# Лекция 4

# Модальное управление с использованием наблюдателей состояния

# Синтез наблюдающего устройства

При решении практических задач управления методами теории пространства состояний мы часто встречаемся со случаями, когда часть переменных вектора состояния оказываются неизмеримыми. Если имеется математическая модель системы, то можно попытаться восстановить состояние системы по измеряемым сигналам входа и выхода.

Восстановление вектора состояния (t) называется его оценкой, а устройство, обеспечивающее получение оценки по измерениям управления **u(t)** и k-мерного (обычно k<n) вектора выхода **y(t)** на конечном интервале времени, ̶ наблюдателем.

Пусть стационарный объект описывается традиционной системой уравнений

(4.1)

Предположим, что матрицы *A, B, C* известны, тогда вектор **x(t)** можно аппроксимировать состоянием **(t)** модели

= (4.2)

которая имеет тот же вход, что и объект (4.1), то есть **u(t)**.

Если модель (4.2) является **идеальной аппроксимацией системы** (4.1), в том смысле, что их параметры и начальные условия идентичны, то состояния **x** и  **также совпадают**. Если начальные условия для систем (4.1) и (4.2) различны, то сходится к **x** только тогда, когда система (4.1) асимптотически устойчива.

При восстановлении (4.2) не используется измеряемый выход.

Качество восстановления улучшается, если ввести в модель разность измеренного выхода и его оценки в виде обратной связи:

= (4.3)

Здесь *L* - некоторая матрица коэффициентов размерностью n×k, обеспечивающая требуемый вид переходных процессов оценки вектора состояния. Такой наблюдатель был предложен Американским учёным Д. Г. Люенбергером и, соответственно, получил его имя (наблюдатель Люенбергера).

Введем ошибку восстановления

. (4.4)

Вычитая (4.3) из дифференциального уравнения (4.1), получим

=

Очевидно, для того чтобы ошибка восстановления стремилась к нулю, необходимо выбрать матрицу *L* так, чтобы система (4.5) была асимптотически устойчива.

Мы видим, что модель (4.5) восстанавливает все составляющие вектора состояния, поэтому она называется наблюдателем состояния полного порядка.

Перед тем, как проектировать наблюдатель, необходимо проверить наблюдаемость системы. Это можно сделать с помощью простого фрагмента программы, аналогичного рассмотренному в лекции 3, для определения управляемости системы:

**A = […];**

**C = […];**

**No = (obsv(A,C)); %Построение матрицы наблюдаемости**

**unno = length (A) - rank (No) ; % Число ненаблюдаемых мод**

**if unno == 0**

**disp ( 'Система полностью наблюдаема' )**

**else**

**M = 'Число ненаблюдаемых мод равняется ' ;**

**disp ([M unno])**

**end**

Таким образом, система является полностью наблюдаемой, если ранг матрицы наблюдаемости равен порядку системы.

Матрица наблюдаемости может быть построена с помощью функции obsv, которая может вызываться в одном из вариантов:

**>> No = obsv(A, С);**

**>> No = obsv(sys);**

**>> No = obsv(sys.A, sys.С);**

Описанные выше функции **acker** и **place**, могут быть применены и для расчета коэффициентов обратных связей наблюдателя одномерной системы.

Для этого надо транспонировать матрицу *A* и заменить *B* на *С’*:

**>>Lt=acker(A',С',p),**

**>>L=Lt'**

где **p** – вектор **желаемых полюсов наблюдателя**.

Например:

**>> A=[0 1; 2 3];**

**>> C=[1 0];**

**>> p=[ -5 -5];**

**>> Lt=acker(A', C', p);**

**>> L=Lt'**

**L=**

**13**

**66**

Общая схема системы управления с наблюдателем состояния полного порядка будет иметь вид, представленный на рис 4.1.

