(olk);
» Margins=[Gm Wcg Pm Wcp]
Margins =
4.4302 296.7989 44.5498 94.8787

```

### Указания к выполнению лабораторной работы

Для следящей системы, состоящей из функционально необходимых элементов (см. лабораторную работу № 1), синтезировать дискретные корректирующие устройства:

- а) последовательного типа;
- б) в виде звена обратной связи,

обеспечивающие качество переходного процесса не хуже, чем при коррекции звеном непрерывного действия.

Рассмотрим пример выполнения лабораторной работы. Пусть передаточная функция нескорректированной системы в разомкнутом виде системы имеет вид:  $W(s) = \frac{k_v}{s(T_1s+1)(T_2s+1)}$ , где  $k_v = \omega_3 = 100 \text{ с}^{-1}$ ,  $T_1 = 1/\omega_2 = 0,1 \text{ с}^{-1}$ ,  $T_2 = 1/\omega_3 = 0,01 \text{ с}^{-1}$ . Представим эту передаточную функцию в виде трех последовательно соединенных звеньев:

```

» W1=tf([0 1],[1 0]);
» W2=tf([0 100],[0.1 1]);
» W3=tf([0 1],[0.01 1]);

```

```

» Wp=W1*W2*W3
Transfer function:
      100
-----
0.001 s^3 + 0.11 s^2 + s

```

Для перехода от непрерывной модели к дискретной используем аналого-цифровой преобразователь (АЦП) в виде экстраполятора нулевого порядка с периодом квантования 0,01 с.

```

» Ts=0.01;
» Hd=c2d(Wp,Ts,'zoh')
Transfer function:
0.01287 z^2 + 0.03984 z + 0.007441
-----
z^3 - 2.273 z^2 + 1.606 z - 0.3329
Sampling time: 0.01

```

С помощью функции **bode** выясним, насколько отличаются частотные характеристики исходной непрерывной **Wp** и полученной **Hd** моделей (рис. 27):

```

» bode(Hd, '--', Hp, '-')

```

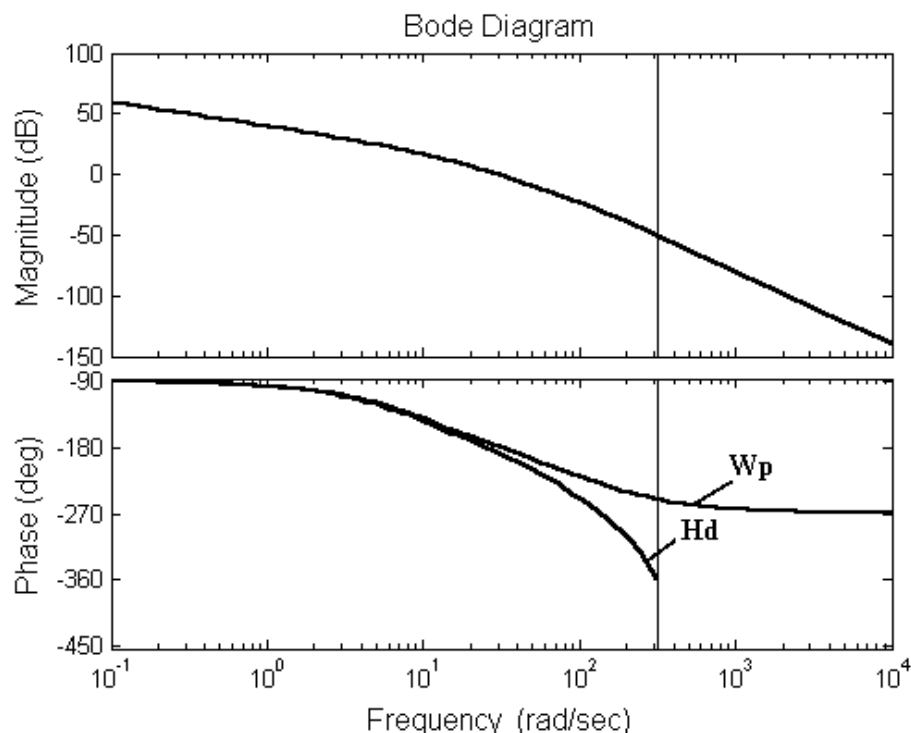


Рис. 27. Частотные характеристики непрерывной **Wp** и дискретной **Hd** моделей

Анализируя полученные кривые, отмечаем совпадение частотных характеристик в существенной области частот.

Сформируем разомкнутую цепь системы как последовательное соедине-

ние моделей объекта и компенсатора. Модель компенсатора принимаем в виде:

```
» Komp=tf([0.04 -0.0792 0.0392],[1 -1.996 0.996],Ts);
```

Transfer function:

$$\frac{0.04 z^2 - 0.0792 z + 0.0392}{z^2 - 1.996 z + 0.996}$$

Передаточная функция разомкнутой системы

```
» Hp=Hd*Komp
```

Transfer function:

$$\frac{0.0005149z^4 + 0.0005744z^3 - 0.002352z^2 + 0.0009715z + 0.0002915}{z^5 - 4.269 z^4 + 7.138z^3 - 5.801z^2 + 2.264z - 0.3315}$$

Sampling time: 0.01

В обратной связи принимаем звено

```
» Woc=1;
```

Передаточная функция замкнутой системы

```
» Hs=feedback(Hp,Woc)
```

Transfer function:

$$\frac{0.0005149z^4 + 0.0005744z^3 - 0.002352z^2 + 0.0009715z + 0.0002915}{z^5 - 4.268z^4 + 7.139z^3 - 5.804z^2 + 2.265z - 0.3313}$$

Sampling time: 0.01

Переходная характеристика (рис. 28):

```
» step(Hs, 1.8)
```

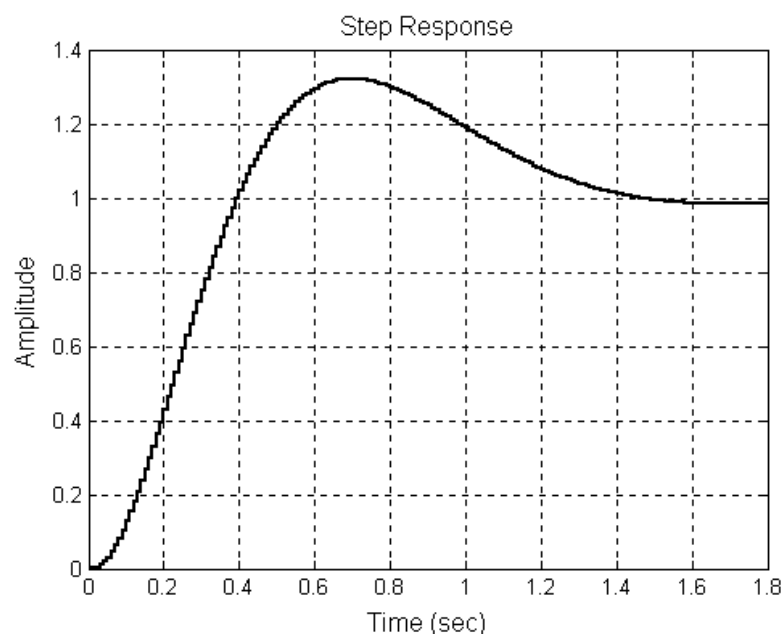


Рис. 29. Переходная характеристика скорректированной системы

Аналогично можно выполнить коррекцию с помощью дискретного звена обратной связи, охватив им наиболее инерционное звено системы.

## Список литературы

1. Кулида, Н.А. Исследование линейных систем автоматического управления и их элементов с помощью вычислительной системы MATLAB (версия 6.5). – Иваново: ИГТА, 2005. – 40 с.
2. Кулида, Н.А. Применение пакета Simulink 5 вычислительной системы MATLAB (версия 6.5) для моделирования элементов систем автоматического управления. – Иваново: ИГТА, 2005. – 68 с.

## Оглавление

### Лабораторная работа № 1

<b>Синтез систем автоматического управления в приложении Control System Toolbox методом корневых годографов.....</b>	<b>3</b>
Краткие сведения из теории .....	3
Свойства корневых годографов (случай отрицательной обратной связи) .....	6
Использование приложения Control System Toolbox при проектировании систем автоматического управления методом корневых годографов.....	7
Функции вызова графического интерфейса .....	8
Указания к выполнению лабораторной работы .....	12

### Лабораторная работа № 2

<b>Моделирование линейных стационарных систем автоматического управления в MATLAB (приложение Control System Toolbox).....</b>	<b>14</b>
Создание моделей стационарных систем.....	14
Преобразование моделей .....	16
«Арифметические» операции с моделями .....	16
Указания к выполнению лабораторной работы .....	20

### Лабораторная работа № 3

<b>Моделирование линейных цифровых систем автоматического управления в MATLAB.....</b>	<b>24</b>
Указания к выполнению лабораторной работы .....	28
Список литературы .....	31