Dr Bob Davidov

Реализация моделей динамических систем средствами контроллера. Импульсные передаточные функции.

Цель работы: рассмотреть варианты и средства реализации передаточных функций для дискретных систем реального времени.

Задача работы: средствами MATLAB обеспечить перевод описания динамических моделей на языки программирования контроллеров.

Приборы и принадлежности: персональный компьютер, интегрированная среда MATLAB.

Введение

За основу задания динамических свойств систем может быть принята любая из форм представления операторов: дифференциальные уравнения (ДУ), передаточные функции (ПФ), временные характеристики (ВХ) или частотные характеристики (ЧХ), однако для конкретных задач целесообразно выбирать наиболее рациональную форму [1].

Возможные преобразования форм представления моделей вход-выход показаны на Рисунок 1. Сплошные линии орграфа показывают однозначные преобразования, штриховые - неоднозначные преобразования экспериментальных данных. Результаты последних преобразований зависят от выбора структуры оператора и алгоритма обработки данных.

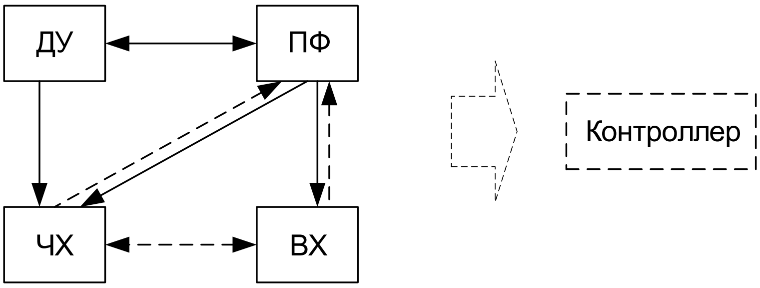


Рисунок 1. Орграф взаимосвязи форм представления моделей.

Модели могут описывать поведение регуляторов систем управления [4], наблюдателей [3], корректирующих звеньев, фильтров [5], и др. динамических систем.

В этой работе рассматриваются MATLAB преобразования форм представления моделей и реализация моделей средствами контроллеров.

ОБЩИЕ СВЕДЕНИЯ

Дискретная модель объекта

Для сравнения результатов преобразования моделей будем использовать одну и ту же систему - объект третьего порядка, дискретная модель которого в форме пространства состояний имеет следующие коэффициенты.

$$A_d = \begin{bmatrix} 1.1 & -0.5 & 0.12 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}; \qquad B_d = \begin{bmatrix} -0.4 \\ 0.6 \\ 0.5 \end{bmatrix}; \qquad (1)$$

$$C_d = [1 \quad 0 \quad 0]; \qquad D_d = [0]$$

Дискретная модель в форме пространства состояний представляется разностными уравнениями первого порядка

$$x[n+1] = \mathrm{A}_d x[n] + \mathrm{B}_d u[n] \qquad (2)$$

$$y[n] = \mathrm{C}_d x[n] + \mathrm{D}_d u[n]$$

Связь переменных уравнений (2) показана на блок схеме Рисунок 2, где X – переменные состояния; 1/z – задержка на один такт.

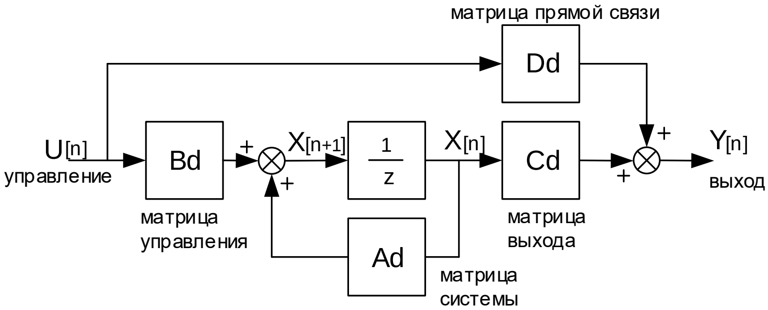


Рисунок 2. Структура дискретной модели объекта в форме пространства состояний.

Система разностных уравнений

Раскроем уравнения (2) подстановкой в уравнения матриц (1) объекта:

$$\begin{aligned} x1[n+1] &= 1.1 \cdot x1[n] + (-0.5) \cdot x2[n] + 0.12 \cdot x3[n] + (-0.4) \cdot u[n] \\ x2[n+1] &= 1 \cdot x1[n] + 0 \cdot x2[n] + 0 \cdot x3[n] + 0.6 \cdot u[n] \\ x3[n+1] &= 0 \cdot x1[n] + 1 \cdot x2[n] + 0 \cdot x3[n] + 0.5 \cdot u[n] \\ y[n] &= 1 \cdot x1[n] + 0 \cdot x2[n] + 0 \cdot x3[n] + 0 \cdot u[n] \end{aligned} \qquad (3)$$

По разностным уравнениям (3) построим Simulink модель дискретного объекта с периодом дискретизации 1 с (см. Рисунок 3).