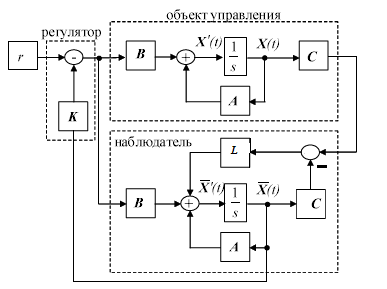


Рис. 4.1. Схема системы управления с наблюдателем полного порядка

При синтезе наблюдателей обычно используют принцип разделимости.

**Принцип разделимости**. Допустим, что объект описывается уравнениями (4.1), регулятор состояния (см. рис. 4.1)

(4.6)

а наблюдатель полного порядка - уравнением (4.3).

Тогда уравнения состояния всей системы имеют вид:

. (4.7)

Если ввести ошибку наблюдения (4.4), то с помощью линейного преобразования

(4.8)

получим

(4.9)

Из (4.9) согласно правилам вычисления определителей блочных матриц (определитель блочной матрицы равен произведению определителей ее диагональных блоков) следует, что характеристическое уравнение всей системы имеет вид,

то есть равно произведению характеристических полиномов идеализированной системы с полными измерениями и собственно наблюдателя. Из этого следует, что характеристические полиномы замкнутой системы управления и наблюдателя независимы и, следовательно, мы можем назначать полюсы наблюдателю независимо от полюсов системы управления.

Процесс, описываемый уравнениями (4.7), как и (4.9), состоит из двух взаимно независимых процессов: из процесса оценки вектора и из собственно процесса управления.

Эту возможность разделения процессов в наблюдателе и регуляторе принято называть ***принципом разделимости***.

С учетом принципа разделимости синтез наблюдателя состояния выполняется ***независимо*** от синтеза регулятора состояния.

В этой связи, очевидно, что для выполнения операций синтеза наблюдателей состояния полного порядка, могут быть использованы те же алгоритмы, которые используются для синтеза регуляторов состояния.

Таким образом, задачу построения матрицы коэффициентов усиления наблюдателя L (размером nxm), можно решить выполнением команд

**l = place(A',C', p); L=l',**

где **p** – вектор желаемых собственных значений или полюсов наблюдателя.

Для одномерных систем, как было показано выше, можно использовать функцию **acker(A',С', p)**.

Пример 1. Построить наблюдатель полного порядка для объекта управления второ­го порядка.

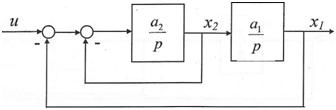


Рис. 4.2

Пусть уравнения состояния, составленные для объекта на рис. 4.2, имеют матрицы *A*, *B* и *C*:

*A*=; *B*=; *C*=.

Составим структурную схему объекта в программе Simulink, положив a1=3, a2=2 (см. рис. 4.3 и 4.4).

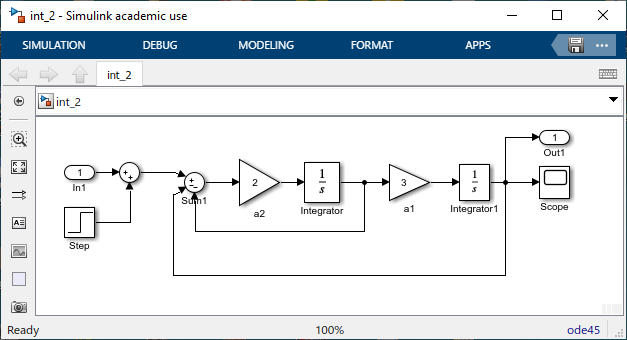


Рис. 4.3. Модель объекта (файл int\_2.slx)

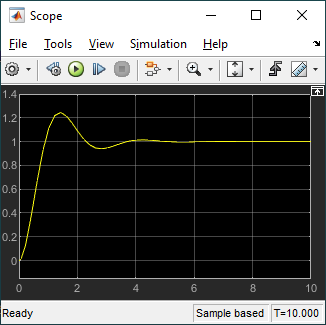


Рис. 4.4. Переходная характеристика ОУ, показанного на рис 4.3

Выполним сначала процедуру настройки модального регулятора для объекта 2 порядка с желаемым временем регулирования 1с по распределению Баттерворта, получим:

**>>[A,B,C,D]=linmod('int\_2')**

A = 0 3

-2 -2

B = 0

2

C = 1 0

%Вводим порядок системы и желаемое время ПП для сценария

**>>batterwort** % (файл batterwort.m), n=2, tgel=1, получаем:

p =

-2.0648 + 2.0648i

-2.0648 - 2.0648i

**>>K = place(A,B,p)**

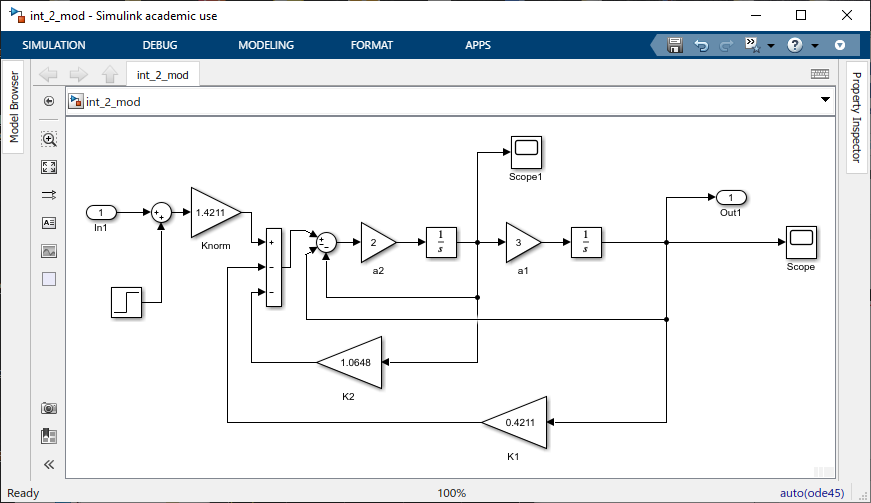
K = 0.4211 1.0648

Нормирующий коэффициент knorm для сообщения замкнутой системе единичного коэффициента усиления может быть вычислен также по выражению:

**>> Knorm=1/(-C/(A-B\*K)\*B)**

Knorm = 1.4211

Составим структурную схему полученной системы в Simulink (файл int\_2\_mod.slx), показанную на рис. 4.5.



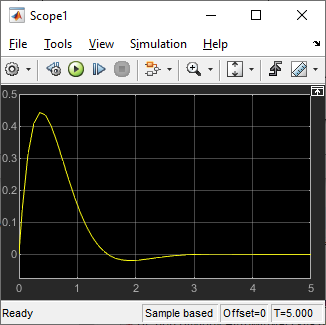
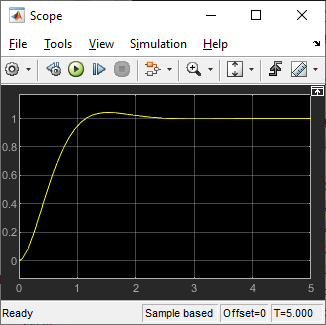
 

Рис. 4.5. Структурная схема системы с модальным регулятором

и результаты моделирования

Произведем расчет наблюдателя (коэффициентов матрицы *L*) с учетом того факта, что динамика замкнутого наблюдателя (собственные значения матрицы *A-LC*) должна быть примерно в 3-10 раз быстрее (обычно в 3 раза), чем динамика системы с модальным управлением.

Так как объект имеет второй порядок, а в качестве выхода используется переменная x1, то матрица *L* будет иметь вид *L*=.

Выберем желаемое время переходного процесса наблюдателя, равное 0,1 с.

