Dr Bob Davidov

Аналитическое конструирование интегрального линейно-квадратичного регулятора

Цель работы: Изучить принцип построения интегрального линейного квадратичного регулятора в MATLAB.

Задача работы: Средствами MATLAB спроектировать интегральный линейно-квадратичный регулятор.

Приборы и принадлежности: Персональный компьютер, интегрированная среда MATLAB.

Введение

Линейно-квадратичный регулятор обеспечивает оптимальное управление замкнутой системой при минимизации квадратичного функционала качества. В отличии от ПИД регулятора, квадратичный регулятор не вычисляет производные, он использует переменные состояния системы отслеживаемые наблюдателем. В этой работе рассматриваются варианты построения интегрального линейно-квадратичного регулятора в интегрированной среде MATLAB.

ОБЩИЕ СВЕДЕНИЯ

Для определения возможности решения задачи управления в теории управления используются два важных понятия: управляемость и наблюдаемость, которые определяют перед разработкой регулятора для заданного объекта управления.

Управляемость (Controllability)

Управляемость, как возможность перевода системы из любого начального состояния в любое конечное состояние за конечный интервал времени при неограниченных входных воздействиях $\mathrm{u}(t)$. Для линейных систем управляемость эквивалентна условию равенства ранга матрицы управляемости

$C_o = [B\ AB\ A^2B \ldots A^{n-1}B]$ порядку системы n (размеру матрицы A системы в формате пространства состояний):

$$rank(C_o) = size(A, 1)$$

В MATLAB матрицу управляемости C_o можно найти функцией ctrb(A,B) или ctrb(SYS), где SYS – модель в формате пространства состояний.

Наблюдаемость (Observability)

Наблюдаемость – возможность определения внутренних состояний системы по значениям ее выходов $y(t)$. Подобно управляемости, для линейных систем наблюдаемость эквивалентна условию равенства ранга матрицы наблюдаемости

$O_b = [C;\ CA\ CA^2 \dots\ CA^{n-1}]$ порядку системы n:

$$rank(O_b) = size(A, 1)$$

В этом случае, вектор состояния может быть вычислен с помощью измеренных векторов входа и выхода и матрицы системы.

В MATLAB матрицу наблюдаемости C_o можно найти функцией obsv(A,B) или obsv(SYS), где SYS – модель в формате пространства состояний.

Асимптотическая устойчивость (Asymptotic stability)

Устойчивость объекта можно исследовать с помощью собственных значений матрицы A. В динамическом режиме наблюдается стабильное поведение системы если действительные части всех корней системы отрицательны; кроме того, отсутствуют периодические колебания, если у корней нет мнимых значений.

Цели управления

Основными целями управления, могут быть:

• соответствие реакции системы входным воздействиям;

• подавление помех, которые присутствуют в реальных системах;

• ограничение требуемых параметров системы;

• обеспечение управления при неопределенности отдельных параметров;

• обеспечение устойчивости замкнутого контура – объект должен оставаться асимптотически устойчивым;

• реакция системы не должна включать колебания.

• обеспечение оптимального управления в линейно-квадратичном смысле;

• стабилизация системы на неограниченном временном интервале;

• неограниченный положительный запас усиления, т.е. замкнутый контур остается устойчивым для любого $u = –\alpha K x$, где $\alpha \rightarrow \infty$.

Модель объекта

Модель объекта может быть представлена системой дифференциальных уравнений в следующем виде.

$$\begin{cases} \frac{dx}{dt} = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}$$, (для непрерывных объектов).

Эта же система в формате пространства состояний показана на Рисунок 1.

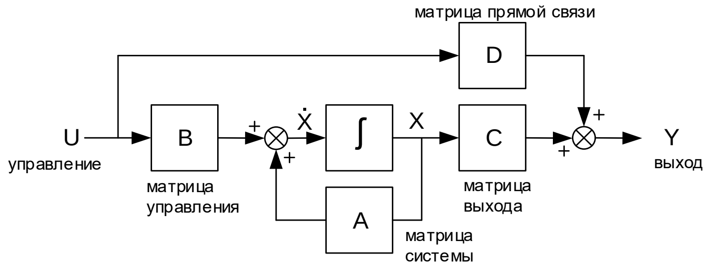


Рисунок 1. Модель системы в формате пространства состояний.

Ниже приведен пример построения дифференциальных уравнений и A,B,C,D матриц пространства состояний системы по ее блок схеме Рисунок 2.

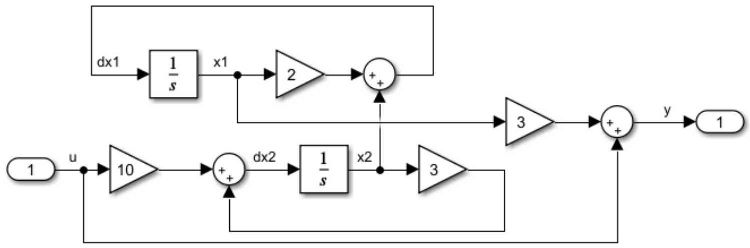


Рисунок 2. Пример динамической системы 2-го порядка.

