**1. Cel i zakres ćwiczenia**

* zapoznanie się ze sposobami zapisu modelu obiektów automatyki w przyborniku
* Control Toolbox,
* przekształcanie schematów blokowych układów automatyki,
* wykreślanie charakterystyk czasowych i częstotliwościowych obiektów
* dynamicznych,
* tworzenie i uruchamianie m-skryptów oraz układów blokowych Simulinka,
* przeprowadzanie symulacji, grupowanie i maskowanie bloków w przyborniku
* Simulink,
* wyznaczanie sterowalności i obserwowalności obiektów automatyki,
* badania stabilności układów,
* wyznaczanie linii pierwiastkowych,
* wyznaczanie postaci kanonicznych obiektów.

**2. Treść realizowanych zadań**

**Zadanie 1**

Dla wybranego układu z tabeli 1.3 zestawu O03 P3 należy napisać kod programu pozwalający na wyznaczenie transmitancji zastępczej całego układu. Następnie należy:

1. Wyznaczyć ciągły model transmitancyjny
2. Wyznaczyć minimalną reprezentację modelu obiektu w przestrzeni stanu
3. Wyznaczyć dyskretny model metodą ekstrapolatora zerowego rzędu (Zero-order Hold Method) dla T=0.1[s]
4. Wyznaczyć równanie charakterystyczne
5. Wyznaczyć wartości własne i zera układu
6. Wyznaczyć mapę biegunów i zer
7. Wyznaczyć częstotliwości własne układu, bezwymiarowy współczynnik tłumienia
8. Wyznaczyć odpowiedz obiektu na wymuszenie impulsowe i skokowe jednostkowe
9. Wyznaczyć charakterystyki częstotliwościowe obiektu: Nyquista (tylko dla dodatnich częstotliwości) i Bodego
10. Wyznaczyć odpowiedz obiektu na wymuszenia sinusoidalne t);

Przyjąć odpowiednio parametry i wymuszenia tak, aby potwierdzić wyniki otrzymane za pomocą charakterystyk częstotliwościowych

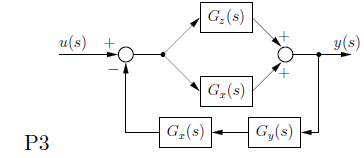
1. Określić zapas fazy i modułu obiektu
2. Wykreślić linie pierwiastkowe
3. Przeprowadzić próbę badań stabilności obiektu dobierając wzmocnienie krytyczne metodą Evansa
4. Zapisać układ w przestrzeni stanu
5. Sprawdzić sterowalność i obserwowalność układu
6. Wyznaczyć postacie kanoniczne (modalną, obserwowalną, sterowalną, Kalmana i Jordana)

**Zadanie 2**

Dla układu z zadania 1 należy zbudować jego model w Simulinku. Wykreślić odpowiedz układu na wymuszenie skokowe. Zgrupować i zamaskować poszczególne człony w jeden układ (pozostawić wartości współczynników transmitancji). Podczas maskowania przyjąć co najmniej 2 parametry.

**3. Rozwiązania realizowanych zadań**

Zestaw O03 P3



Rys. 1. Schemat blokowy układu regulacji

gdzie transmitancje bloków:

=;

=;

=

**Realizacja zadania 1.**

1. **Wyznaczyć ciągły model transmitancyjny**

%% Wyznaczenie ciągłego modelu transmisyjnego

%Zapisanie członu opóźniającego

[l,m]=pade(0.8,3)

%% Wyznaczenie modelu zastępczego

%Zapis Gx(s)

Gx1=tf(l,m)

Gx2=tf([1],[2 1])

Gx=series(Gx1,Gx2)

%Zapis Gy(s)

Gy=tf([4],[1 0.5])

%Zapis Gz(s)

Gz=tf([3],[2 0 1])

G1=parallel(Gz,Gx)

G2=series(Gy,Gx)

% 1. Transmitancja wypadkowa

G=feedback (G1,G2)

Transmitancja wypadkowa G(s):



Transmitancja wypadkowa G(s) posiada dziesiąty rząd licznika oraz jedenasty rząd mianownika.

1. **Wyznaczyć minimalną reprezentację modelu obiektu w przestrzeni stanu**

%% Wyznaczenie minimalnej reprezentacji modelu zastępczego

Gr=minreal(G)

Zminimalizowana transmitancja Gr (s):



Funkcja minreal skraca bliskie zera i bieguny redukując rząd układu. Jak widać, w powyższej transmitancji zredukowanej Gr(s), rząd licznika i mianownika nie uległ zmianie w stosunku do G(s). Zmieniły się natomiast współczynniki przy poszczególnych potęgach.

