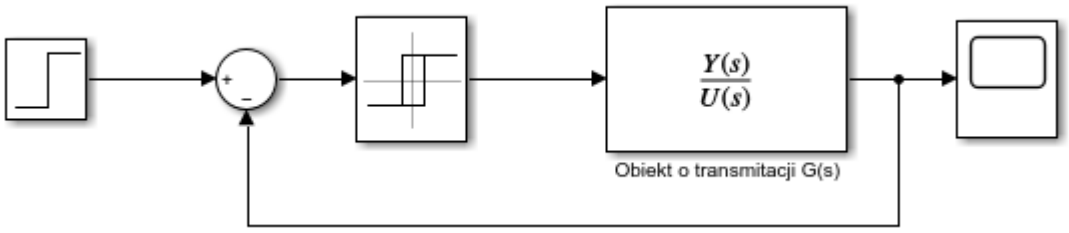
Sprawozdanie - WEAIiIB			
Podstawy automatyki			
Ćwiczenie 6: Układ regulacji 2 położeniowej – część 2			
Czwartek godz.	14.30	Data wykonania:	27.04.2023
Imię i nazwisko:	Janusz Pawlicki	Data zaliczenia:	
		Ocena:	

1. Cel ćwiczenia

Ilustracja i rozwinięcie zagadnień prezentowanych podczas ćwiczeń rachunkowych pt. "Stabilność układu regulacji z liniowym obiektem i nieliniowym regulatorem statycznym"



Rys1. Obiekt w programie Simulink

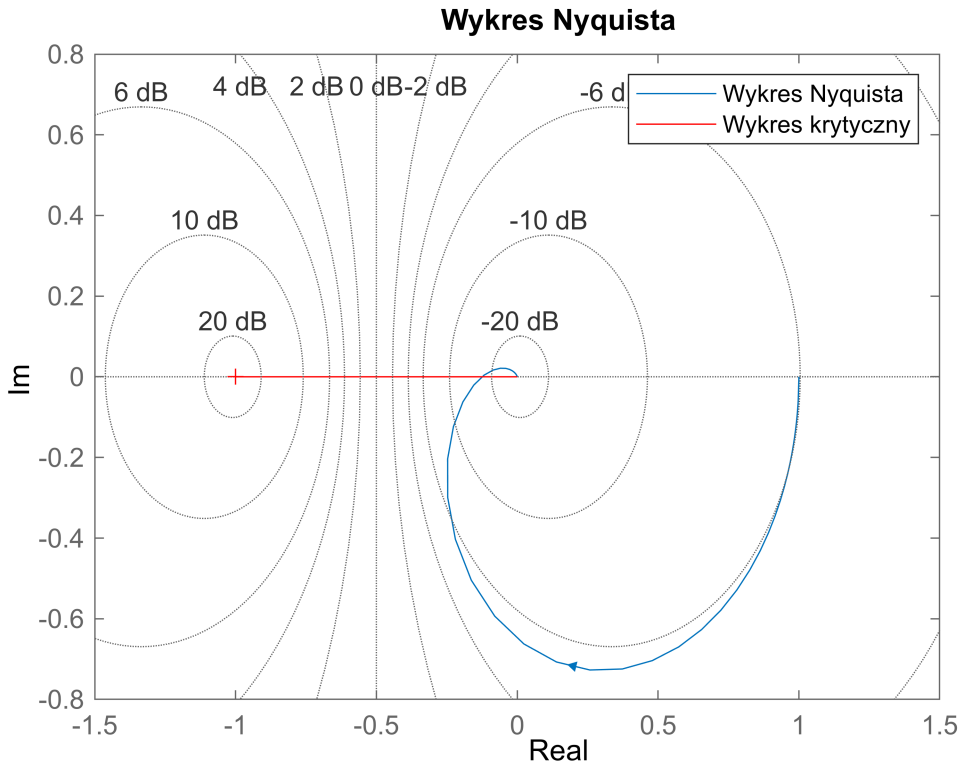
2. Przebieg ćwiczenia

$$G(s) = \frac{1}{s^3 + 3s^2 + 3s + 1}.$$

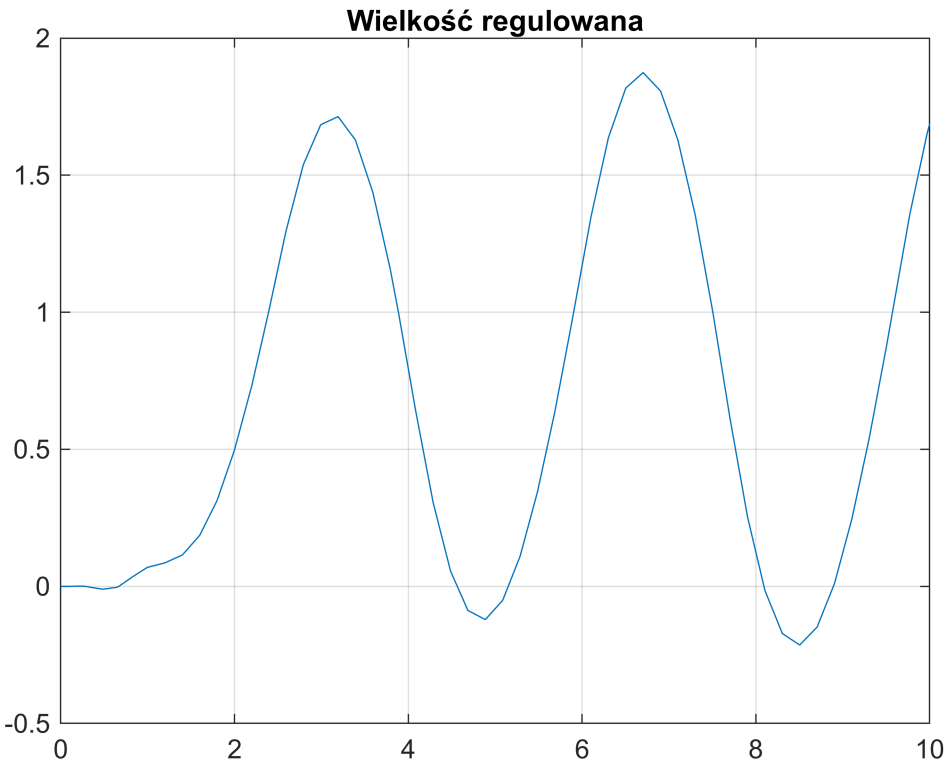
h = 0.0

```
licz = [1];

mian1 = [1 3 3 1];
mian2 = [1 2 1 0];
mian3 = [1 10 35 50 24];

h = 0;
ym = 2.1*pi;

c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
```