Система дифференциальных уравнений модели показанной на Рисунок 2:

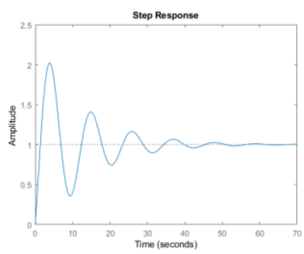
$$\begin{cases} \dfrac{dx_1}{dt} = -0.1667x_1 \ -0.6667x_2 \ +u \\ \dfrac{dx_2}{dt} = \qquad 0.5x_1 \\ \quad y = \quad 0.6667x_1 + 0.6667x_2 \end{cases}$$

Матрицы A, B, C, D системы уравнений:

$$A = \begin{bmatrix} -0.1667 & -0.6667 \\ 0.5 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad C = \begin{bmatrix} 0.6667 & 0.6667 \end{bmatrix}, \qquad D = [0]$$

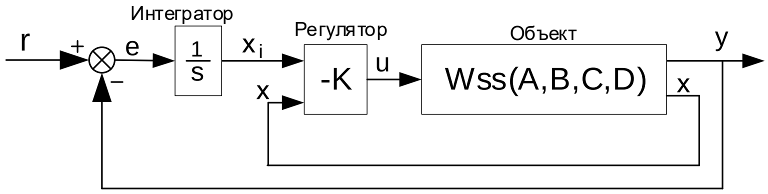
Далее дан пример вычисления передаточной функции по матрицам пространства состояний в MATLAB.

```
clear all
A = [-0.1667 -0.6667;0.5 0];
B = [1;0];
C = [0.6667 0.6667];
D = 0;
Wss = ss(A,B,C,D);

Wtf = tf(Wss)
      Wtf =

          2 s + 1
      -----------------
      3 s^2 + 0.5 s + 1

step(Wtf);
```



Регулятор

Положение интегрального линейно-квадратичного регулятора K = [K_i; K_x] в составе системы управления показано на Рисунок 3 в двух вариантах. На вход регулятора подается вектор интегрированного сигнала рассогласования x_i и вектор x переменных состояния объекта.



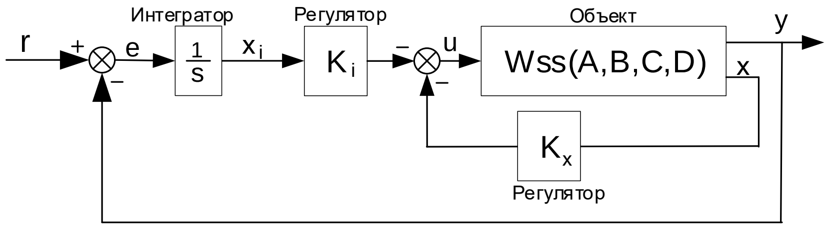


Рисунок 3. Варианты обозначения положения интегрального линейно-квадратичного регулятора K = [K_i; K_x] в составе системы управления. Вариант [3] вверху и доработанный вариант [2] внизу.

Коэффициенты оптимального регулятора можно найти библиотечной функцией MATLAB [3]

[K,S,e] = lqi(SYS,Q,R,N),

в которой аргумент SYS – модель объекта в формате пространства состояний, вычисляется как SYS = ss(A,B,C,D); Q – неотрицательно-определенная весовая матрица “штрафа за отклонение от нуля” (например, единичная матрица); скаляр r или матрица R – положительный весовой коэффициент (например, 1 или 10) или матрица “штрафа за использование управления”, N – весовая матица, если N не используется, ее значение принимается равным нулю.

Скалярное управление $u = -Kz = -K[x; x_i]$ минимизирует следующий функционал (для управляющего воздействия r = 0).

$$J(u) = \int_{t=0}^{\infty}\{z^T Qz{+}u^T Ru + 2z^T Nu\}dt \text{ для непрерывного времени}$$

$$J(u) = \sum_{n=0}^{\infty}\{z^T Qz{+}u^T Ru + 2z^T Nu\} \text{ для дискретного времени}$$

В дискретном времени, lqi вычисляет выход интегратора x_i используя формулу Эйлера

$$x_i[n+1] = x_i[n] + T_s(r[n] - y[n]),$$

где T_s – период дискретного времени модели.

Если матрица N не указывается, ее значение принимается равным нулю. Помимо коэффициентов **K** регулятора, функция [K,S,e] = lqi также находит решение **S** алгебраического уравнения Риккати и корни ***e*** характеристического полинома замкнутого контура.

Матрица коэффициентов регулятора находится из соотношения [1, стр. 148]: $K = B^T \hat{K}$, где $\hat{K}$ является решением нелинейного матричного уравнения Риккати (J. Riccati)

$$-\hat{K}A - A^T\hat{K} + \hat{K}BB^T\hat{K}\,/r - Q = 0$$

В MATLAB это уравнение обозначается как

$$A^TS + SA - SBR^{-1}B^TS + Q = 0, \qquad K = R^{-1}B^TS$$

Пример применения стандартной функции MATLAB lqi()

Найдем параметры оптимального линейно-квадратичного интегрального регулятора для объекта с передаточной функцией $W_o = \frac{1}{s^2+0.5s+1}$; $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ и $r = 1$

Решение:

```
Wo_tf = tf(1,[1 0.5 1]) % структура объекта в формате ПФ
      Wo_tf =
               1
        ---------------
        s^2 + 0.5 s + 1
      Continuous-time transfer function

Wo_ss = ss(Wo_tf)  % объект в формате пространства состояний
      Wo_ss =
        A =
                x1    x2
         x1   -0.5    -1
         x2      1     0
        B =
              u1
         x1    1
         x2    0
        C =
              x1   x2
         y1    0    1
        D =
              u1
         y1    0

      Continuous-time state-space model

[K,S,e] = lqi(Wo_ss,Q,r,0) % K - параметры регулятора; S - решение
алгебраического уравнения Риккати; e - корни характеристического полинома
замкнутой системы
      K =
          1.5461    1.4682   -1.0000
      S =
          1.5461    1.4682   -1.0000
          1.4682    3.5502   -2.0461
         -1.0000   -2.0461    2.4682
      e =
        -0.6365 + 0.0000i
        -0.7048 + 1.0365i
        -0.7048 - 1.0365i
```

Два варианта построения структур идентичных систем управления с lqi регулятором в Simulink показаны на Рисунок 4.