**>> batterwort** %(файл batterwort.m), n=2, tgel=0.1 с, получаем:

tgel = 0.1000

Вектор желаемых полюсов

p = -20.6475 +20.6475i

-20.6475 -20.6475i

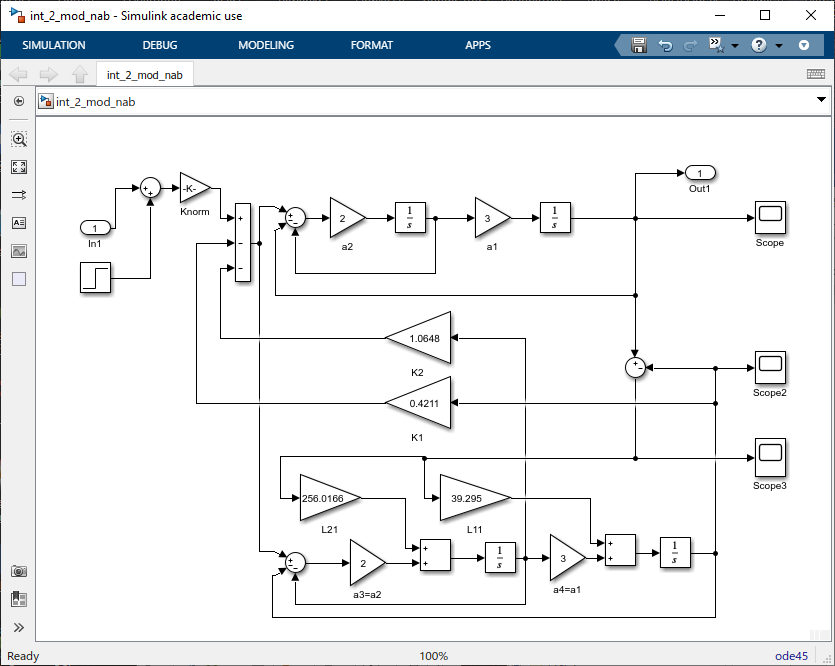
**>> l=place(A',C',p);**

**>>L = l'**

L = 39.2950

256.0166

Составим структурную схему системы, замкнутую с помощью рассчитанного ранее модального регулятора и наблюдателя, показанную на рис. 4.6 (файл int2\_mod\_nab.slx).



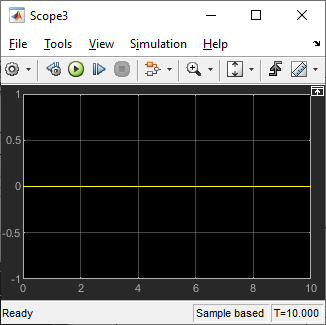
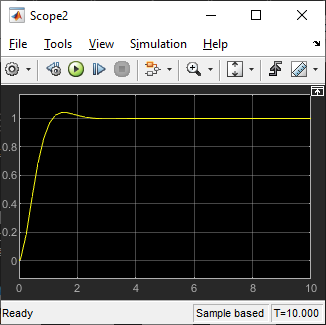
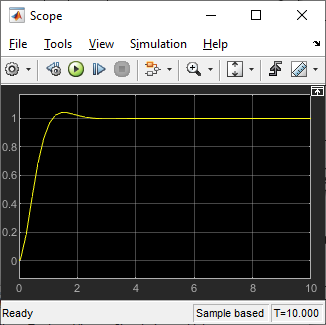