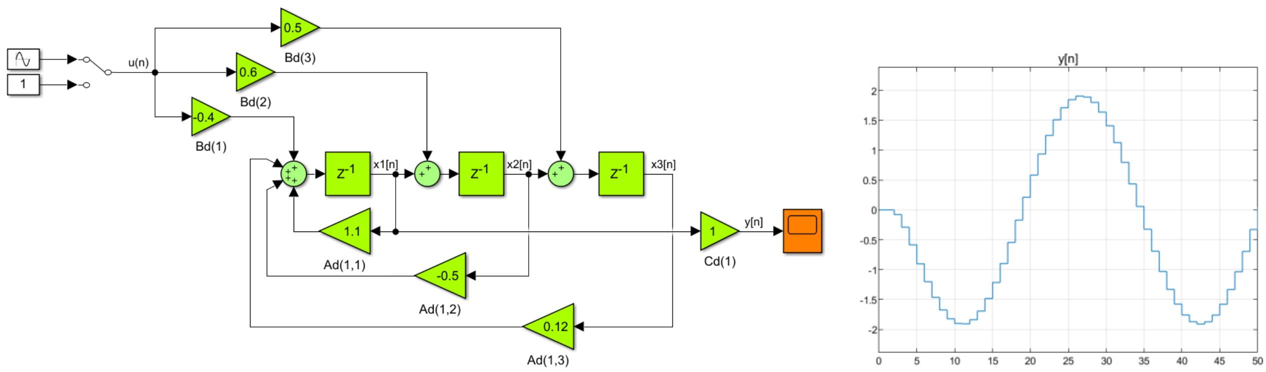


Рисунок 3. Дискретная модель объекта (1) и реакция $\boldsymbol{y}[\boldsymbol{n}]$ объекта на синусоидальное $\boldsymbol{u}[\boldsymbol{n}]$ воздействие.

Импульсная передаточная функция

Воспользуемся преобразованием форм MATLAB для перевода дискретной модели из пространства состояний в формат импульсной передаточной функции, затем в формат непрерывной ПФ и, наконец, в формат пространства состояний непрерывной модели. Ниже показан код m-программы с промежуточными результатами, сдвинутыми вправо на 1 Tab.

```
Ad = [1.1   -0.5   0.12
       1.0    0.0    0.0
       0.0    1.0    0.0];

Bd = [-0.4
       0.6
       0.5];

Cd = [1 0 0];

Dd = 0;

Wo_ss_d = ss(Ad,Bd,Cd,Dd,1); % структура дискретной модели с периодом 1 с
Wo_tf_d = tf(Wo_ss_d)  % дискретная ПФ
```

$$\frac{-0.4z^2 - 0.24z + 0.072}{z^3 - 1.1z^2 + 0.5z - 0.12} \tag{3a}$$

Непрерывная передаточная функция

```
Wo_tf_c = d2c(Wo_tf_d) % непрерывная ПФ
```

$$\frac{0.01803s^2 - 1.079s - 1.661}{s^3 + 2.12s^2 + 2.425s + 0.819}$$

Непрерывная модель в форме пространства состояний

```
Wo_ss_c = ss(Wo_tf_c) % непрерывная модель в форме пространства состояний
      Wo_ss_c =
        A =
                 x1       x2       x3                                    (4)
          x1   -2.12   -1.213   -0.819
          x2       2        0        0
          x3       0      0.5        0
        B =
               u1
          x1   2
          x2   0
          x3   0
        C =
                   x1         x2         x3
          y1  0.009016   -0.2697   -0.8307
        D =
               u1
          y1   0

      Continuous-time state-space model.
```

ВНИМАНИЕ. Изменение шага дискретизации приводит к соответствующему изменению динамики модели. Для того, чтобы эквивалентная непрерывная модель “не отставала” и “не убегала” от дискретной модели с новым шагом дискретизации, необходимо пересчитать коэффициенты непрерывной модели.

Приведенный многоступенчатый перевод дискретной модели (1, 2, 3) в непрерывную модель пространства состояний (4) можно выполнить одной командой d2c: `Wo_ss_c = d2c(Wo_ss_d)`. В этом варианте получится набор матриц с другими коэффициентами относительно многоступенчатого варианта, но это не является ошибкой, поскольку одна и та же ПФ (с одинаковым составом нулей и полюсов) может быть представлена в форме пространства состояний разными вариантами коэффициентов матриц A, B, C, D. Нули и полюса ПФ можно вычислить командами MATLAB: zeros, poles, eig.

Система дифференциальных уравнений

Матрицы (4) непрерывной модели связывают переменные состояния $x(\mathrm{t})$ с входом $y(\mathrm{t})$ и выходом $\mathrm{u(t)}$ модели следующим образом.

$$\begin{aligned} \dot{x}(\mathrm{t}) &= \mathrm{A}_c x(\mathrm{t}) + B_c \mathrm{u}(\mathrm{t}) \\ y(\mathrm{t}) &= \mathrm{C}_c x(\mathrm{t}) + \mathrm{D}_c u(\mathrm{t}), \end{aligned} \qquad (5)$$