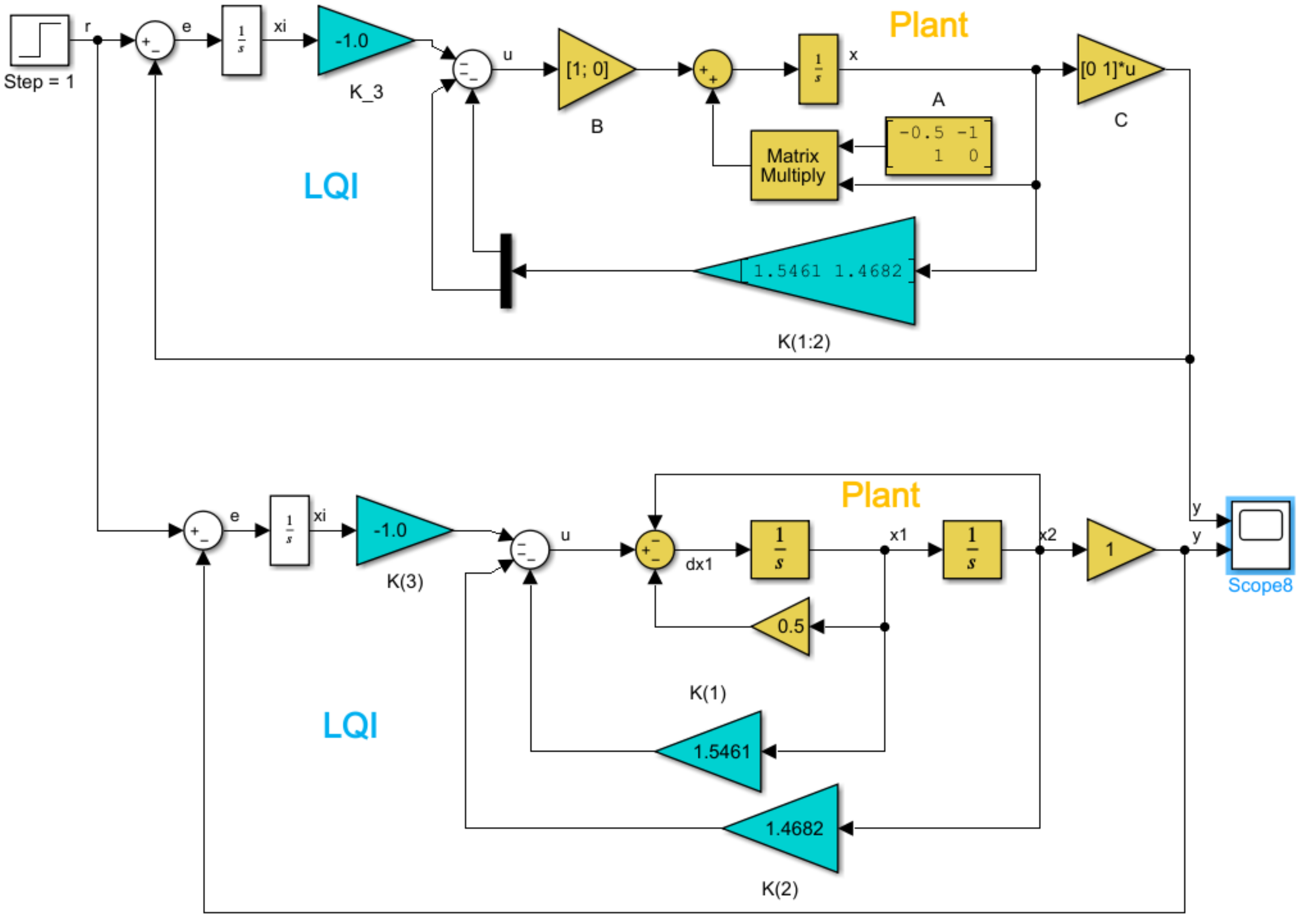


Рисунок 4. Два варианта системы управления с lqi регулятором, ПФ объекта $Wo = \frac{1}{s^2+0.5s+1}$. $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $r = 1$. Блоки объекта выделены желтым, блоки регулятора – бирюзовым цветом.

Аналитическое конструирование интегрального линейно-квадратичного регулятора

Вычислим в MATLAB параметры оптимального интегрального линейно-квадратичного регулятора для того же объекта: $Wo = \frac{1}{s^2+0.5s+1}$, $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $r = 1$ без применения стандартной функции lqi.