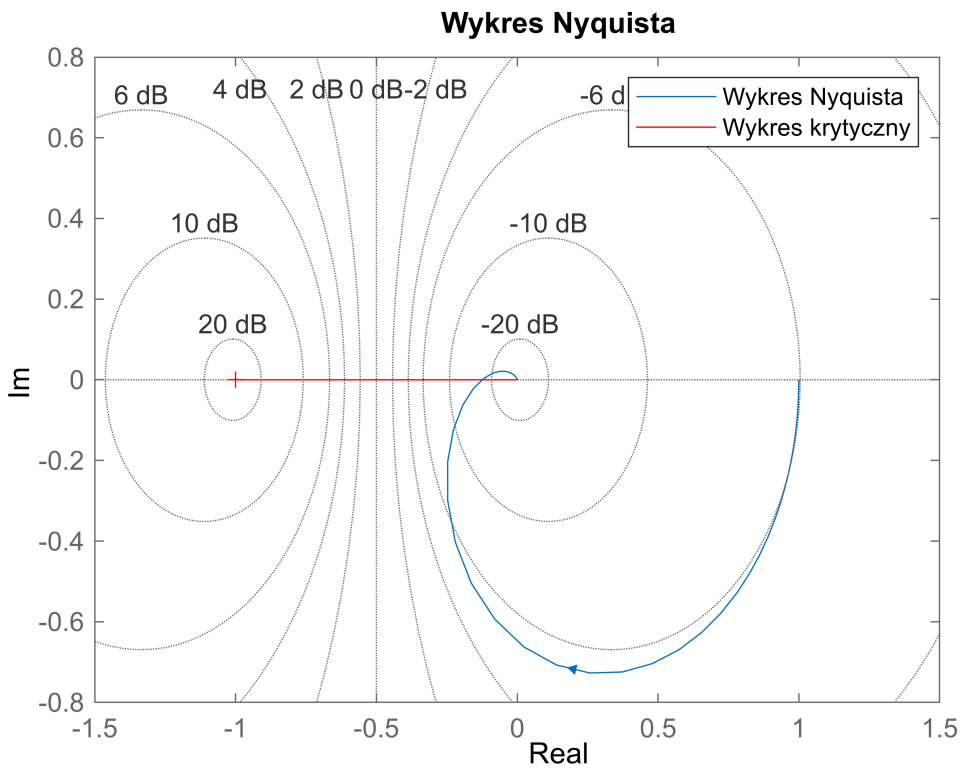
```
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



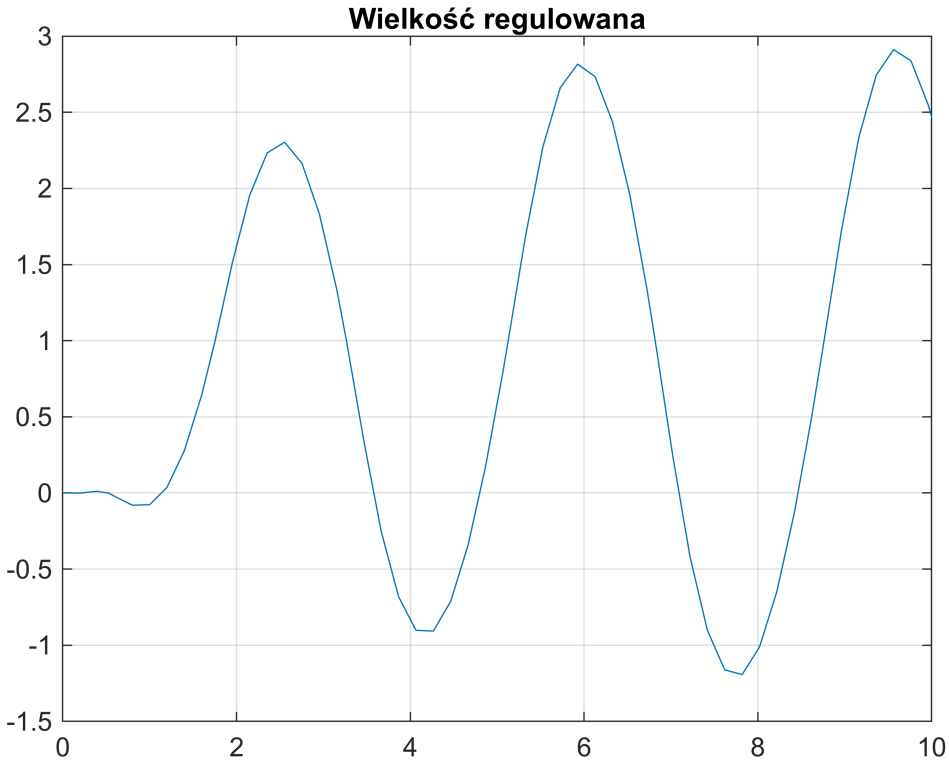

```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```



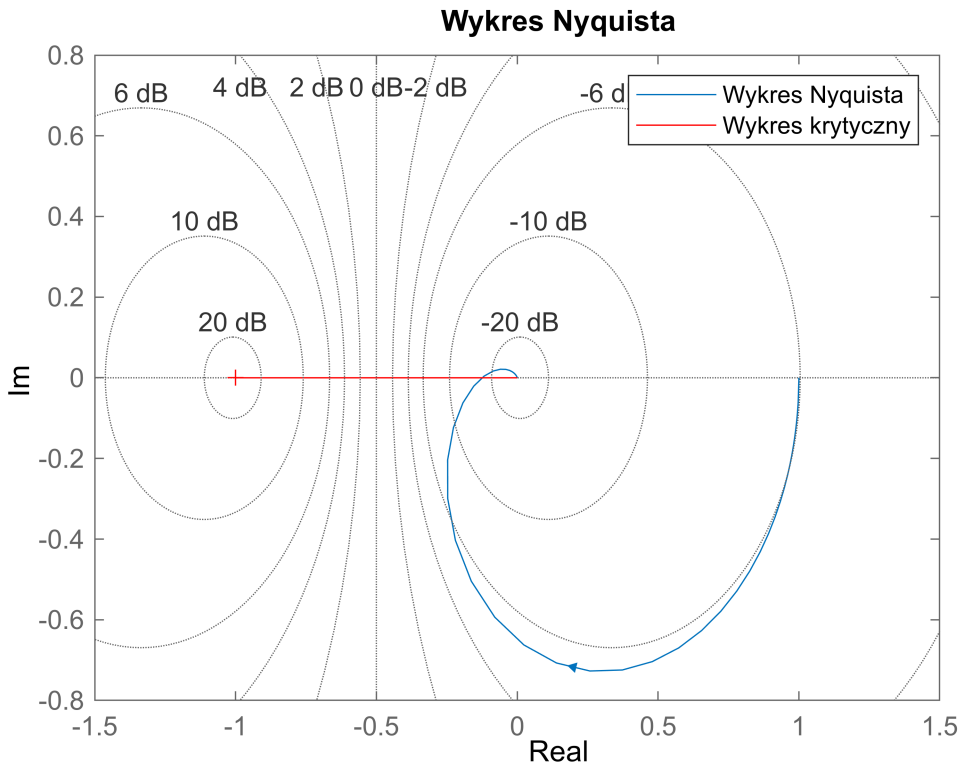

```
h = 0.0;
ym = 4.1*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



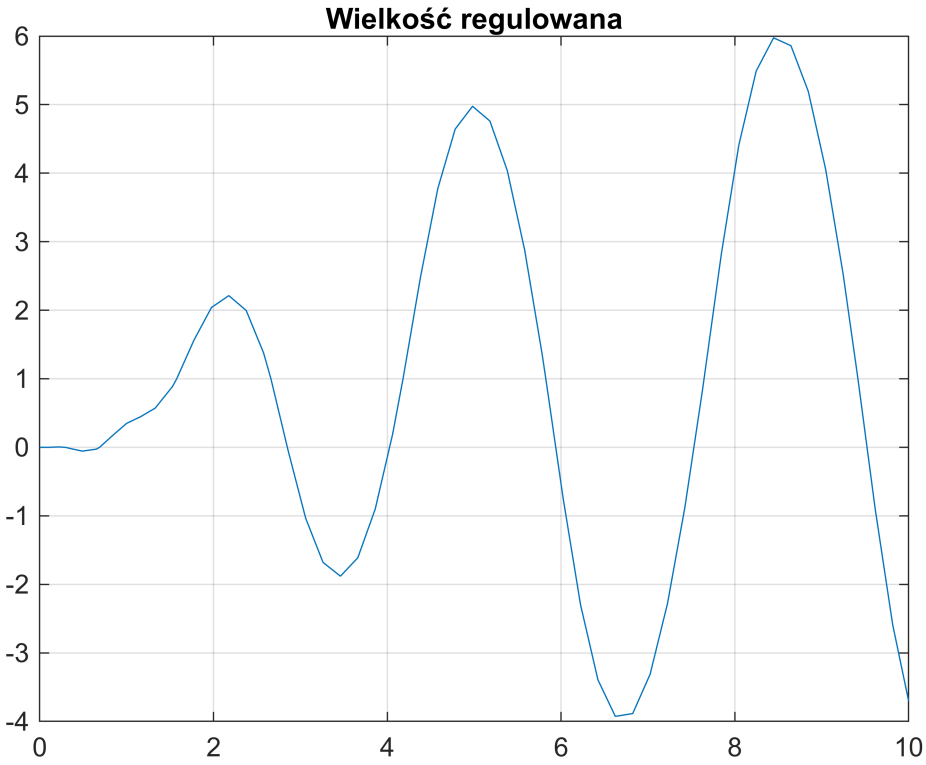