Раскроем систему дифференциальных уравнений (5) подстановкой коэффициентов матриц (4):

$$\begin{cases} \frac{dx_1(t)}{dt} = (-2.12)x_1(t) + (-1.213)x_2(t) + (-0.819)x_3(t) + 2u(t) \\ \frac{dx_1(t)}{dt} = 2x_1(t) + 0 \cdot x_2(t) + 0 \cdot x_3(t) + 0 \cdot u(t) \\ \frac{dx_1(t)}{dt} = 0 \cdot x_1(t) + 0.5x_2(t) + 0 \cdot x_3(t) + 0 \cdot u(t) \\ y(t) = 0.009016x_1(t) + (-0.2697)x_2(t) + (-0.8307)x_3(t) + 0 \cdot u(t) \end{cases} \qquad (6)$$

Как в случае построения дискретной Simulink модели (Рисунок 3) по разностным уравнениям (3), построим Simulink модель непрерывного объекта (см. Рисунок 4) по дифференциальным уравнениям (6).

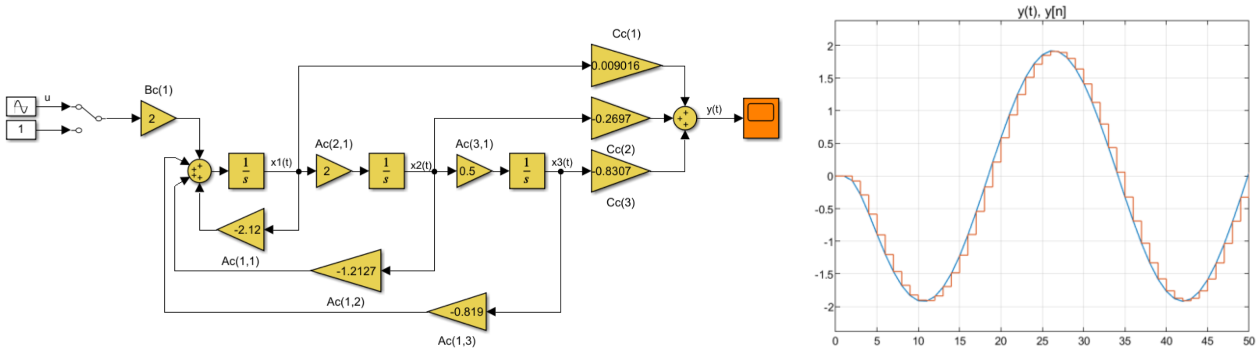


Рисунок 4. Непрерывная модель объекта (1) и реакция $y(t)$ объекта на синусоидальное $u(t)$ воздействие (синий график). Для сравнения, показан коричневый график $y[n]$ дискретного объекта Рисунок 3, график $y(t)$ сдвинут вправо на 0.5 с - половину дискреты.

Перевод модели в частотную область

В MATLAB по данным модели пространства состояний или передаточным функциям можно построить соответствующие амплитудно-фазовые частотные характеристики (АФЧХ). Примеры перевода для рассматриваемого объекта показаны ниже.

```
bode(Wo_ss_d)   % АФЧХ дискретного объекта представленного в форме пространства
состояний
bode(Wo_tf_d)   % АФЧХ дискретного объекта представленного в форме ПФ
bode(Wo_ss_c)   % АФЧХ непрерывного объекта представленного в форме пространства
состояний
bode(Wo_tf_c)   % АФЧХ непрерывного объекта представленного в форме ПФ
grid % нанесение координатной сетки
```

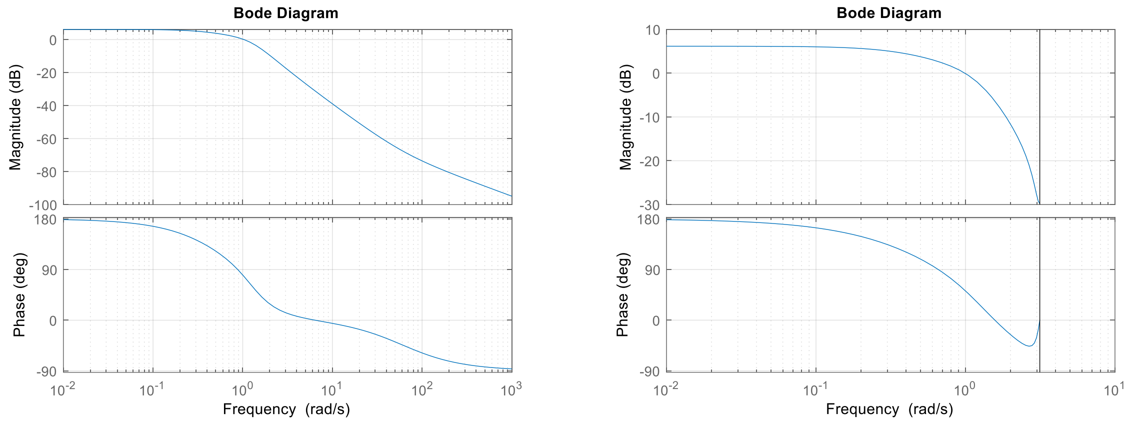


Рисунок 5. Амплитудно-фазовые частотные характеристики непрерывного объекта (слева) и дискретного объекта (справа).