%% Zapisanie minimalnej reprezentacji modelu w przestrzeni stanów

[liczl,mianl]=tfdata(Gr,'v')

%Macierze układu w postaci jawnej

[Ar,Br,Cr,Dr]=tf2ss(liczl,mianl)

%Model obiektu w przestrzeni stanu

Gr\_ss=ss(Ar,Br,Cr,Dr)

**Model obiektu w przestrzeni stanów**

Macierz stanu A:



Macierz wejść B: 

Macierz wyjść C:

Macierz bezpośredniego wpływu wejścia na wyjście D:



Zamodelowany układ posiada jedenaście zmiennych stanu (od x1 do x11), jedno wejście określone na zmiennej stanu x1 oraz jedenaście wyjść. W tym modelu wejście nie wpływa bezpośrednio na wejście, zatem macierz D wynosi 0.

1. **Wyznaczyć dyskretny model metodą ekstrapolatora zerowego rzędu (Zero-order Hold Method) dla T=0.1[s]**

%% Wyznaczenie dyskretnego modelu metodą eksploratora zerowego rzędu

Gd=c2d(Gr,0.1,'zoh')

Transmitancja modelu dyskretnego Gd(s):



Model dyskretny został utworzony na podstawie modelu ciągłego przy użyciu ekstrapolatora zerowego rzędu. Jest to matematyczny model, który opisuje układ dokonując konwersji sygnału poprzez podtrzymanie wartości każdej z próbek przez jeden okres próbkowania.

1. **Wyznaczyć równanie charakterystyczne**

%% Wyznaczenie równania charakterystycznego

I=eye(size(Ar)); %utworzenie macierzy jednostkowej I

s=sym('s');

Row\_ch=det(s\*I-Ar)% wyznacznik

Rów\_ch=



Równaniem charakterystycznym jest wielomian utworzony z mianownika transmitancji G(s).

1. **Wyznaczyć wartości własne i zera układu**

%% Wyznaczenie wartości własnych (biegunów)i zer układu

bieguny=pole(Gr)

zera=zero(Gr)

%[V,d]=eig(Ar) to samo co pole

Wartości własne układu:

bieguny =

-8.9898 + 0.0000i

-6.3700 + 5.1481i

-6.3700 - 5.1481i

-3.5127 + 5.1812i

-3.5127 - 5.1812i

-1.3606 + 2.0884i

-1.3606 - 2.0884i

0.6445 + 0.7945i

0.6445 - 0.7945i

-0.6562 + 0.5538i

-0.6562 - 0.5538i

Zera układu:

zera =

17.2279 + 0.0000i

1.0616 + 5.9536i

1.0616 - 5.9536i

-4.5973 + 4.3860i

-4.5973 - 4.3860i

-5.8055 + 0.0000i

-0.6756 + 0.5363i

-0.6756 - 0.5363i

-0.5000 + 0.0000i

-0.5000 - 0.0000i

Wartości własne układu (bieguny) są pierwiastkami równania charakterystycznego (mianownika transmitancji). Zera układu to pierwiastki wielomianu określającego licznik transmitancji modelu. Powyższy model posiada jeden biegun rzeczywisty i pięć podwójnych biegunów zespolonych oraz cztery zera rzeczywiste (w tym jedno zero podwójne) i trzy podwójne zera zespolone.

1. **Wyznaczyć mapę biegunów i zer**

%% Mapa biegunów i zer

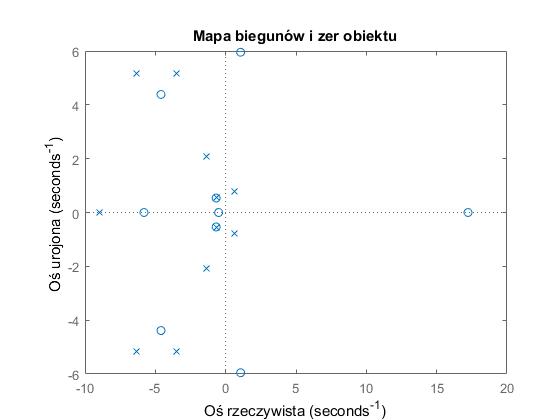
figure(1)

pzmap(Gr)

title('Mapa biegunów i zer obiektu');

xlabel('Oś rzeczywista');

ylabel('Oś urojona');

Jak widać z wykresu położenia zer i biegunów, układ posiada dwa bieguny leżące po dodatniej stronie osi rzeczywistej, a więc jest niestabilny. ****

Rys. 2. Mapa biegunów i zer obiektu

1. **Wyznaczyć częstotliwości własne układu, bezwymiarowy współczynnik tłumienia**

%% Częstotliwości własne układu, bezwymiarowy współczynnik tłumienia

[wn z p ]=damp(Gr)

%czesto\_wn – wektor częstotliwości własnych układu

%wsp\_tlum – wektor współczynników tłumienia układu

%p – wektor wartości własnych

Wektor częstotliwości własnych układu [Hz]:



Wektor współczynników tłumienia układu:



Częstotliwości własne układu są częstotliwościami drgań, jakie może wykonać układ po wytrąceniu z położenia równowagi i odizolowaniu od wpływu oddziaływań zakłócających równowagę. Są to wartości bezwzględne z wartości własnych.