```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.0;
ym = 33;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```

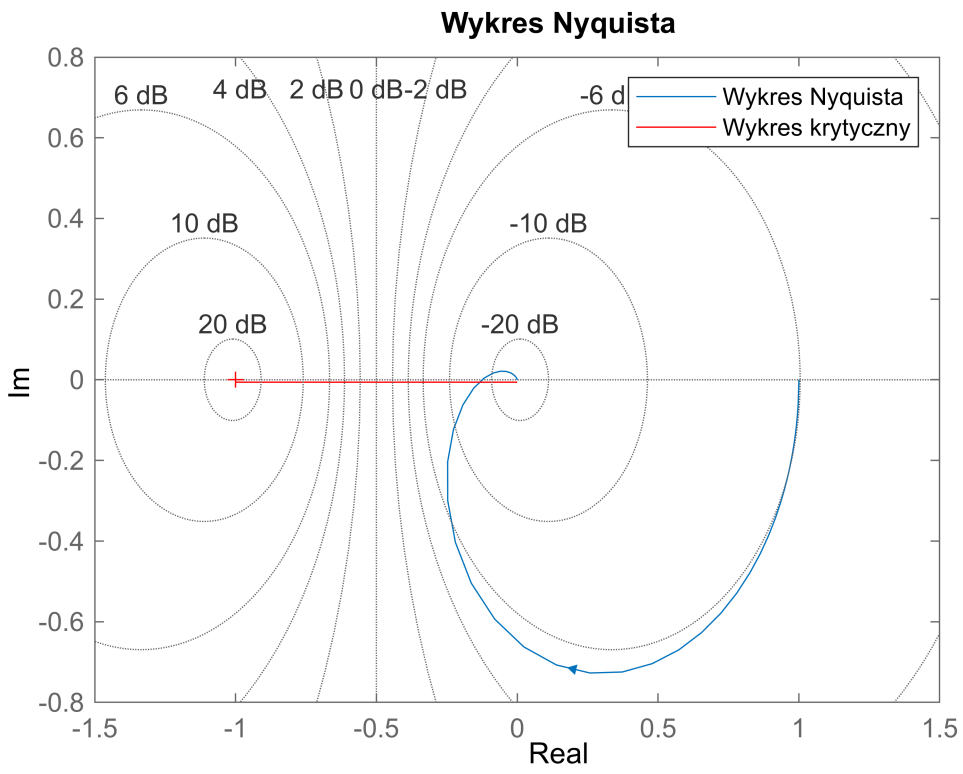



```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```

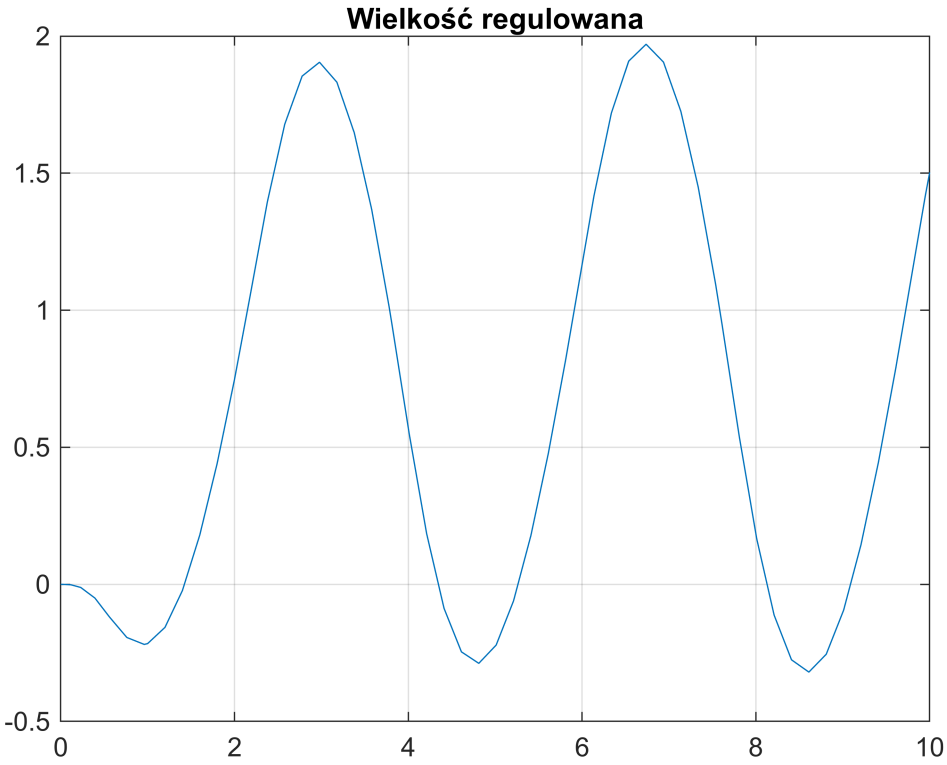


h = 0.05

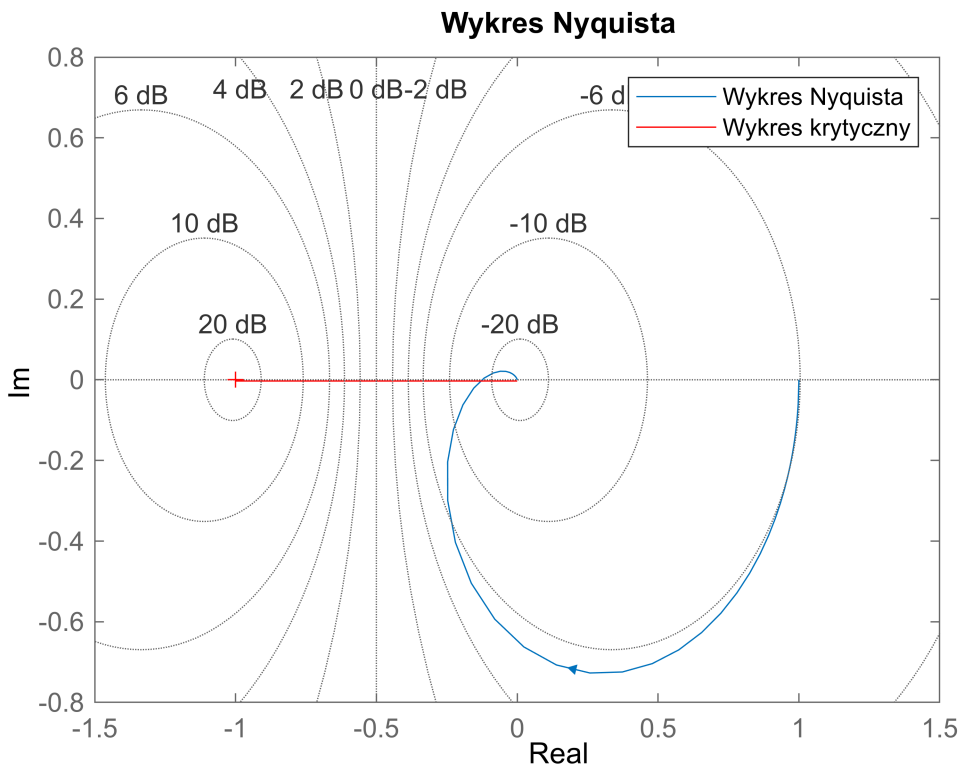
```
h = 0.05;
ym = 2.1*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