1. Перевод передаточной функции в формат пространства состояния

```
Wo_tf = tf(1,[1 0.5 1]) % структура объекта в формате ПФ
      Wo_tf =
               1
        ---------------
        s^2 + 0.5 s + 1
      Continuous-time transfer function

Wo_ss = ss(Wo_tf)  % структура объекта в формате пространства состояний
```

```
Wo_ss =
  A =
          x1    x2
   x1   -0.5    -1
   x2      1     0
  B =
        u1
   x1   1
   x2   0
  C =
        x1  x2
   y1   0   1
  D =
        u1
   y1   0
 Continuous-time state-space model
```

```
A = Wo_ss.A; B = Wo_ss.B; C = Wo_ss.C;
```

2. Проверка объекта на управляемость, наблюдаемость и устойчивость

```
Co = [B A*B] % контролируемость
      Co =
          1.0000   -0.5000
               0    1.0000
rank(Co) % ранг матрицы равен порядку системы ?
      ans =
           2
Ob = [C; C*A] % наблюдаемость
      Ob =

           0     1
           1     0
rank(Ob) % ранг матрицы равен порядку системы?
      ans =
           2

eig(A) % корни объекта
      ans =
        -0.2500 + 0.9682i
        -0.2500 - 0.9682i
```

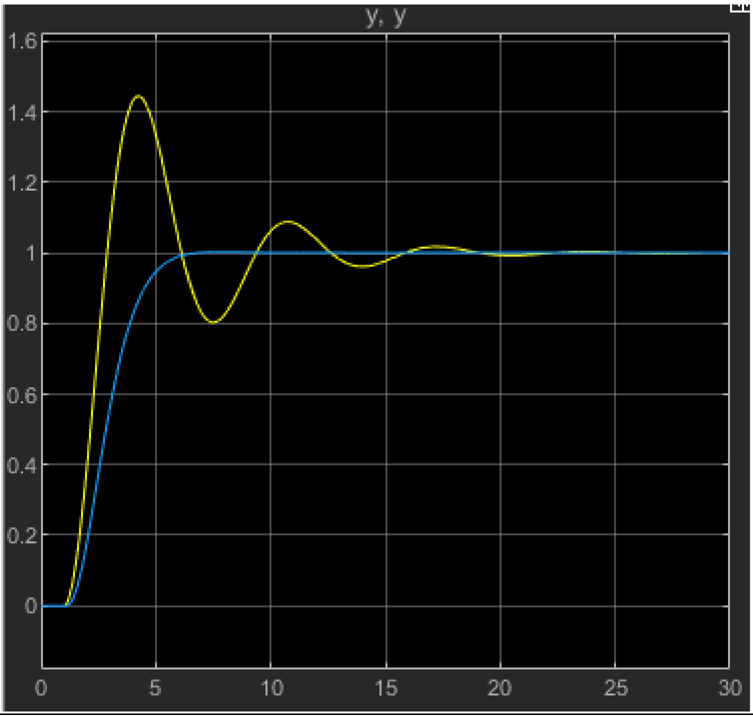
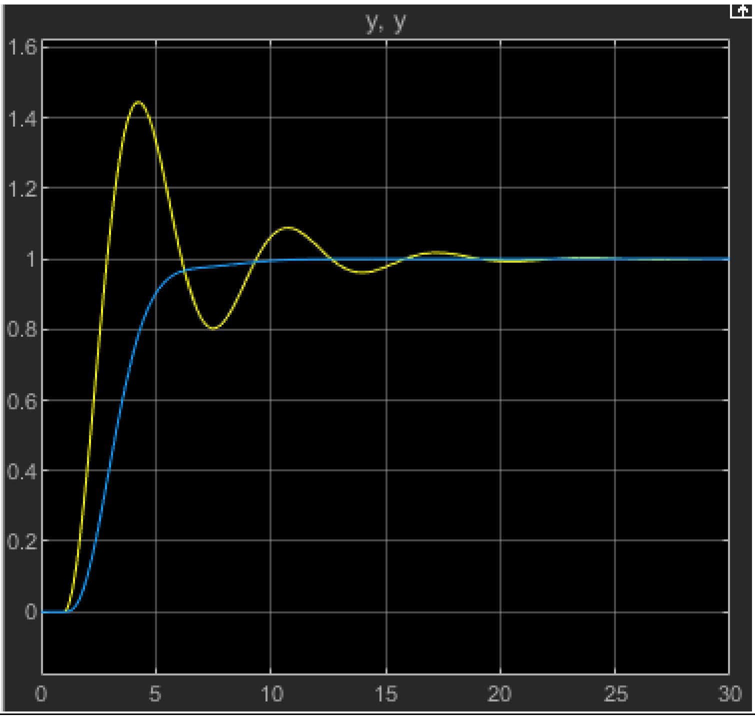
3. Решение уравнения Риккати с модифицированными матрицами [2] $\hat{A} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}$ и $\hat{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}$