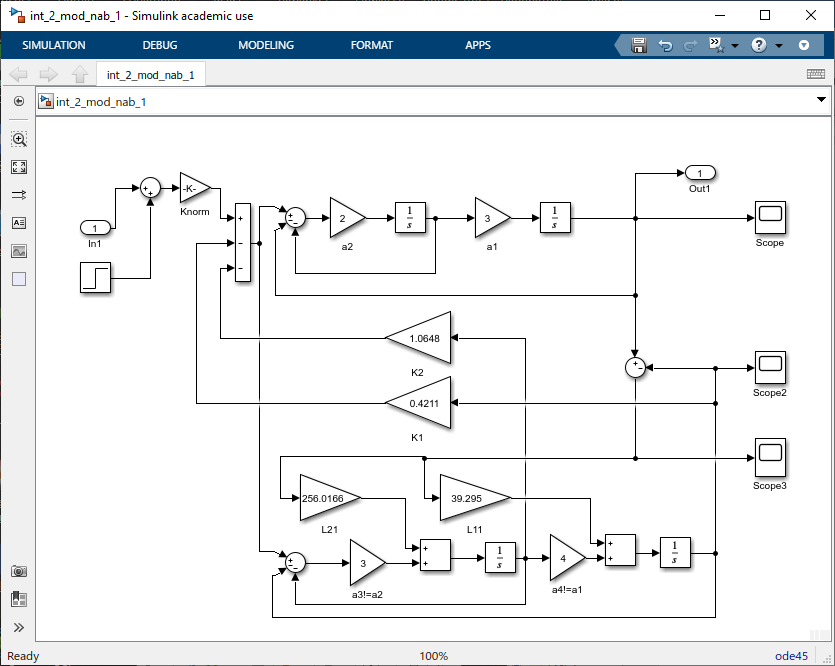
Рис. 4.6. Система с наблюдателем и модальным регулятором

Как видно из результатов моделирования (см. рис. 4.6), ошибка наблюдения, в том числе и на начальном интервале, равна нулю! Почему?

Это происходит потому, что обе подсистемы: объект управления и наблюдатель имеют одинаковые параметры, следовательно, и процессы на их выходах будут одинаковыми. В такой ситуации наблюдатель не работает.

Может ли такая ситуация иметь место на **практике**? Конечно же нет!

Изменим параметры наблюдателя: пусть a2=3, а a1=4 (это довольно приличное расхождение, см. файл int\_2\_mod\_nab\_1.slx) и выполним симуляцию (см. рис. 4.7).



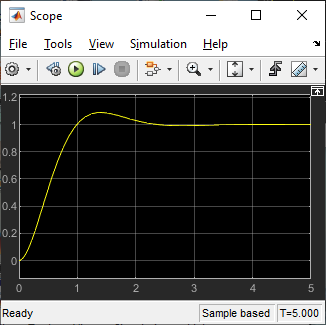
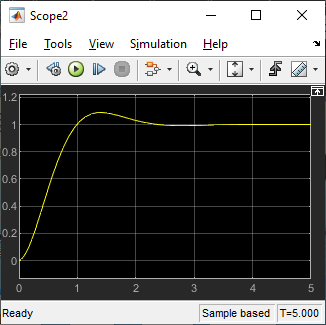
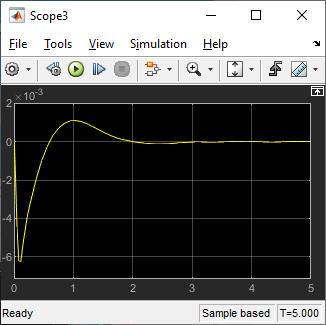
  

Рис. 4.7. Система с модифицированным наблюдателем и модальным регулятором

Ошибка наблюдения появилась. Ее максимальное значение -6.3·10-3.

Другим способом, с помощью которого можно показать работу наблюдателя при **совпадающих** параметрах объекта и наблюдателя, является задание ненулевых начальных условий объекта управления (или наблюдателя). Это продемонстрировано в Simulink-модели **int\_2\_mod\_nab\_3**. Здесь заданы ненулевые начальные условия у обоих интеграторов (0.05 и 0.1). По результатам моделирования видно, что начальная ошибка компенируется практически полностью через 0.2 с. Ознакомьтесь с работой этой модели самостоятельно и сделайте выводы.

Для решения задач проектирования наблюдателей в среде MATLAB могут использоваться функции пакета Control System Toolbox, перечисленные ниже.

Для формирования ***динамического*** регулятора **rsys** для заданной в пространстве состояний модели sys в системе MATLAB можно использовать функцию **reg**. Аргументами этой функции являются матрицы коэффициентов обратных связей по переменным состояния *K* и матрицы *L* коэффициентов обратных связей наблюдателя полного порядка (см. рис. 4.8).