Układ jest słabo tłumiony, ponieważ współczynniki tłumienia są mniejsze od jedynki. Oznacza to również, że w układzie występują pierwiastki zespolone.

1. **Wyznaczyć odpowiedz obiektu na wymuszenie impulsowe i skokowe jednostkowe**

%% Odpowiedź obiektu na wymuszenie impulsowe i skokowe jednostkowe

figure(2)

impulse(Gr,50)

xlabel('Czas [s]')

ylabel('Amplituda [m]')

title('Odpowiedź obiektu na wymuszenie impulsowe')

figure(3)

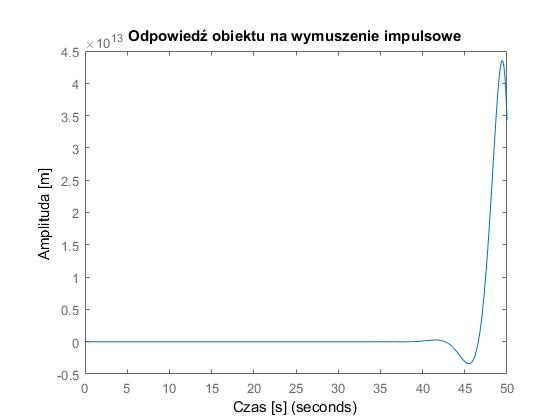
step(Gr,50)

xlabel('Czas [s]')

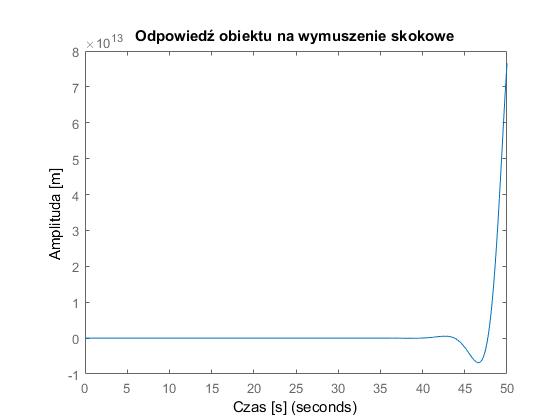
ylabel('Amplituda [m]')

title('Odpowiedź obiektu na wymuszenie skokowe')

Z odpowiedzi układu na wymuszenie skokowe oraz impulsowe widać, że układ nie osiągają stanu ustalonego w czasie dążącym do nieskończoności. Jest on zatem niestabilny.



Rys. 3. Odpowiedź obiektu na wymuszenie impulsowe



Rys. 4. Odpowiedź obiektu na wymuszenie skokowe

1. **Wyznaczyć charakterystyki częstotliwościowe obiektu: Nyquista (tylko dla dodatnich częstotliwości) i Bodego**

%% Charakterystyka Nyquista i Bodego

figure(4)

str=nyquistoptions;

str.ShowFullContour='off';%wyłączenie ujemnych częstotliwości

str.XLabel.String='Oś rzeczywista';

str.YLabel.String='Oś urojona';

str.Title.FontSize=12;

str.FreqUnits='Hz';%zmiana jednostek na osi x

nyquist(Gr,str);

title('Charakterystka Nyquista');

figure(5)

str1=bodeoptions;

str1.XLabel.String='Oś rzeczywista';

str1.Ylabel.String='Oś urojona';

str1.Ylabel.String={'Moduł' 'Faza'};

str1.Xlabel.String='Częstotliwośc';

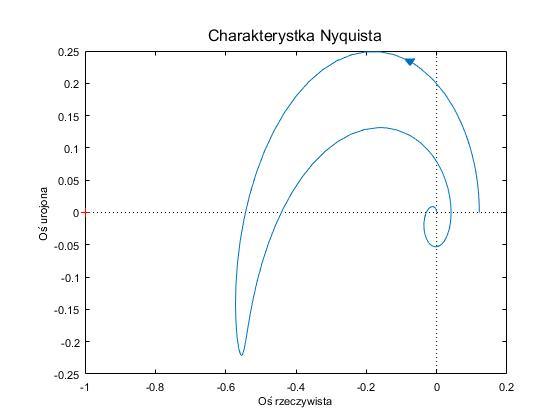
str1.Title.FontSize=12;

str1.FreqUnits='Hz';

bode(Gr,str1);