Построение передаточной функции по частотным характеристикам

Имеется множество вариантов обратного перевода модели из частотной области. Вот один из них.

```
[mag,phase,freq] = bode(Wo_tf_c) % выделение амплитуды, фазы и частоты АФЧХ
complex_f = frd(mag.*exp(j*phase.*pi/180),freq) % модель объекта в частотной области
tfest(complex_f,3) % ПФ непрерывного объекта, 3 - порядок объекта
```

$$\frac{0.01803s^2 - 1.079s - 1.661}{s^3 + 2.12s^2 + 2.425s + 0.819}$$

В этом примере зависимости амплитуды `mag` и фазы `phase` от частоты `freq` вычислены оператором bode для ПФ `Wo_tf_c`. Оператор frd сформировал структуру `complex_f` - зависимость комплексных аргументов `mag.*exp(j*phase.*pi/180)` от частоты, которая использовалась оператором tfest для построения ПФ непрерывного объекта третьего порядка. Полученная ПФ точно совпадает с исходной ПФ `Wo_tf_c`.

Как упоминалось во введении, при использовании экспериментальных зависимостей (амплитуды и фазы от частоты) мы бы получили неоднозначный вариант ПФ.

Перевод модели во временную область

MATLAB предлагает множество вариантов вычисления реакции объекта на входное воздействие. Вот один из них – реакция (Рисунок 6) на единичное ступенчатое воздействие вычисляется оператором step:

```
% [y,t] = step(Wo_tf_c) % выделение данных реакции на единичное воздействие
step(Wo_tf_c) % построение графика реакции на единичное воздействие
```

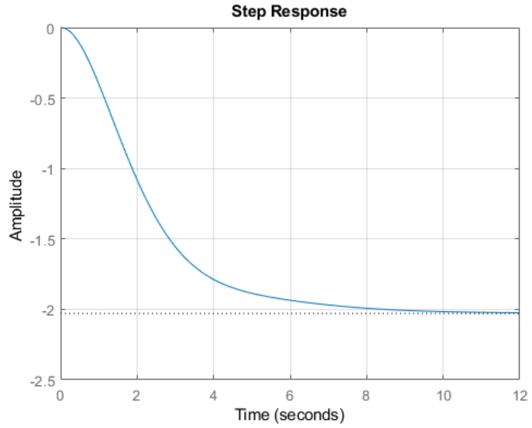


Рисунок 6. Реакция модели непрерывного объекта (ПФ: `Wo_tf_c`) на единичное ступенчатое воздействие.

Построение передаточной функции по временным характеристикам

Обратное преобразование – построение передаточной функции по временным характеристикам входа и выхода объекта выполняется командами iddata и tfest.

Перевод импульсной передаточной функции в код контроллера

Модель Рисунок 3 импульсной передаточной функции (3а) дискретного объекта (1) можно представить в виде схемы показанной на Рисунок 7.

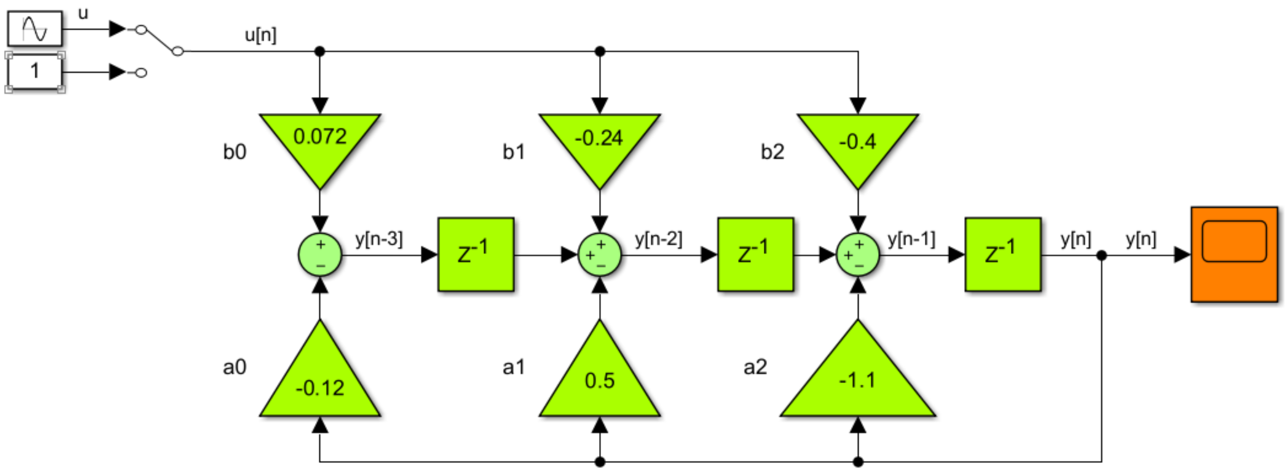


Рисунок 7. Модель импульсной передаточной функции $\frac{-0.4z^2 - 0.24z + 0.072}{z^3 - 1.1z^2 + 0.5z - 0.12}$. Блок z^{-1} реализует задержку на 1 такт. Для рассматриваемого объекта установлен период дискретизации в 1 с.