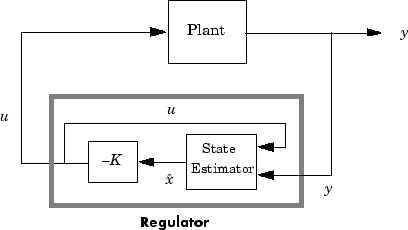


Рис. 4.8. Объект управления с наблюдателем полного порядка и модальным регулятором

Обращение к этой функции выглядит следующим образом:

>>sys = ss(A,B,C,D);

>>rsys = reg(sys,K,L);

***Пример 2. Построение наблюдателя с помощью функции reg().***

% В качестве файла модели объекта управления используем **int\_2**

**>> K= [0.4211 1.0648];**

**>> L=[39.2950; 256.0166];**

**>> [A,B,C,D]=linmod('int\_2');**

**>> sys=ss(A,B,C,D);**

**>> rsys=reg(sys,K,L)**

**rsys =**

**A =**

x1\_e x2\_e

x1\_e -39.3 3

x2\_e -258.9 -4.13

**B =**

y1

x1\_e 39.3

x2\_e 256

**C =**

x1\_e x2\_e

u1 -0.4211 -1.065

**D =**

y1

u1 0

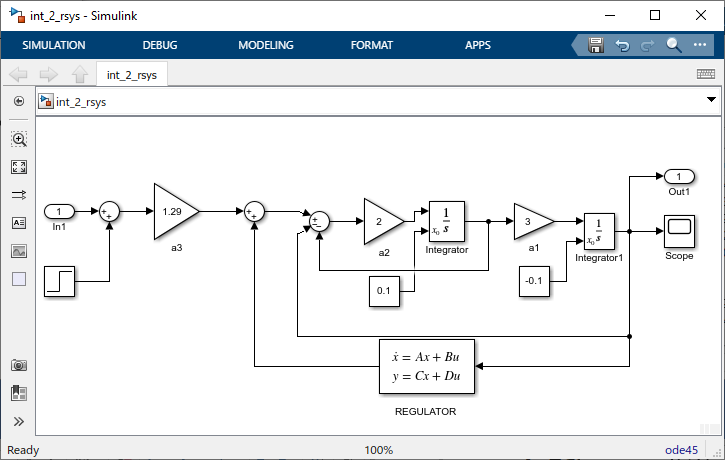


Рис. 4.9. Модель объекта управления с комбинированным наблюдателем полного порядка и модальным регулятором

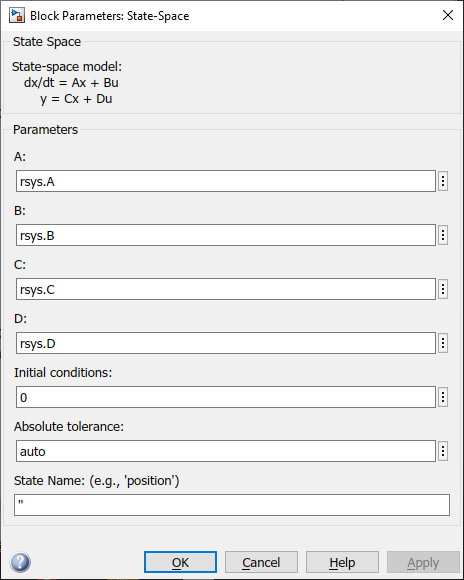
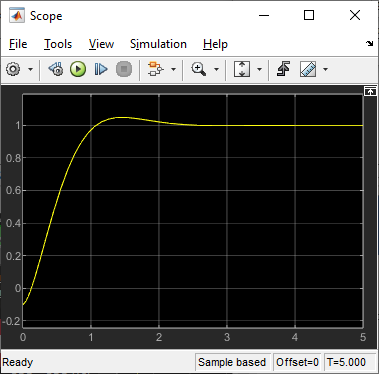
 

Рис. 4.10. Настройка блока REGULATOR и результат моделирования

Как видно из рис. 4.9 комбинированный наблюдатель-регулятор (блок REGULATOR) соединяется с объектом с помощью **положительной** обратной связи.