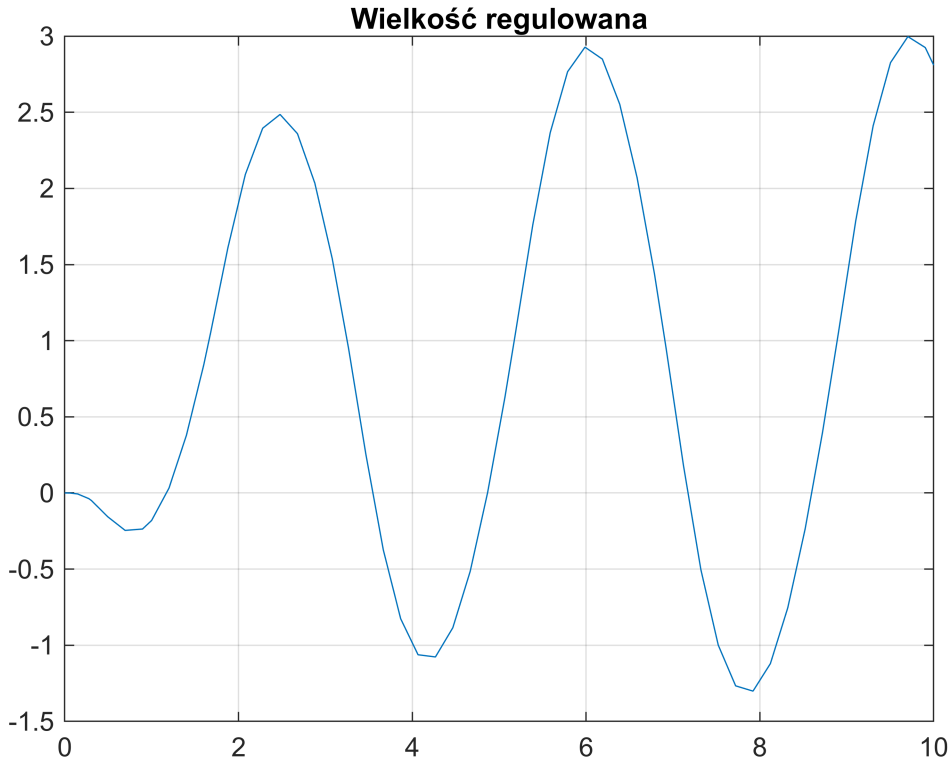

```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```



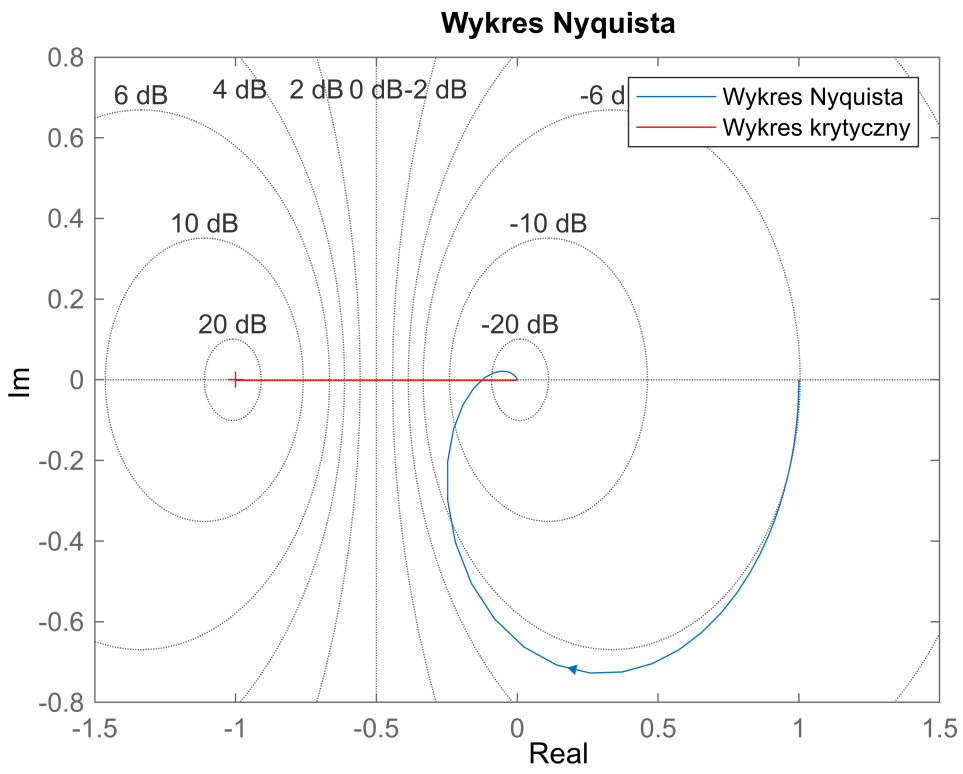

```
h = 0.05;
ym = 4.1*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



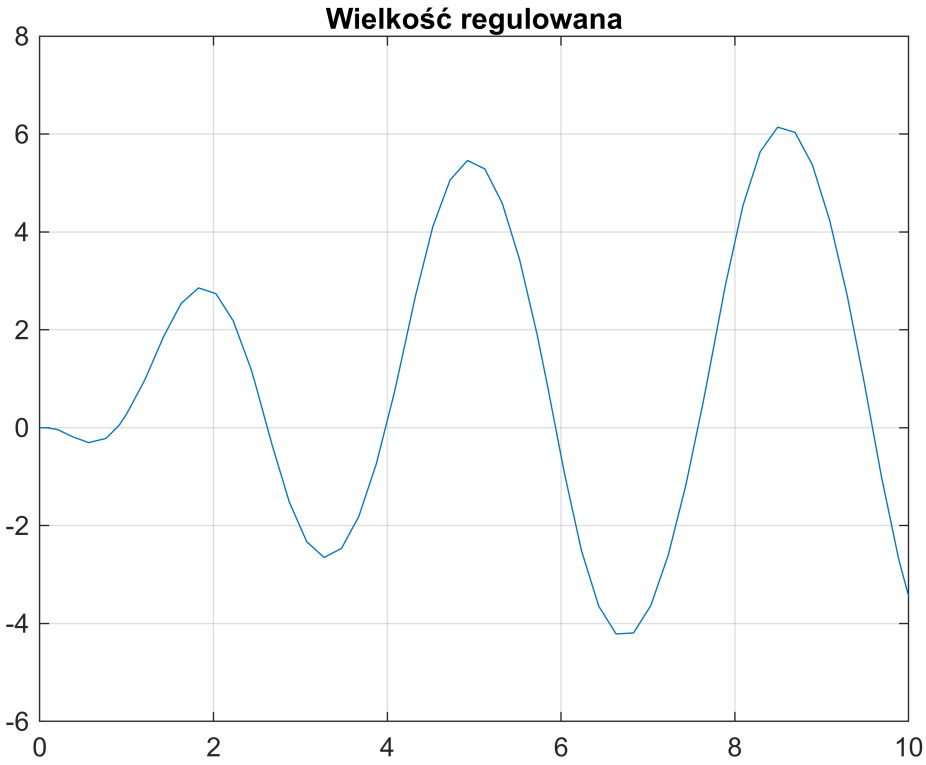

```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.05;
ym = 33;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```

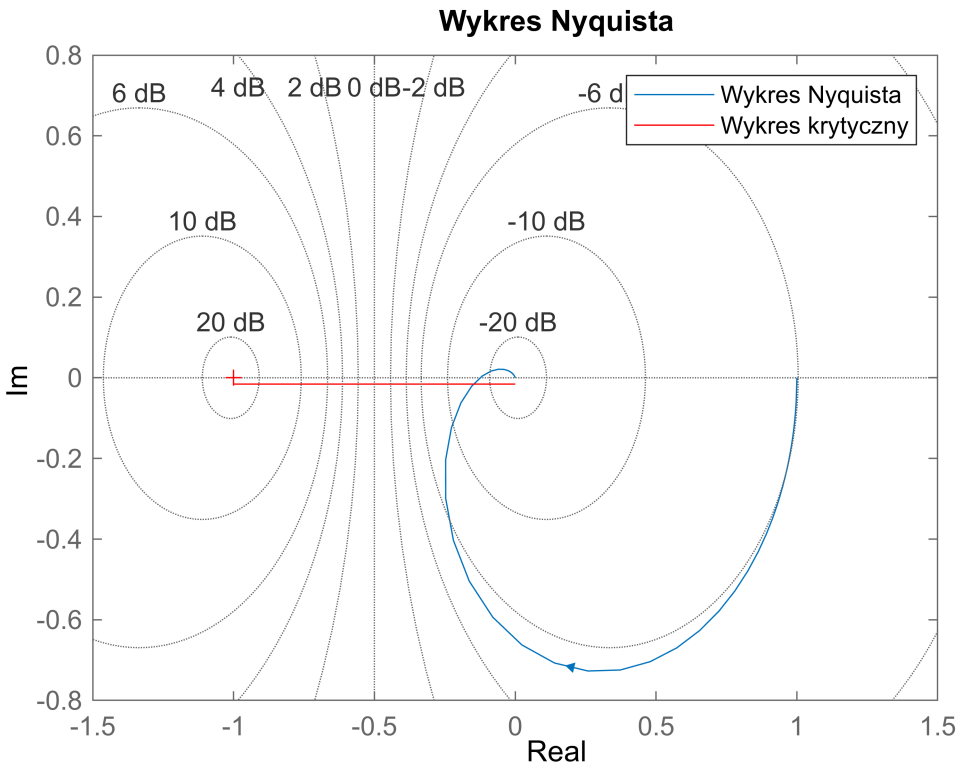