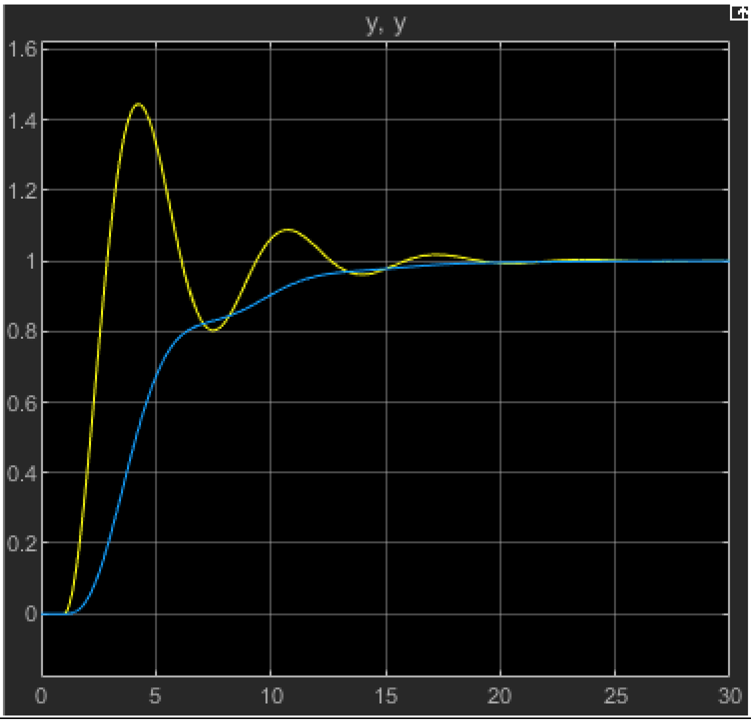
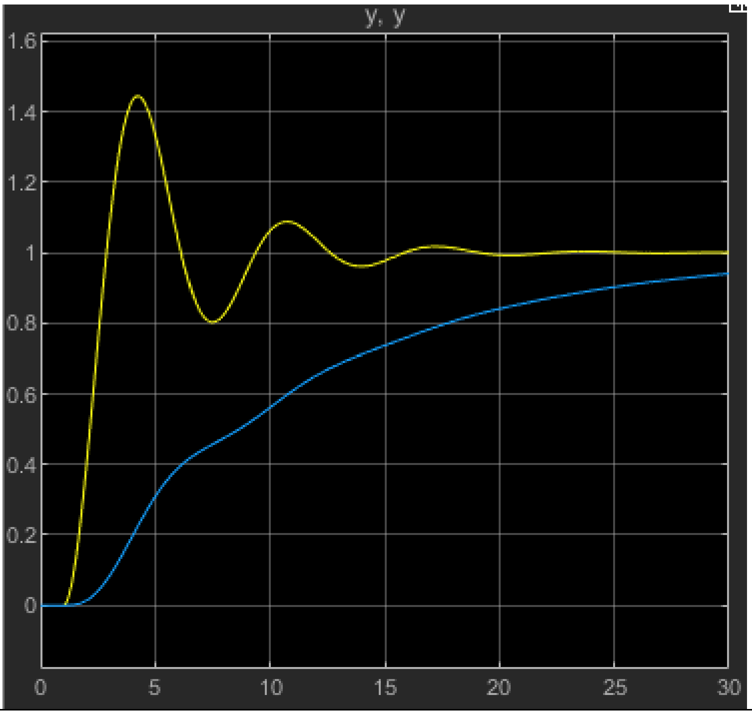
```
Ar = [A zeros(size(A,1),size(C,1)); -C zeros(size(C,1))] % Ar = [A 0; -C 0]
      Ar =
         -0.5000   -1.0000         0
          1.0000         0         0
               0   -1.0000         0
Br = [B; zeros(size(C,1))] % Br = [B; 0]
      Br =
           1
           0
           0
Q = eye(size(Ar,1))
      Q =
           1     0     0
           0     1     0
           0     0     1

r = 1;
```

```
syms sr [1,9]
Sr = [sr1 sr2 sr3; sr4 sr5 sr6; sr7 sr8 sr9]
      Sr =
      [ sr1, sr2, sr3]
      [ sr4, sr5, sr6]
      [ sr7, sr8, sr9]

S = Sr*Ar+Ar'*Sr-Sr*Br*Br'*Sr/r+Q % уравнение Риккати
      S =
      [- sr1^2 - sr1 + sr2 + sr4 + 1,   sr5 - sr2/2 - sr3 - sr1 - sr1*sr2,
      sr6 - sr3/2 - sr1*sr3]
      [ sr5 - sr4/2 - sr1 - sr7 - sr1*sr4, 1 - sr4 - sr6 - sr8 - sr2*sr4 -
      sr2, - sr3 - sr9 - sr3*sr4]
      [sr8 - sr7/2 - sr1*sr7,  - sr7 - sr9 - sr2*sr7,    1 - sr3*sr7]
Kr = solve(S) % решение уравнения Риккати
        struct with fields:
          sr1: [20×1 sym]
          sr2: [20×1 sym]
          sr3: [20×1 sym]
          sr4: [20×1 sym]
          sr5: [20×1 sym]
          sr6: [20×1 sym]
          sr7: [20×1 sym]
          sr8: [20×1 sym]
          sr9: [20×1 sym]
% среди множества решений системы уравнений Kr выбираем вариант i_n, у
которого первый член матрицы имеет максимальное значение ?
i_n = 1;
Kr_sr1_max = vpa(Kr.sr1(i_n))
      Kr_sr1_max =
      - 1.2047746903018979406021177489811 -
      1.0364826372822298091698760334599i
for i = 2:size(Kr.sr1,1) %
    if vpa(Kr.sr1(i)) > Kr_sr1_max
        Kr_sr1_max = vpa(Kr.sr1(i));
        i_n = i;
    end
end
      i_n =
           20
% Матрица S из [K,S,e] = lqi(Wo_ss,Q,r,0)
Kr = vpa([Kr.sr1(i_n) Kr.sr2(i_n) Kr.sr3(i_n); ...
          Kr.sr4(i_n) Kr.sr5(i_n) Kr.sr6(i_n); ...
          Kr.sr7(i_n) Kr.sr8(i_n) Kr.sr9(i_n)])
      Kr =
      [1.5460851506864994151830082004954, 1.4682322219298975091252783348851,
      -1.0]
      [1.4682322219298975091252783348851, 3.5502132977367076113339530389897,
      -2.0460851506864994151830082004954]
      [-1.0, -2.0460851506864994151830082004954,
      2.4682322219298975091252783348851]

Параметры регулятора
K = Br'*Kr/r
      K =
      [ 1.5460851506864994151830082004954, 1.4682322219298975091252783348851,
      -1.0]
K = eval(K) % перевод символьных данных в численные
      K =
          1.5461    1.4682   -1.0000
```