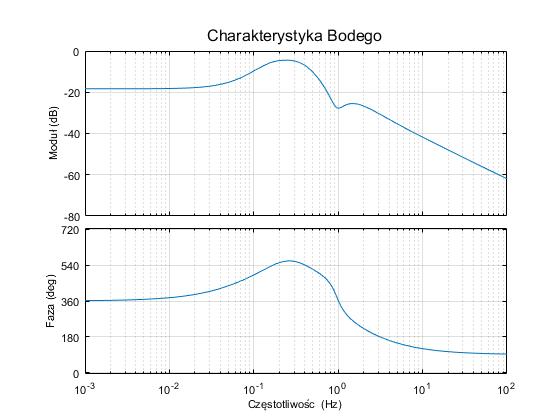
title('Charakterystyka Bodego');

grid on;

****

Rys. 5. Charakterystyka Nyquista

Z przebiegu charakterystyki Nyquista widać, że charakterystyka amplitudowo-fazowa układu otwartego nie obejmuje punktu (-1,j0), zatem układ zamknięty jest stabilny.

****

Rys. 6. Charakterystyka Bodego

Na podstawie analizy charakterystyki Bodego można stwierdzić, że badany model nie jest stabilny, ponieważ nie posiada ujemnej wartości logarytmicznej charakterystyki amplitudowej (zapasu modułu) dla pulsacji odpowiadającej przesunięciu fazowemu -180°.

1. **Wyznaczyć odpowiedz obiektu na wymuszenia sinusoidalne t); Przyjąć odpowiednio parametry i wymuszenia tak, aby potwierdzić wyniki otrzymane za pomocą charakterystyk częstotliwościowych**

%% Odpowiedź na wymuszenie sinusoidalne

t=[0:0.1:20]

fz=0.5\*2\*pi %częstotliwość f=0.5 Hz

az=1 %amplituda=1m

z1=az\*sin(fz\*t)

figure(6)

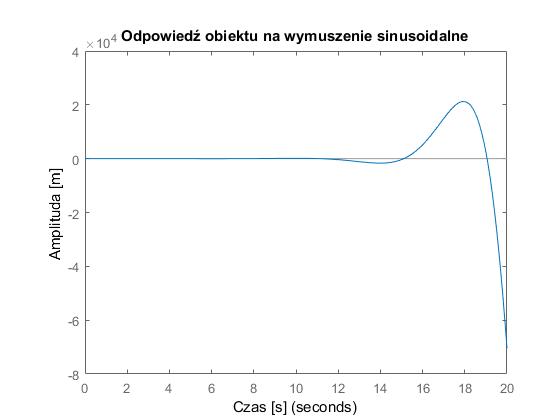
lsim(Gr,z1,t)

xlabel('Czas [s]')

ylabel('Amplituda [m]')

title('Odpowiedź obiektu na wymuszenie sinusoidalne')

Modelowany układ jest niestabilny, nie można więc określić zapasu modułu i fazy oraz odpowiadającej jej częstotliwości. Na wykresie przedstawiono odpowiedź układu na wymuszenie sinusoidalne o amplitudzie równej 1 m i częstotliwości 0.5 Hz. Układ pozostaje nadal niestabilny.

****

Rys. 7. Odpowiedź obiektu na wymuszenie sinusoidalne

1. **Określić zapas fazy i modułu obiektu**

%% Zapas fazy i modułu

[Gm,Pm,Wgm,Wpm] = margin(Gr)

%Gm- wartośc zapasu modułu, Wgm- częstotliwość zapasu modułu

%Pm- wartośc zapasu fazy, Wpm- częstotliwość zapasu fazy

Wartość zapasu modułu:

Gm = 1.8383 [dB]

Częstotliwość zapasu modułu:

Pm = Inf [Hz]

Wartość zapasu fazy:

Wgm = 1.0567 [deg]

Częstotliwość zapasu fazy:

Wpm = NaN [Hz]

Zapas modułu wynosi 1.8383 [db] dla częstotliwości dążącej do nieskończoności. Zapas fazy natomiast nie istnieje. Potwierdza to niestabilność modelowanego układu.