```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```

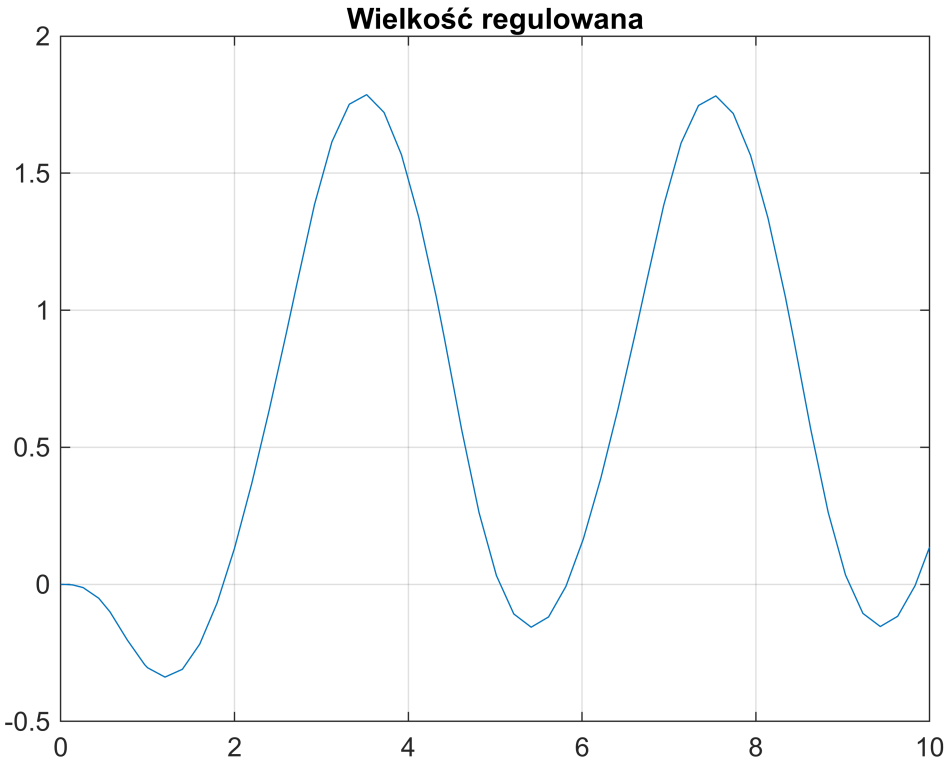


h = 0.1

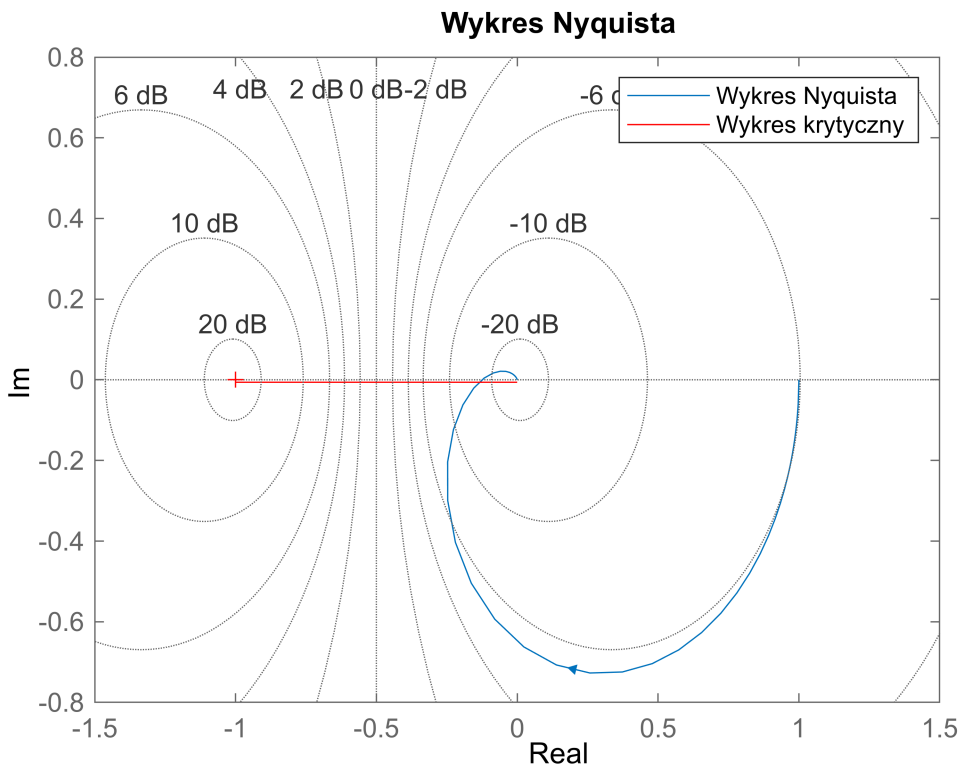
```
h = 0.1;
ym = 1.6*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



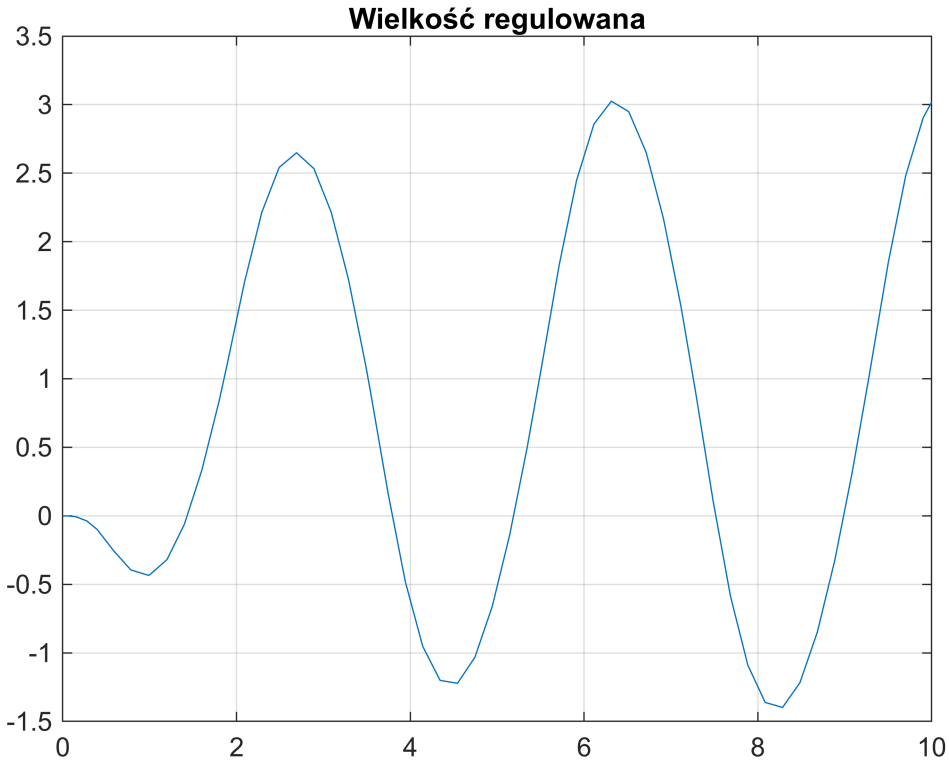