Влияние r, R на качество управления

В следующей таблице показана зависимость параметров регулятора **K**, корней **e** замкнутой системы и реакции на ступенчатое воздействие от скалярной величины **r**. Реакция системы управления показана вместе с реакцией объекта $\frac{1}{s^2+0.5s+1}$ без регулятора.

r	K регулятора, корни **e**	Графики реакций
0.1	K = 4.0699 5.3168 -3.1623 e = -0.9755 + 0.5059i -0.9755 - 0.5059i -2.6189 + 0.0000i	
1	K = 1.5461 1.4682 -1.0000 e = -0.6365 + 0.0000i -0.7048 + 1.0365i -0.7048 - 1.0365i	

10	K = 0.4884 0.3134 -0.3162 e = -0.2840 + 0.0000i -0.3522 + 0.9947i -0.3522 - 0.9947i	
100	K = 0.1251 0.0654 -0.1000 e = -0.0987 + 0.0000i -0.2632 + 0.9717i -0.2632 - 0.9717i	

Графики, для сравнения реакции объекта и системы управления, выполнялись на модели:

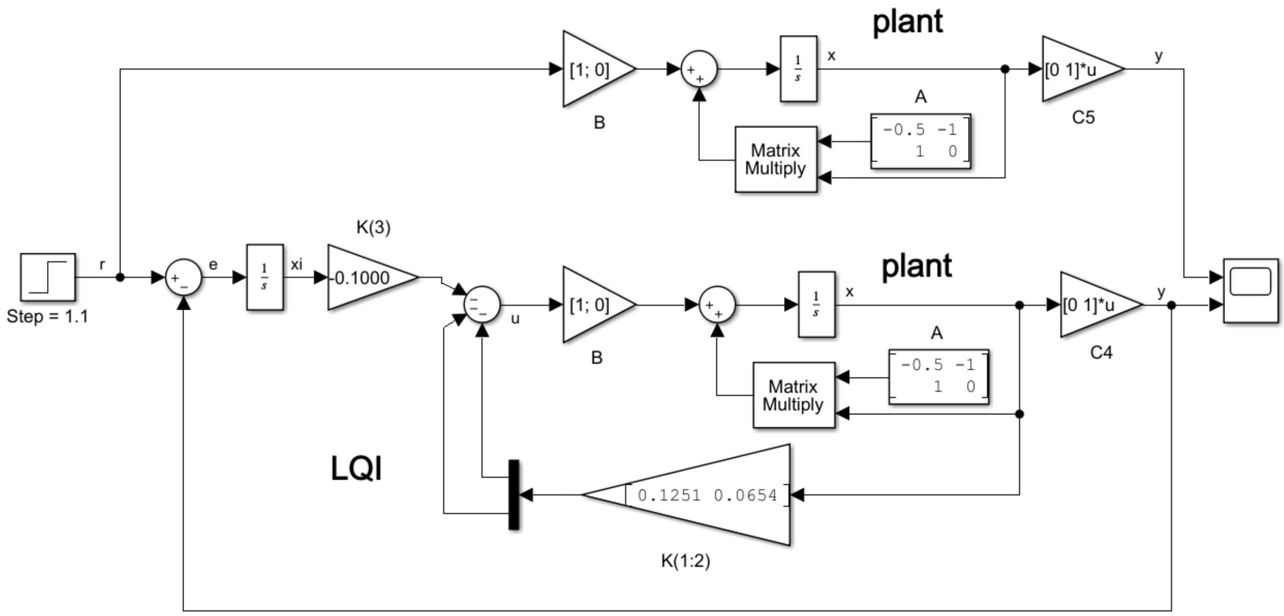


Рисунок 5. Модель объекта (вверху) и системы управления объектом на базе интегрального линейно-квадратичного регулятора lqi.

Как члены матрицы R, так и веса членов диагональной матрицы Q и их распределение влияют на качество управления. Это можно увидеть, например, на Рисунок 6, где показаны реакции следующих систем управления с lqi регулятором:

$Wo = \frac{1}{s^2+0.5s+1}$, $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $r = 1$ (синий график) и $Wo = \frac{1}{s^2+0.5s+1}$, $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$*1e2, $r = 1$ (желтый график).