Работа схемы Рисунок 7 подобна реализации Рисунок 8, в которой блок fcn, содержащий m-функцию пользователя, выполняет совместно с z^{-1} блоками работу рассматриваемой импульсной ПФ.

```
function [yn,ym1,ym2,ym3] = fcn(u,ym1,ym2,ym3)

b2 = -0.4; b1 = -0.24; b0 = 0.072;
a3 = 1; a2 = -1.1; a1 = 0.5; a0 = -0.12;

yn = a3*ym1;
ym1 = b2*u - a2*yn + ym2
ym2 = b1*u - a1*yn + ym3
ym3 = b0*u - a0*yn
```

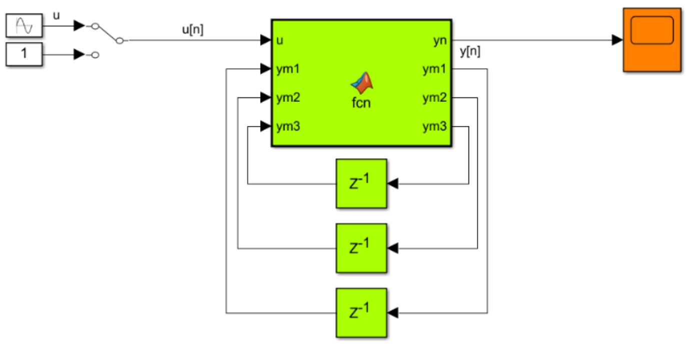


Рисунок 8. Simulink модель эквивалентной импульсной передаточной функции с fcn блоком, содержащим представленный m-код.

Следующая программа вычисляет реакцию объекта, представленного импульсной ПФ. Полученная реакция (**Ошибка! Источник ссылки не найден.**) полностью совпадает с реакциями дискретных моделей рассматриваемого объекта и соответствует реакциям непрерывных моделей этого же объекта (Рисунок 10).

```
b2 = -0.4; b1 = -0.24; b0 = 0.072;  % коэффициенты полинома числителя ИПФ
a3 = 1; a2 = -1.1; a1 = 0.5; a0 = -0.12; % коэффициенты полинома знаменателя ИПФ

y = zeros(1,50); % резервирование памяти для хранения реакции
u = ones(1,50); % единичное воздействие

ym1 = 0; ym2 = 0; ym3 = 0 % начальные значения

for i = 1:50
    y(i) = a3*ym1;
    ym1 = b2*u(i)-a2*y(i)+ym2
    ym2 = b1*u(i)-a1*y(i)+ym3
    ym3 = b0*u(i)-a0*y(i)
end

% График реакции дискретных моделей

if 1
    % Построение графиков
    figure (1) % активное окно графопостроителя
    clf  % очистка предыдущих данных
    plot(y,'xb') % график y[n] m-файла
    hold on
    plot(out.ScopeData1.signals(3).values)  % график y[n] Simulink модели, ver.1
    plot(out.ScopeData1.signals(4).values)  % график y[n] Simulink модели, ver.2
    plot(out.ScopeData1.signals(5).values)  % график y[n] Simulink модели, ver.3
    plot(out.ScopeData1.signals(6).values)  % график y[n] Simulink модели, ver.4
    legend('m-file','Simulink v.1','Simulink v.2','Simulink v.3','Simulink v.4')
    grid
    xlabel('Номер отсчета')
    ylabel('Амплитуда реакции'),
    title('Сравнение реакций дискретных моделей')
end
```

(7)

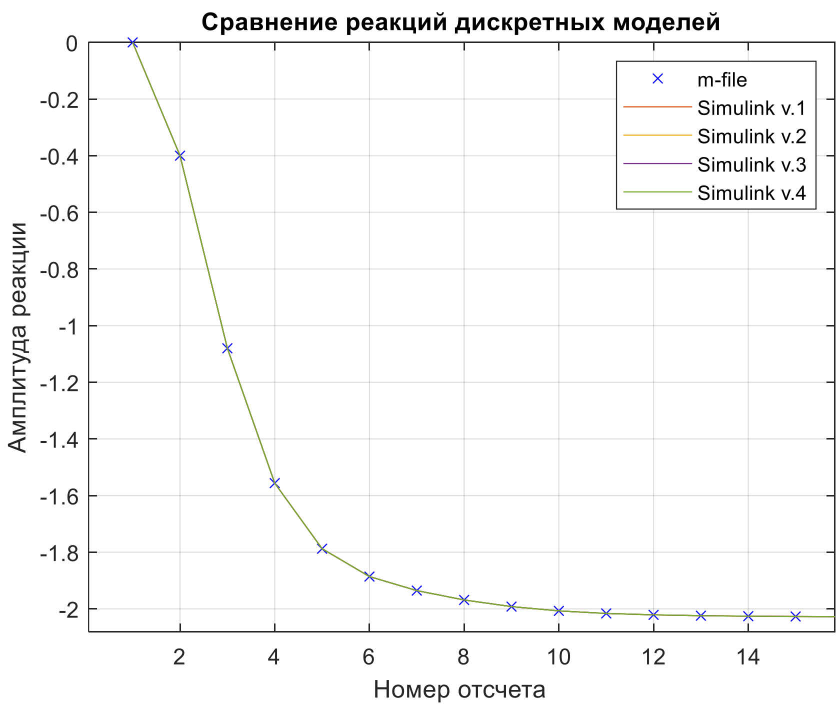


Рисунок 9. 100% совпадение реакций дискретных моделей и вычислений m-программы.