1. **Wykreślić linie pierwiastkowe**

%% Linie pierwiastkowe i próba badań stabilności metodą Evansa

figure(7)

h = rlocusplot(Gr);

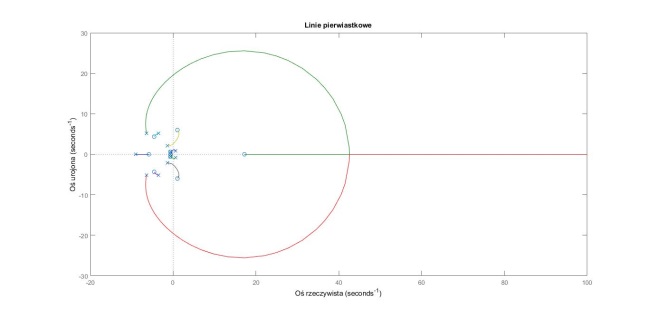
pp = getoptions(h);

pp.Title.String = 'Linie pierwiastkowe';

pp.XLabel.String='Oś rzeczywista';

pp.YLabel.String='Oś urojona';

setoptions(h,pp); %Zatwierdzenie zmian na wykresie

****

Rys. 8. Linie pierwiastkowe

Na podstawie wykresy linii pierwiastkowych (charakterystyka Evansa) można stwierdzić, że układ jest niestabilny, ponieważ część linii znajduje się po prawej stronie płaszczyzny zespolonej.

1. **Przeprowadzić próbę badań stabilności obiektu dobierając wzmocnienie krytyczne metodą Evansa**

Badany układ otwarty jest niestabilny, jednak charakterystyka Nyquista wskazuje, że układ zamknięty jest stabilny, a więc można znaleźć wzmocnienie krytyczne. Aby sprawdzić wartość wzmocnienia krytycznego dla układu, przybliżono jedną z linii łączących zero i biegun na wykresie Evansa. Następnie uzyskując bardzo duże przybliżenie, określono przedział wzmocnień w którym może znajdować się zerowe położenie bieguna, a więc układ znalazłby się na granicy stabilności. Poszukiwane wzmocnienie k wynosiło ok. 1.84. Układ został połączony w sprzężeniu razem ze wzmocnieniem. Do poszukiwania wzmocnienia wykorzystano pętlę for. Szukane wzmocnienie graniczne k=1.8445.

for i=1.84:0.0001:1.85

figure(9);

hold on;

step(feedback(i\*Gr,1));

hold on;

end

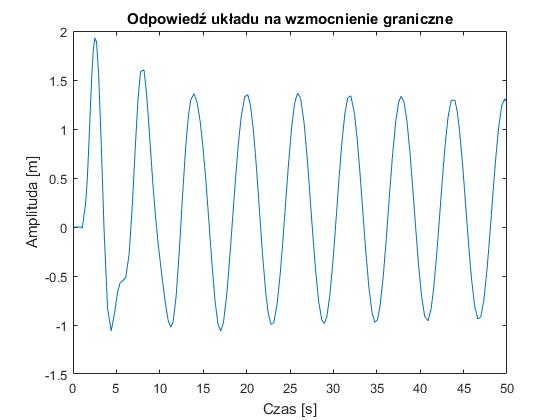
figure(10)

plot(kgran.Time,kgran.Data)

xlabel('Czas [s]');

ylabel('Amplituda [m]');

title('Odpowiedź układu na wzmocnienie graniczne')



Rys. 9. Odpowiedź układu na wymuszenie skokowe w pętli ze wzmocnieniem granicznym

1. **Zapisać układ w przestrzeni stanu**

Macierz stanu A:



Macierz wejść B: 

Macierz wyjść C:



Macierz bezpośredniego wpływu wejścia na wyjście D:



1. **Sprawdzić sterowalność i obserwowalność układu**

%% Sterowalność

S=ctrb(Ar,Br)

rank(Ar)

rank(S)

if rank(Ar)==rank(S)

disp('Układ jest sterowalny')

else

disp('Układ nie jest sterowalny')

end

%% Obserwowalność

O=obsv(Ar,Cr)

rank(Ar)

rank(O)

if rank(Ar)==rank(O)

disp('Układ jest obserwowalny')

else

disp('Układ nie jest obserwowalny')

end

Układ liniowy jest sterowalny, jeżeli dla dowolnego stanu początkowego możemy zastosować takie sterowanie, które w skończonym czasie pozwoli na sprowadzenie sygnału wyjściowego do 0. Układ jest sterowalny, jeśli rząd macierzy sterowalności jest równy rzędowi macierzy A. Rząd macierzy stanu A wynosi 11. Rząd macierzy sterowalności wynosi 7, zatem układ nie jest sterowalny. Potwierdza to niestabilność układu. Układ jest obserwowalnym jeśli na podstawie odczytu sygnału sterującego oraz sygnału wejściowego można określić wewnętrzny stan obiektu. Układ jest obserwowalny, jeśli rząd macierzy obserwowalności jest równy rzędowi macierzy A. Rząd macierzy obserwowalności wynosi 11, a więc układ jest obserwowalny.