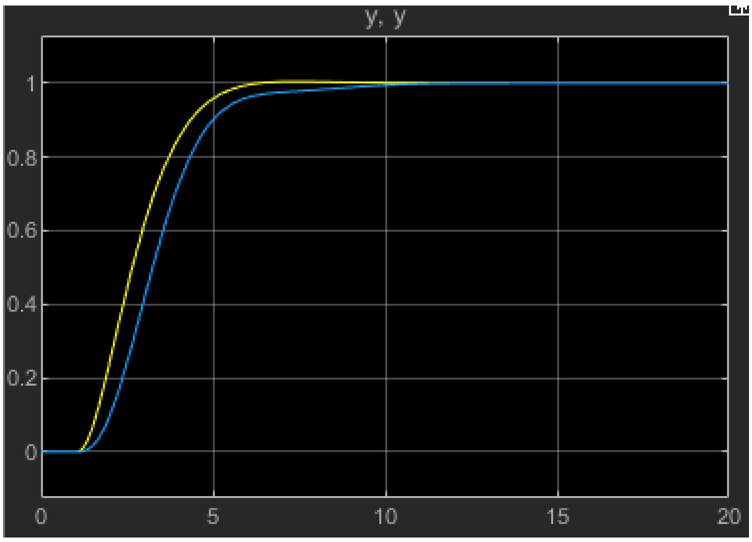


Рисунок 6. Реакции на единичное ступенчатое воздействие систем управления с lqi регулятором: $Wo = \frac{1}{s^2+0.5s+1}$, $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $r = 1$ (синий график) и $Wo = \frac{1}{s^2+0.5s+1}$, $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$***100**, $r = 1$ (желтый график).

Применение наблюдателя в структурах с линейно-квадратичными регуляторами

Если объект наблюдаем полностью, то по измеренным значениям переменной выхода y можно вычислить текущее состояние объекта [1, стр. 148]. При этом управляющее воздействие на объект формируется по оценкам вектора состояния.

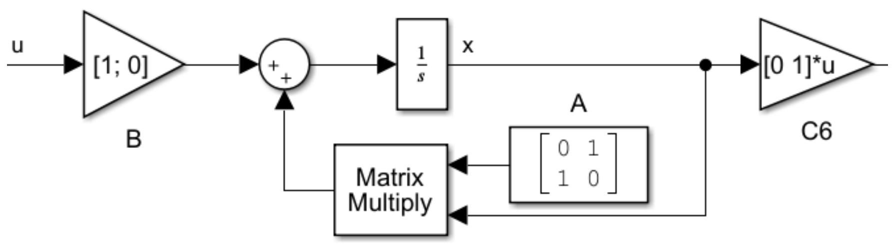
Для оценки вектора состояния в систему включают модель наблюдателя, внутри которого находится модель объекта управления. Поскольку, оценка может отличаться от состояния объекта из-за различия начальных условий, действующих на объект возмущений, а также неточности описания объекта, то наблюдатель дополняют обратной связью при которой оценка должна асимптотически стремиться к состоянию объекта.

Синтез наблюдателя поясняется в соответствующей работе курса [4].

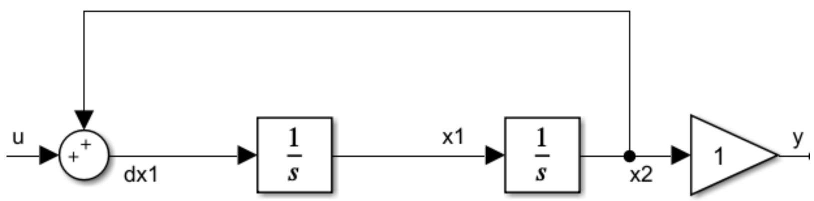
ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Задание 1. Конструирование интегрального линейно-квадратичного регулятора для управления неустойчивым объектом.

1. Постройте в Simulink модель неустойчивого объекта со следующим распределением корней p = [-1 +1].