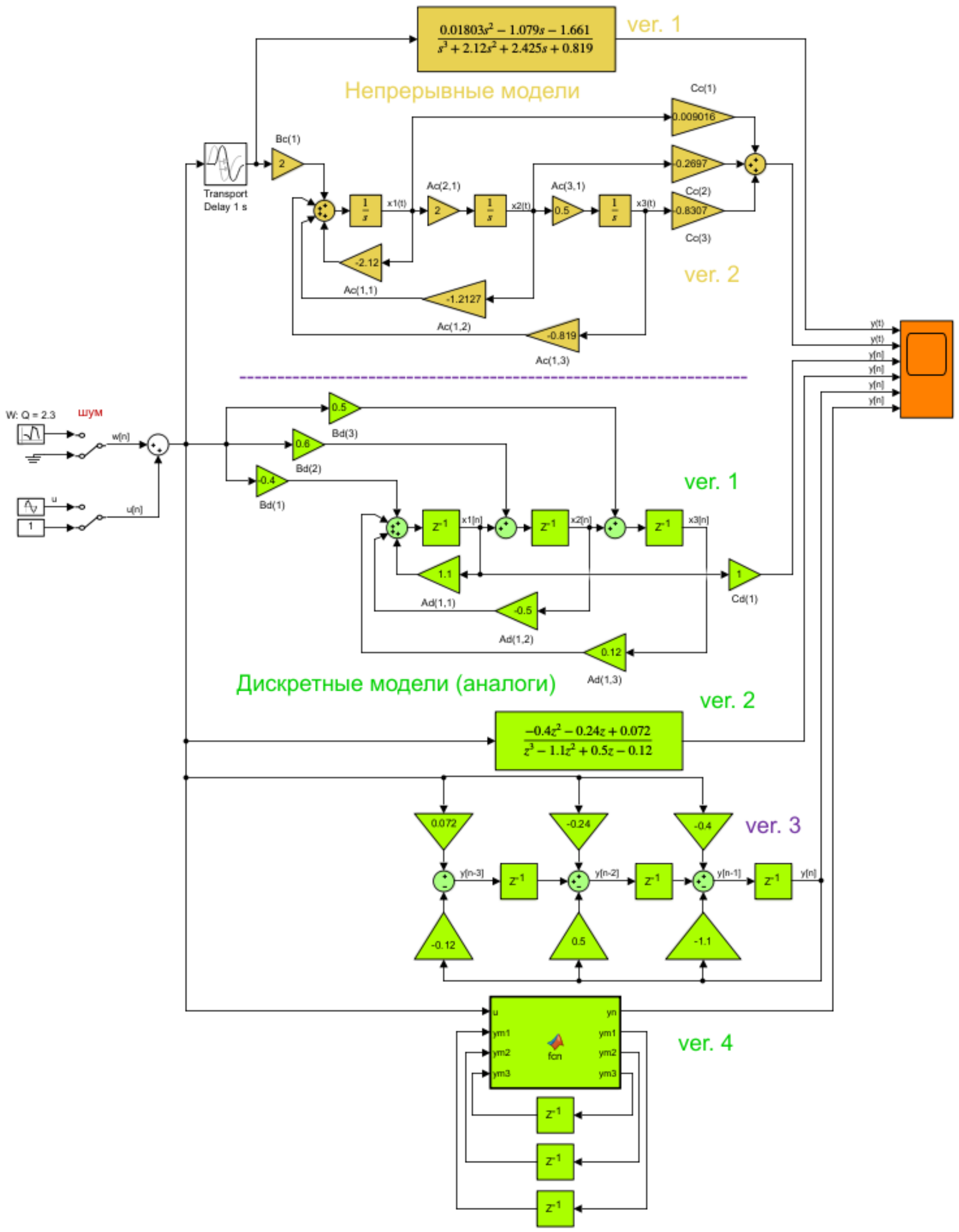


Рисунок 10. Варианты построения эквивалентных Simulink моделей рассматриваемого объекта.

Порядок программирования динамических моделей контроллерами

Для построения модели динамической системы средствами контроллеров предлагается выполнить следующее.

1. Средствами MATLAB преобразовать динамическую систему в форму импульсной передаточной функции для требуемого шага дискретизации (3а).
2. Выделить коэффициенты полиномов числителя и знаменателя импульсной ПФ (Рисунок 7)
3. Средствами контроллера реализовать последовательность вычисления импульсной ПФ (7)

Структура и порядок динамической системы могут отличаться от рассмотренного примера.

Заключение

Рассмотренные варианты преобразования форм представления динамических моделей в интегрированной среде MATLAB могут быть использованы для реализации моделей средствами контроллеров путем построения кода эквивалентной импульсной передаточной функции (Рисунок 11).