```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```



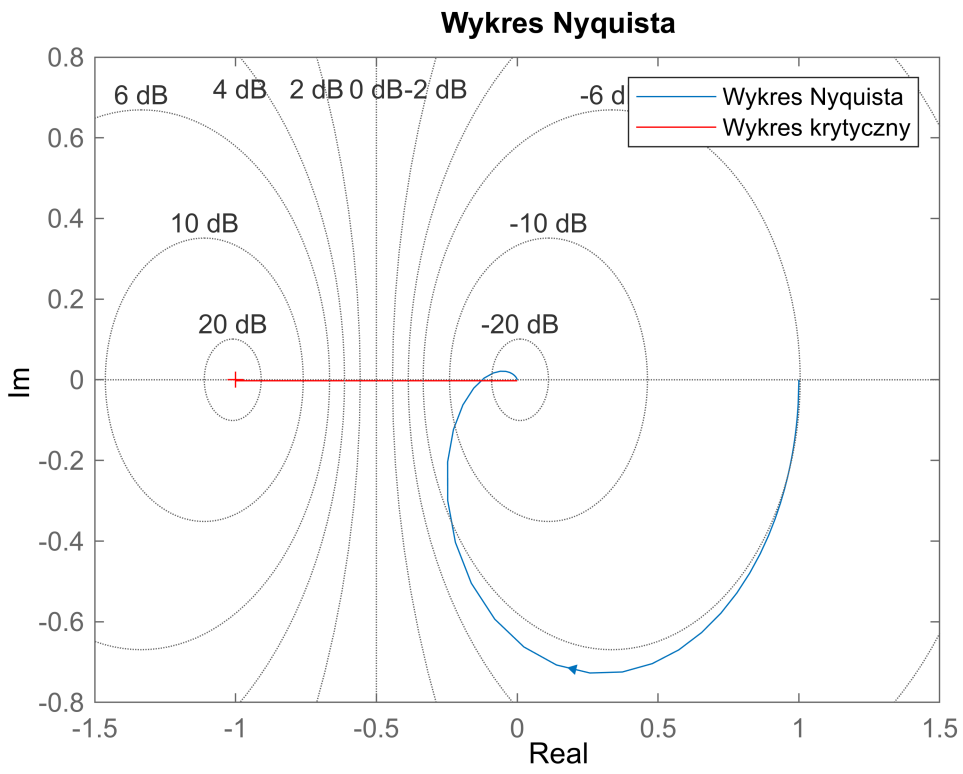

```
h = 0.1;
ym = 4.1*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



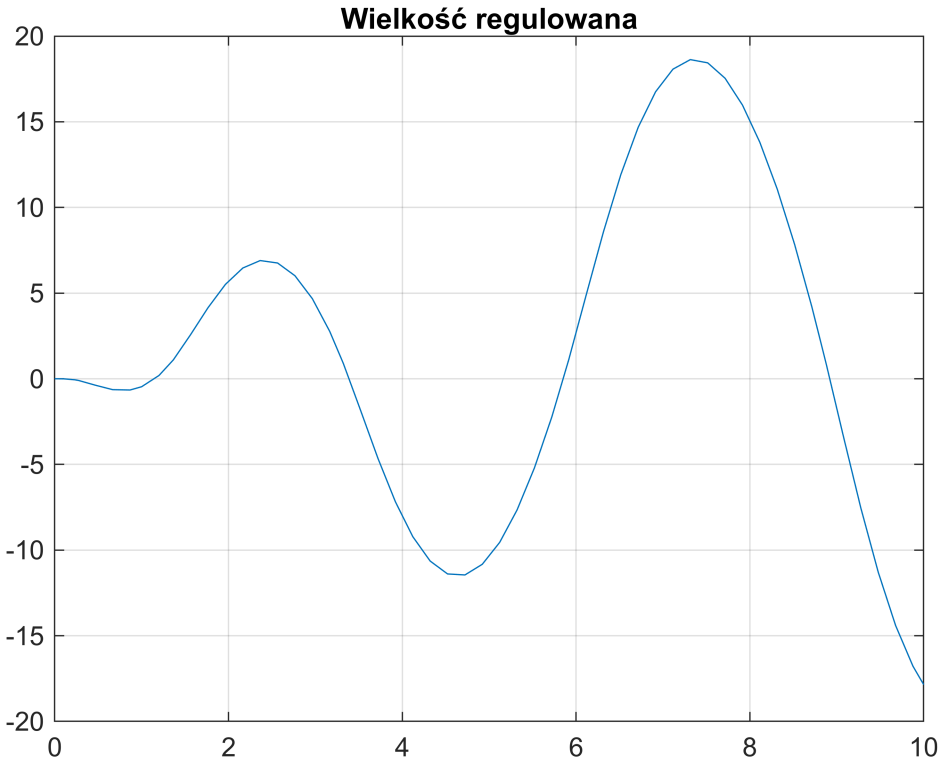

```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.1;
ym = 33;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf([licz], [mian1]));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```




```
out1 = sim("ukl_reg_II.slx");
figure;
plot(out1.y.time, out1.y.signals.values)
grid on;
title('Wielkość regulowana');
```