1. **Wyznaczyć postacie kanoniczne (modalną, obserwowalną, sterowalną, Kalmana i Jordana)**

**Postać kanoniczna modalna**

%%Modalna/diagonalna

M=canon(Gr,'modal')

Macierz stanu A:

****

Macierz wejść B:

****

Macierz wyjść C:

****

Macierz bezpośredniego wpływu wejścia na wyjście D:

****

Po przekształceniu układu do postaci kanonicznej (diagonalnej) zera i bieguny układu znajdują się na przekątnej macierzy stanu A.

**Postać kanoniczna sterowalna i obserwowalna**

%%Obserwowalna=Sterowalna'

[Aster,Bster,Cster,Tster,kster]=ctrbf(Ar,Br,Cr)

[Aobsv,Bobsv,Cobsv,Tobsv,k] = obsvf(Ar,Br,Cr)

%Inny sposób wyznaczenia macierzy sterowalności

Aobser=Aster';

Bobser=Cster';

Cobser=Bster';

**Postać kanoniczna sterowalna**

Macierze stanu:







**Postać kanoniczna obserwowalna:**







**Postać kanoniczna Kalmana**

Gss=ss(Gr)

[sys,Ts]=canon(Gss,'modal')

Akalm=inv(Ts)\*Ar\*Ts

Bkalm=inv(Ts)\*Br

Ckalm=Cr\*Ts

Postać kanoniczna Kalmana dzieli układ na 4 rozłączne części: sterowalną i nieobserwowalną, sterowalną i obserwowalną, niesterowalną i nieobserwowalną, niesterowalną i obserwowalną.











**Postać kanoniczna Jordana**

disp('Postać kanoniczna Jordana');

[V,D]=eig(Ar);

[V1,J1] = jordan(Ar);



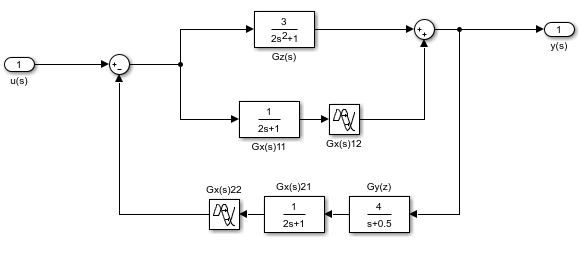
**Macierz Jordana J1=**



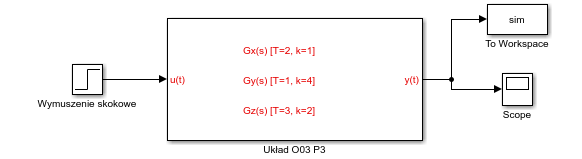
Na przekątnej macierzy Jordana tworzą się klatki, składające się z wartości własnych układu oraz liczby jeden ponad nimi. Na tej podstawie można określić wielokrotności wartości własnych układu. Klatki zawierające wielokrotne wartości własne tworzą podmacierze kwadratowe. Macierz V1 jest macierzą przekształcenia przez podobieństwo sprowadzającą macierz A do postaci Jordana J1.

**Zadanie 2**

Dla układu z zadania 1 należy zbudować jego model w Simulinku. Wykreślić odpowiedz układu na wymuszenie skokowe. Zgrupować i zamaskować poszczególne człony w jeden układ (pozostawić wartości współczynników transmitancji). Podczas maskowania przyjąć co najmniej 2 parametry.



Rys. 10. Podgląd układu zamodelowanego w Simulinku



Rys. 11. Maska układu zamodelowanego w Simulinku

Komendy użyte do utworzenia maski:

color('red')

%opis parametrów

text(0.3,0.75,'Gx(s) [T=2, k=1]')

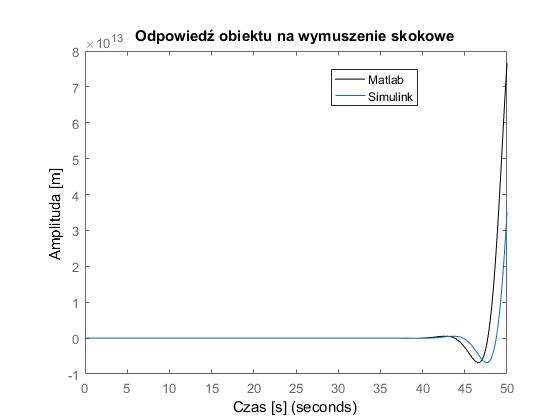
text(0.3,0.50,'Gy(s) [T=1, k=4]')

text(0.3,0.25,'Gz(s) [T=3, k=2]')