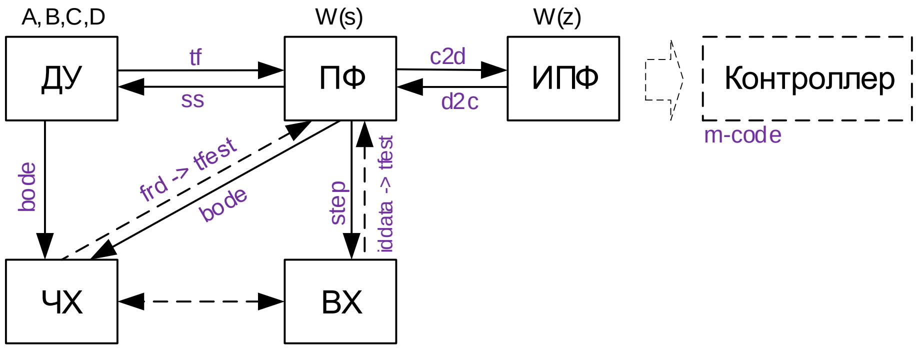


Рисунок 11. Операторы MATLAB преобразования моделей и построения эквивалентного кода для программирования контроллеров.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Задание 1. Реализация модели пространства состояний программными средствами.

Исходные параметры непрерывной модели:

$$A_d = \begin{bmatrix} -2.7 & -0.5 & -0.4 \\ 1 & 0 & 0 \\ 0 & 0.25 & 0 \end{bmatrix}; \qquad B_d = \begin{bmatrix} 8 \\ 0 \\ 0 \end{bmatrix}; \qquad (1)$$

$$C_d = [0\ 0.25\ 0]; \qquad D_d = [0]$$

Требуемый период работы программы: 2 с.

1. Введите исходные данные.

```
Ac = [-2.7  -0.5  -0.4
      1     0     0
      0  0.25     0];

Bc = [ 8;
     0;
     0];

Cc = [0  0.25 5];

Dc = [0];
```

2. Постройте структуру непрерывной модели в пространстве состояний. Обратите внимание на формат полученной структуры в Workspace.

```
Wo_ss_c = ss(Ac,Bc,Cc,Dc);
```

3. Постройте соответствующую непрерывную передаточную функцию.

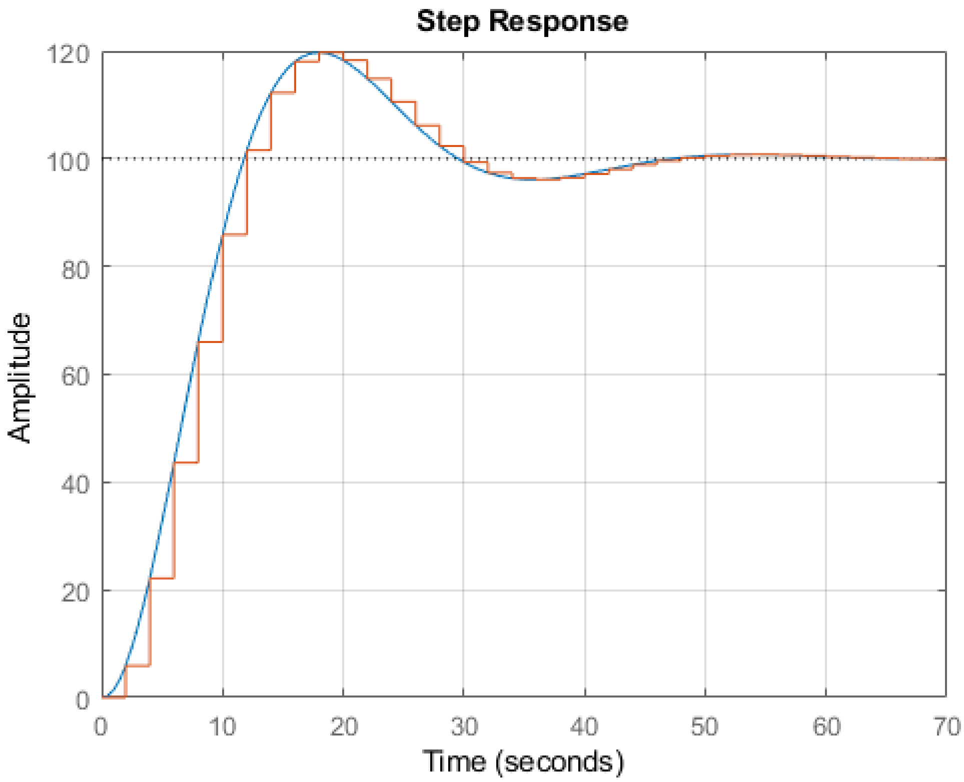
```
Wo_tf_c = tf(Wo_ss_c)
   Wo_tf_c =
              2 s + 10
      ---------------------------
      s^3 + 2.7 s^2 + 0.5 s + 0.1
```

4. Переведите модель в формат эквивалентной импульсной ПФ с периодом дискретизации 0.1с.

```
Wo_tf_d = c2d(Wo_tf_c,2)
   Wo_tf_d =
          5.908 z^2 + 6.96 z + 0.1809
      -------------------------------------
      z^3 - 1.569 z^2 + 0.7039 z - 0.004517
```