или

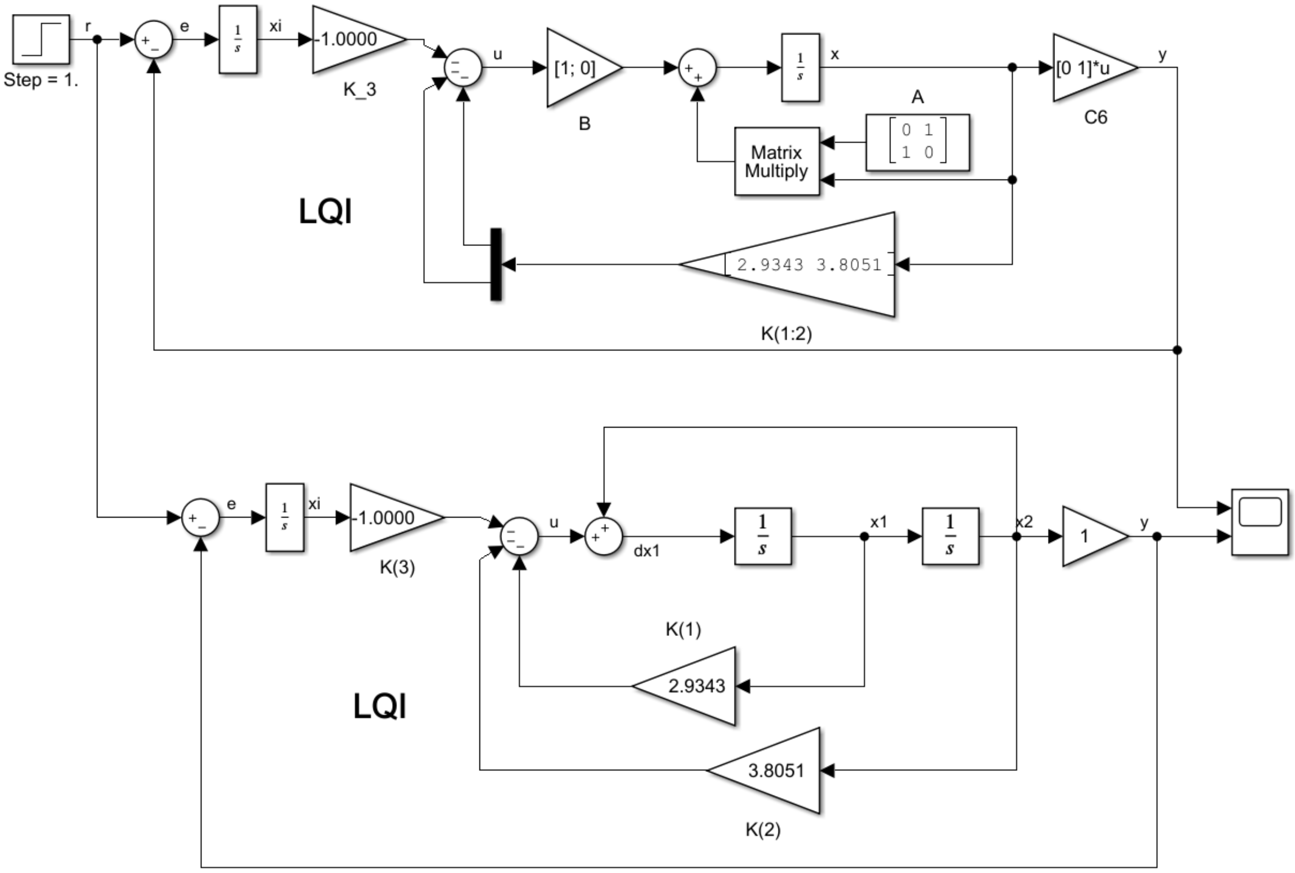


Для выполнения этого пункта можно использовать функции MATLAB:

W_zpk = zpk([],[1 -1],1); ss(W_zpk)

2. Рассчитайте коэффициенты регулятора “**K**” и определите корни замкнутой системы управления “**e**”, как показано в разделе “Пример применения стандартной функции MATLAB lqi()”, [**K**,S,**e**] – результат выполнения функции lqi.
3. Рассчитайте коэффициенты регулятора, как показано в разделе “Аналитическое конструирование интегрального линейно-квадратичного регулятора”.
4. Сравните результаты, полученные в п.2 и п.3.

5. Постройте модель системы управления с квадратичным регулятором (даны два варианта) и график реакции системы на единичное ступенчатое воздействие.



6. Изучите влияние параметров весовых матриц Q и R на качество управления.
7. По блок-схеме п.5 постройте систему дифференциальных уравнений системы в формате пространства состояний.

$$\begin{aligned}
\dot{x}1 &= K(1) * x1 + (1 - K2) * x2 + K(3) * xi + 0 * u \\
\dot{x}2 &= \quad 1 * x1 + \quad\quad 0 * x2 + \quad 0 * xi + 0 * u \\
\dot{x}i &= \quad 0 * x1 + \quad (-1) * x2 + \quad 0 * xi + 1 * u \\
y &= \quad 0 * x1 + \quad\quad 1 * x2 + \quad 0 * xi + 0 * u
\end{aligned}$$

8. Из системы дифференциальных уравнений выделите матрицы A, B, C, D пространства состояний замкнутой системы.

$$A_з = \begin{bmatrix} -K(1) & 1 - K(2) & -K(3) \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} A_o(1{,}1) - K1 & A_o(2{,}1) - K2 & -K(3) \\ A_o(2{,}1) & A_o(2{,}2) & 0 \\ 0 & -1 & 0 \end{bmatrix};$$

$$B_з = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};$$

$$C_з = [0 \quad 1 \quad 0];$$

$D_з = [0]$.

9. По матрице Az = $A_з$ найдите корни характеристического полинома замкнутой системы.

   >> eig(Az)

   -1.5247 + 0.0000i

   -0.7048 + 0.3989i

   -0.7048 - 0.3989i

10. Сравните корни замкнутой системы управления п.9 с результатами выполнения п.2.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие преимущества дает оптимальный линейно-квадратичный интегральный регулятор в сравнении с ПИД регулятором?
2. Влияют ли параметры весовых матриц Q и R на коэффициенты регулятора и, как следствие, на качество управления?
3. При помощи каких средств можно получить переменные состояния, поступающие на вход регулятора?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. А.А.Алексеев, Д.Х.Имаев, Н.Н.Кузьмин, В.Б.Яковлев. Теория управления: Учеб./СПб.: Изд-во СПбГЭТУ “ЛЭТИ”, 1999. – 435 с.
2. István Kisszölgyémi, Károly Beneda and Zsolt Faltin. Linear Quadratic Integral (LQI) Control for a Small Scale Turbojet Engine with Variable Exhaust Nozzle. 2017 International Conference on Military Technologies (ICMT) May 31 – June 2, 2017, Brno, Czech Republic.
3. Help MATLAB.
4. Dr. Bob Davidov. Компьютерные технологии управления в технических системах http://portalnp.ru/author/bobdavidov