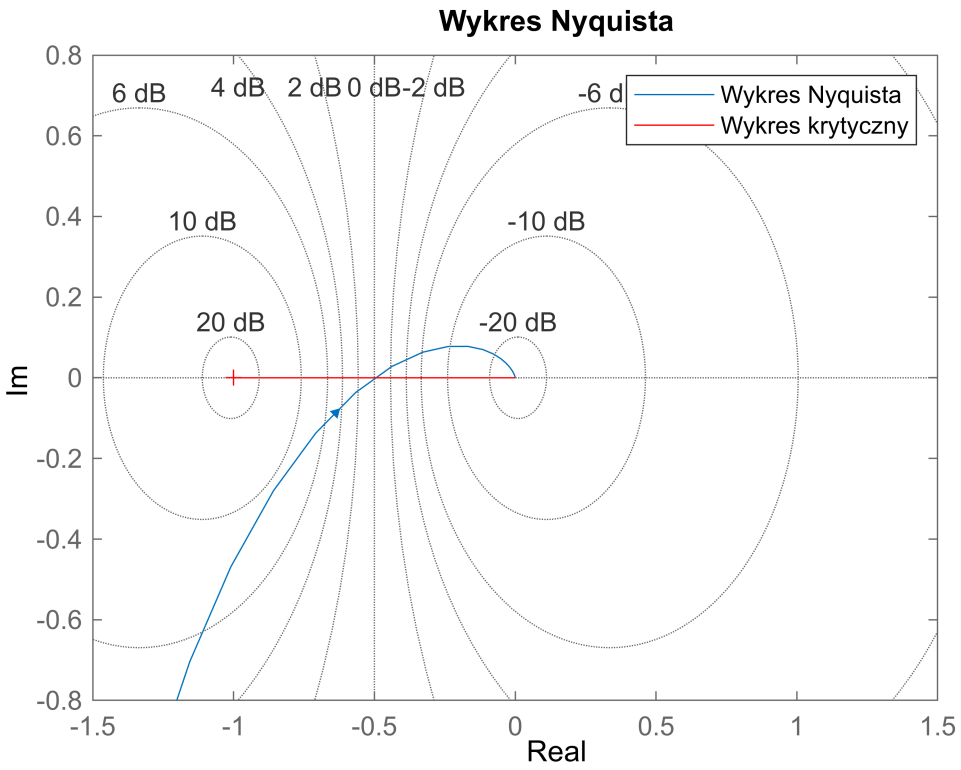


2.2

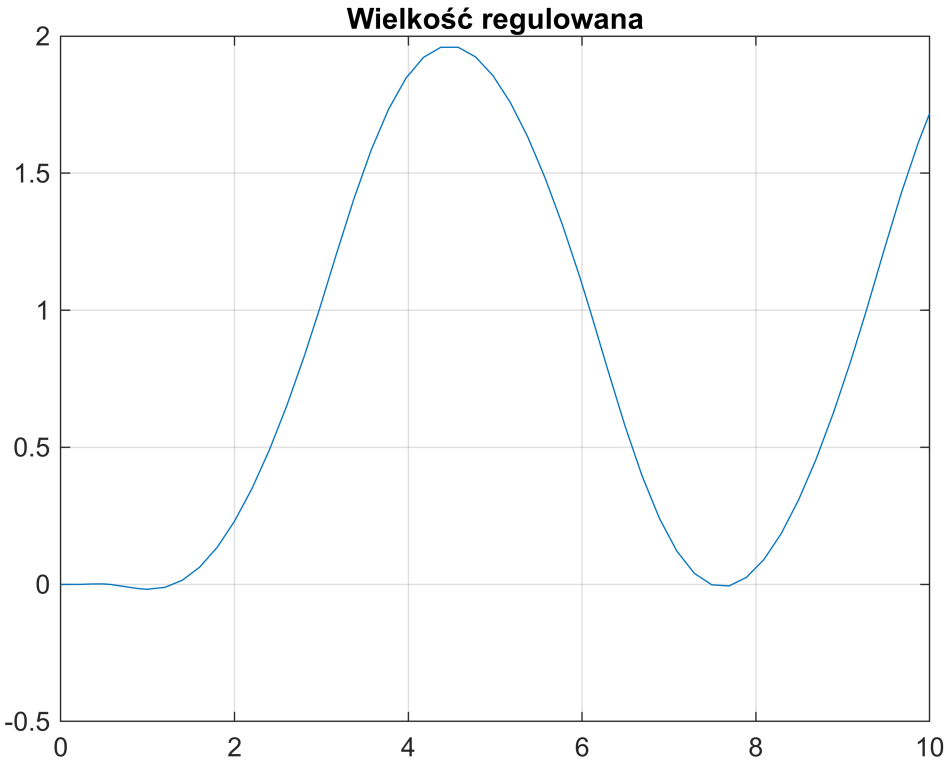
$$G(s) = \frac{1}{s(s^2 + 2s^2 + 1)}.$$

h = 0.0

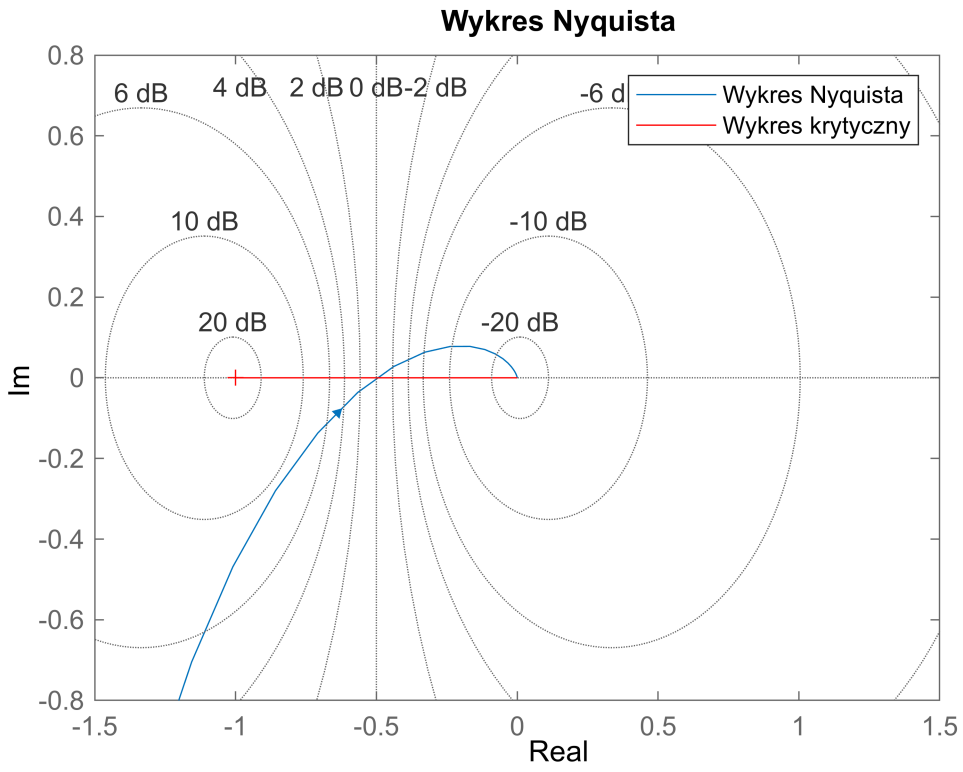
```
h = 0.0;
ym = 0.5*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



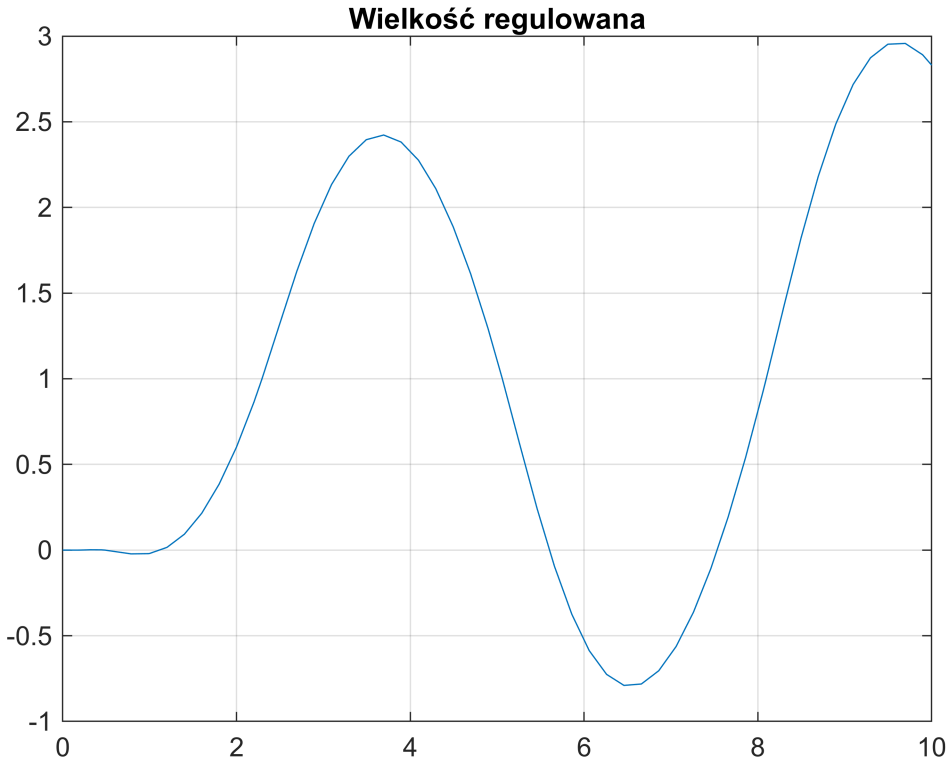