5. Постройте графики реакции на единичное ступенчатое воздействие непрерывного и дискретного объекта.

```
step(Wo_tf_c,Wo_tf_d)
grid
```



6. Измените период дискретизации модели на 0.5 с. Постройте графики реакций непрерывного и дискретного объекта.

```
Wo_tf_d1 = d2d(Wo_tf_d,0.5)
```

```
    Wo_tf_d1 =
       0.3181 z^2 + 0.3889 z - 0.02805
      ----------------------------------
      z^3 - 2.187 z^2 + 1.453 z - 0.2592
step(Wo_tf_c,Wo_tf_d1)
grid
```

Почему изменились коэффициенты импульсной ПФ?

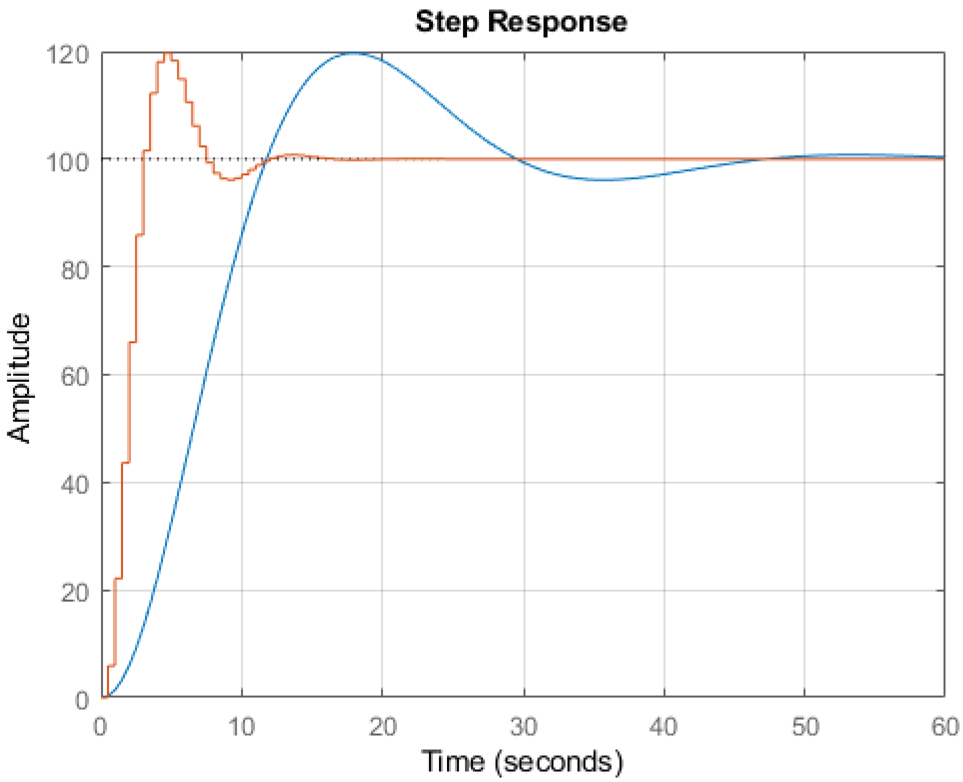
7. Выделите коэффициенты полиномов числителя и знаменателя `Wo_tf_d`. Постройте ИПФ с выделенными полиномами и периодом 0.5 s.

```
[num den] = tfdata(Wo_tf_d, 'v')
Wo_tf_d2 = tf(num,den,0.5)
    Wo_tf_d2 =

           5.908 z^2 + 6.96 z + 0.1809
      -------------------------------------
      z^3 - 1.569 z^2 + 0.7039 z - 0.004517
```

8. Постройте графики реакций непрерывного и дискретного `Wo_tf_d2` объекта.

```
step(Wo_tf_c,Wo_tf_d2)
grid
```



Почему реакция дискретного объекта стала опережать реакцию непрерывного объекта?

9. Используя пример структуры ИПФ (Рисунок 7) и ее программную реализацию (7), напишите программу вычисления реакции ИПФ на единичное ступенчатое воздействие, постройте график реакции и сравните его с предыдущими графиками.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Назовите основные формы описания динамических систем.
2. Какие средства MATLAB доступны для взаимного преобразования следующих форм моделирования: дифференциальные уравнения, передаточные функции, частотные характеристики, временные характеристики?
3. Опишите последовательность реализации импульсной передаточной функции программными средствами?
4. Покажите вариант перевода модели объекта из формы пространства состояний в частотную область.

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. Help MATLAB.
3. Dr. Bob Davidov. Синтез наблюдателя состояний. http://portalnp.ru/2021/01/10862
4. Dr. Bob Davidov. Аналитическое конструирование линейно-квадратичного регулятора http://portalnp.ru/2021/01/10857
5. Dr. Bob Davidov. Фильтр Калмана в линейно-квадратичном гауссовском управлении. http://portalnp.ru/2021/02/10877
6. Dr. Bob Davidov. Компьютерные технологии управления в технических системах http://portalnp.ru/author/bobdavidov