%opis wejść/wyjść

port\_label('output', 1, 'y(t)')

port\_label('input', 1, 'u(t)')

****

Rys. 12. Porównanie odpowiedzi układu na wymuszenie skokowe w Matlabie i Simulinku

Z powyższego wykresu widać, że układ w Simulinku został zamodelowany prawidłowo, ponieważ wartości odpowiedzi na wymuszenie skokowe znajdują się w tych samych zakresach skali. W obu przypadkach układy są niestabilne.

**4. Wnioski**

W zamodelowanym układzie w transmitancji otrzymano dziesiąty rząd licznika oraz jedenasty rząd mianownika. Zmniejszenie postaci transmitancji nie wpłynęło na zmniejszenie liczby biegunów oraz zer układu. Zapisanie modelu obiektu w przestrzeni stanu pozwala na określenie zależności pomiędzy zmiennymi stanu oraz wejściami i wyjściami układu. Badany układ został opisany za pomocą jedenastu zmiennych stanu, jednego wyjścia oraz jedenastu wyjść. Z mapy zer i biegunów, odpowiedzi na wymuszenie skokowe, wymuszenie impulsowe oraz rozkładu linii pierwiastkowych można było stwierdzić, że badany układ jest niestabilny. Z uwagi na ten fakt nie można było określić zapasu modułu oraz fazy, jak również odpowiedzi na wymuszenie sinusoidalne.. Odpowiedź układu na wymuszenie skokowe zrealizowane w Simulinku pokrywa się z wymuszeniem otrzymanym w środowisku Matlab. Z analizy charakterystyki Nyqista wynikło, że układ zamknięty jest stabilny. Dzięki temu możliwe było znalezienie wzmocnienia krytycznego w pętli sprzężenia z obiektem.

Badany układ nie jest sterowalny, ale jest obserwowalny. Na podstawie macierzy sterowalności można sprawdzić, czy da się sterować każdą ze zmiennych stanu niezależnie. Postać kanoniczna modalna przedstawia wartości własne układu na przekątnej. Postać kanoniczna Kalmana dzieli układ na cztery niezależne części pod względem sterowalności i obserwowalności. Postać kanoniczna Jordana pozwala na wykrycie i przedstawienie pierwiastków wielokrotnych układu przy pomocy klatek Jordana. Postać kanoniczną obserwowalną można uzyskać po transponowaniu odpowiednich macierzy postaci kanonicznej sterowalnej.

**Pełny kod użyty do realizacji projektu**

%%Projekt 1

%% Wyznaczenie ciągłego modelu transmitancyjnego

%Zapisanie członu opóźniającego

[l,m]=pade(0.8,3)

%% Wyznaczenie modelu zastępczego

%Zapis Gx(s)

Gx1=tf(l,m)

Gx2=tf([1],[2 1])

Gx=series(Gx1,Gx2)

%Zapis Gy(s)

Gy=tf([4],[1 0.5])

%Zapis Gz(s)

Gz=tf([3],[2 0 1])

G1=parallel(Gz,Gx)

G2=series(Gy,Gx)

%Transmitancja wypadkowa

G=feedback(G1,G2)

%% Wyznaczenie minimalnej reprezentacji modelu zastępczego

Gr=minreal(G)

%% Zapisanie minimalnej reprezentacji modelu w przestrzeni stanów

[liczl,mianl]=tfdata(Gr,'v')

%Macierze układu w postaci jawnej

[Ar,Br,Cr,Dr]=tf2ss(liczl,mianl)

%Model obiektu w przestrzeni stanu

Gr\_ss=ss(Ar,Br,Cr,Dr)

%% Wyznaczenie dyskretnego modelu metodą eksploratora zerowego rzędu

Gd=c2d(Gr,0.1,'zoh')

%% Wyznaczenie równania charakterystycznego

I=eye(size(Ar)); %utworzenie macierzy jednostkowej I

s=sym('s');

Row\_ch=det(s\*I-Ar)% wyznacznik

%% Wyznaczenie wartości własnych (biegunów)i zer układu

bieguny=pole(Gr)

zera=zero(Gr)

%[V,d]=eig(Ar) to samo co pole

%% Mapa biegunów i zer

figure(1)

pzmap(Gr)

title('Mapa biegunów i zer obiektu');

xlabel('Oś rzeczywista');

ylabel('Oś urojona');