Ознакомьтесь с работой файлов Simulink-моделей **int\_2\_rsys** и **int\_2\_rsys1**. Их отличие заключается в том, что у модели ОУ в **int\_2\_rsys1** начальные условия ненулевые.

В том случае, когда часть выходов системы доступны непосредственному измерению, можно использовать функцию reg или dreg в следующей форме:

rsys = reg(sys,K,L,sensors,known,controls);

где индексы векторов sensors и known определяют, какие выходы являются измеряемыми и какие входы известными (см. рис. 4.11). В этом случае наблюдатель будет иметь пониженный порядок. Именно такой случай встречается в **практических задачах** проектирования.

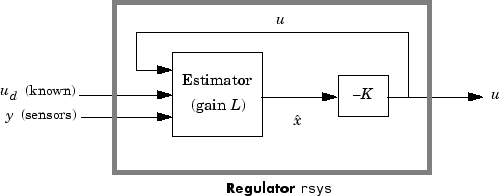


Рис. 4.11. Модальный регулятор с наблюдателем пониженного порядка

Для формирования наблюдающего устройства предназначена также функция **estim** (форма обращения для непрерывных и для дискретных систем одинакова):

est = estim (sys,L);

est = estim(sys,L,sensor,known).

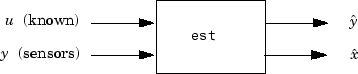


Рис. 4.12. Наблюдатель состояния, реализованный с помощью функции **estim**

В Simulink-моделях обычно блоки регуляторов и наблюдателей (reg и estim) оформляются в виде LTI-блоков с указаниями всех номеров входов и выходов, чтобы избежать путаницы. При необходимости на входах и выходах используются мультиплексоры и демультиплексоры.

Пример. Рассмотрим систему модели пространства состояний с семью выходами и четырьмя входами. Предположим, вы разработали матрицу усиления наблюдателя L, используя выходы 4, 7 и 1 установки в качестве измерений датчиков, а входы 1, 4 и 3 установки в качестве известных (детерминированных) входов. Затем вы можете сформировать оценку, т.е. построить наблюдатель состояния для остальных переменных с помощью следующих команд:

sensors = [4,7,1];

known = [1,4,3];

est = estim(sys,L,sensors,known)

Как отмечалось выше, для повышения точности оценивания полюса наблюдателя должны гарантировать более быстрые процессы в наблюдателе, чем в объекте управления. Поэтому размещение полюсов наблюдателя на комплексной плоскости должно быть левее соответствующих полюсов объекта. При этом значение среднегеометрического корня наблюдателя выбирается из диапазона ω0наб =(3÷10)⋅ω0сист (обычно 3÷5).

Следует учитывать, что с одной стороны качество наблюдения с увеличением коэффициентов матрицы *L* (т.е. увеличение быстродействия наблюдателя) улучшает процесс наблюдения, а с другой, ведет к ухудшению помехоустойчивости.

Все рассмотренные выше функции имеют дискретные аналоги. С особенностями использования этих функций вы ознакомитесь в процессе выполнения лабораторной работы 3, посвященной проектированию дискретных наблюдателей для исходных непрерывных объектов управления.

При затруднениях обращайтесь к справочной системе MATLAB и ресурсам Internet.

**Задание к практической работе 4**

1. Изучить представленный материал и методические примеры лекции 4.
2. Построить наблюдатель полного порядка для следящей системы с выбранным типом двигателя постоянного тока (по вариантам).
3. В качестве выхода системы использовать скалярный сигнал с датчика положения привода.
4. При построении наблюдателя ввести невязку по параметрам наблюдателя и ОУ в виде разницы между параметрами и/или начальными условиями. Невязка по параметрам должна быть не более ±5% от их номинальных значений.
5. Выполнить исследование процесса наблюдения.
6. Сделать выводы.
7. Составить краткий отчет по работе.