%% Częstotliwości własne układu, bezwymiarowy współczynnik tłumienia

[wn z p ]=damp(Gr)

%czesto\_wn – wektor częstotliwości własnych układu

%wsp\_tlum – wektor współczynników tłumienia układu

%p – wektor wartości własnych

%% Odpowiedź obiektu na wymuszenie impulsowe i skokowe jednostkowe

figure(2)

impulse(Gr,50)

xlabel('Czas [s]')

ylabel('Amplituda [m]')

title('Odpowiedź obiektu na wymuszenie impulsowe')

figure(3)

step(Gr,50,'k')

xlabel('Czas [s]')

ylabel('Amplituda [m]')

title('Odpowiedź obiektu na wymuszenie skokowe')

legend('Matlab')

%% Charakterystyka Nyquista i Bodego

figure(4)

str=nyquistoptions;

str.ShowFullContour='off';%wyłączenie ujemnych częstotliwości

str.XLabel.String='Oś rzeczywista';

str.YLabel.String='Oś urojona';

str.Title.FontSize=12;

str.FreqUnits='Hz';%zmiana jednostek na osi x

nyquist(Gr,str);

title('Charakterystka Nyquista');

figure(5)

str1=bodeoptions;

str1.XLabel.String='Oś rzeczywista';

str1.Ylabel.String='Oś urojona';

str1.Ylabel.String={'Moduł' 'Faza'};

str1.Xlabel.String='Częstotliwośc';

str1.Title.FontSize=12;

str1.FreqUnits='Hz';

bode(Gr,str1);

title('Charakterystyka Bodego');

grid on;

%% Odpowiedź na wymuszenie sinusoidalne

t=[0:0.1:20]

fz=0.5\*2\*pi %częstotliwość f=0.5 Hz

az=1 %amplituda=1m

z1=az\*sin(fz\*t)

figure(6)

lsim(Gr,z1,t)

xlabel('Czas [s]')

ylabel('Amplituda [m]')

title('Odpowiedź obiektu na wymuszenie sinusoidalne')

%% Zapas fazy i modułu

[Gm,Pm,Wgm,Wpm] = margin(Gr)

%Gm- wartośc zapasu modułu, Wgm- częstotliwość zapasu modułu

%Pm- wartośc zapasu fazy, Wpm- częstotliwość zapasu fazy

%% Linie pierwiastkowe i próba badań stabilności metodą Evansa

figure(7)

h = rlocusplot(Gr);

pp = getoptions(h);

pp.Title.String = 'Linie pierwiastkowe';

pp.XLabel.String='Oś rzeczywista';

pp.YLabel.String='Oś urojona';

setoptions(h,pp); %Zatwierdzenie zmian na wykresie

%% Zapis układu w przestrzeni stanu

Ar,Br,Cr,Dr

%% Sterowalność

S=ctrb(Ar,Br)

rank(Ar)

rank(S)

if rank(Ar)==rank(S)

disp('Układ jest sterowalny')

else

disp('Układ nie jest sterowalny')

end

%% Obserwowalność

O=obsv(Ar,Cr)

rank(Ar)

rank(O)

if rank(Ar)==rank(O)

disp('Układ jest obserwowalny')

else

disp('Układ nie jest obserwowalny')

end

%% Postacie kanoniczne

%% Modalna/diagonalna

M=canon(Gr,'modal')

%% Obserwowalna=Sterowalna'

[Aster,Bster,Cster,Tster,kster]=ctrbf(Ar,Br,Cr)

[Aobsv,Bobsv,Cobsv,Tobsv,k] = obsvf(Ar,Br,Cr)

%canon(Gr,'companion')%postać sterowalna

%%Postać kanoniczna Kalmana

Gss=ss(Gr)

[sys,Ts]=canon(Gss,'modal')

Akalm=inv(Ts)\*Ar\*Ts

Bkalm=inv(Ts)\*Br

Ckalm=Cr\*Ts

%% Jordana

disp('Postać kanoniczna Jordana');

[V,D]=eig(Ar);

[V1,J1] = jordan(Ar);

jordan(Ar)

%% Wymuszenie skokowe simulink

figure(8)

plot(sim.Time,sim.Data)

xlabel('Czas [s]');

ylabel('Amplituda [m]');

title('Odpowiedź na wymuszenie skokowe w simulinku')

%% K krytyczne

for i=1.84:0.0001:1.85

figure(9);

hold on;

step(feedback(i\*Gr,1));

hold on;

end

figure(10)

plot(kgran.Time,kgran.Data)

xlabel('Czas [s]');

ylabel('Amplituda [m]');

title('Odpowiedź układu na wzmocnienie graniczne')