```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```



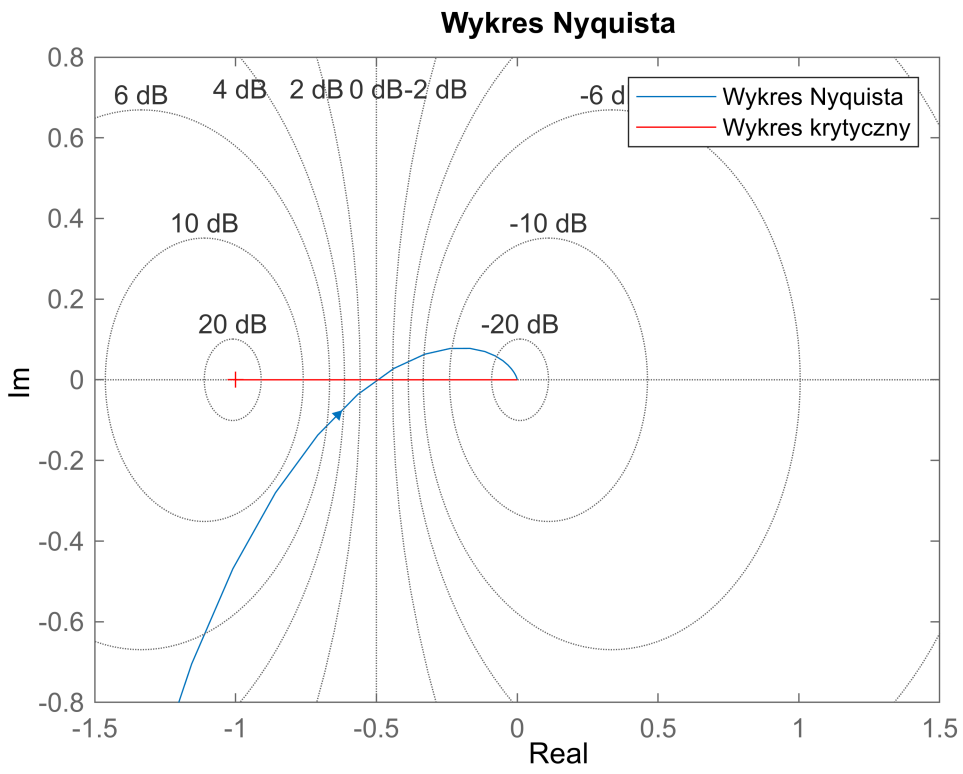

```
h = 0.0;
ym = 1*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



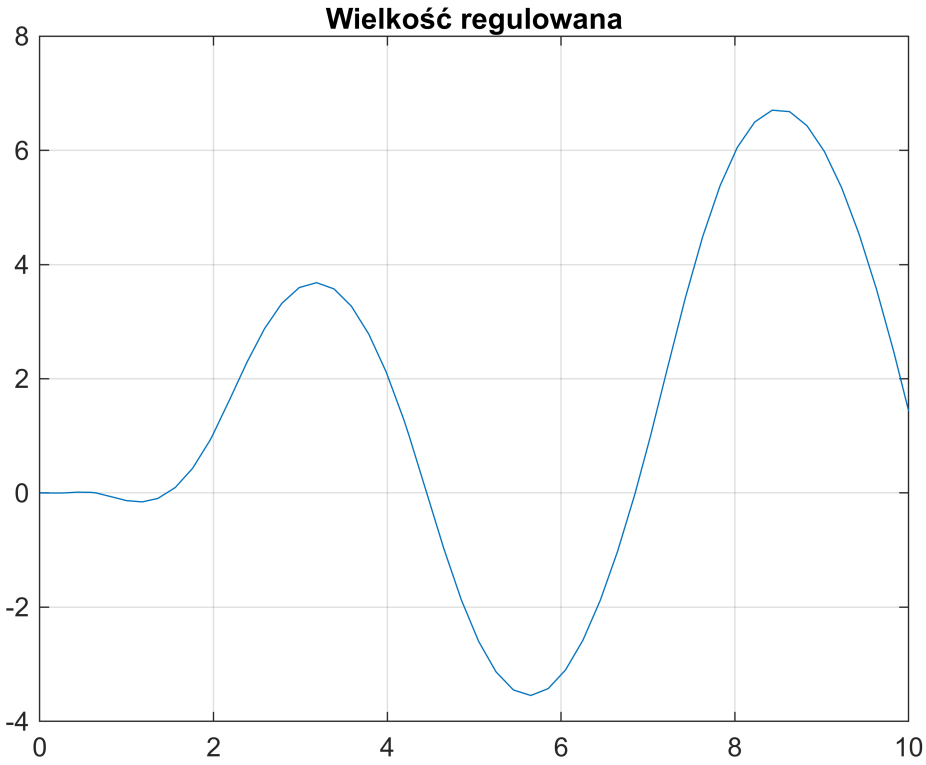

```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.0;
ym = 10;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```

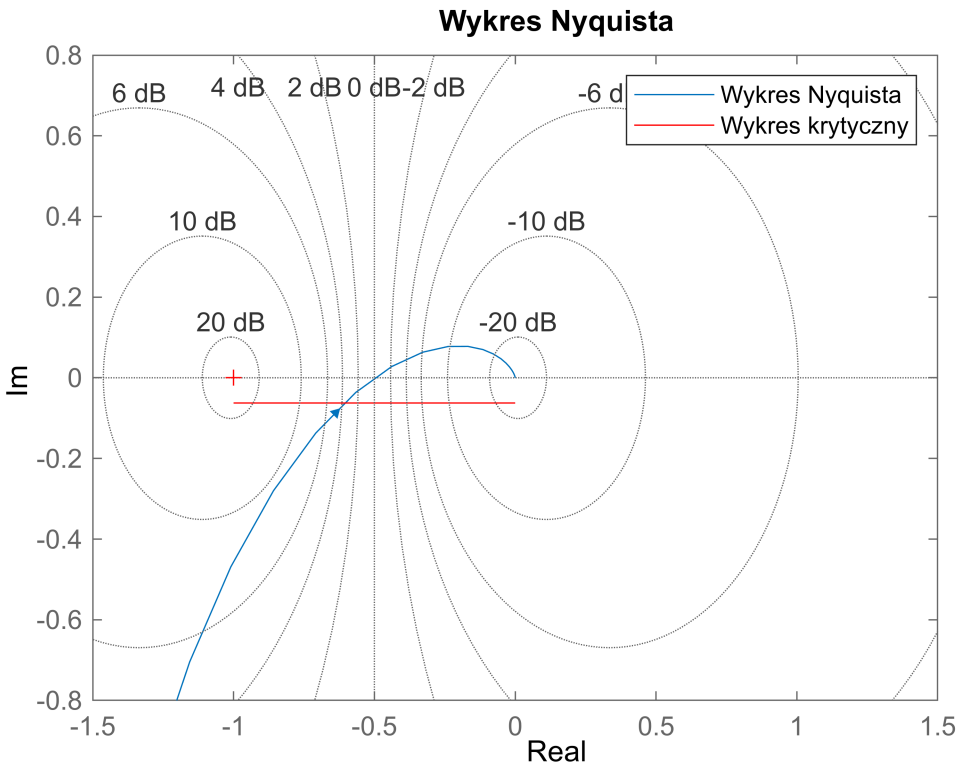



```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```

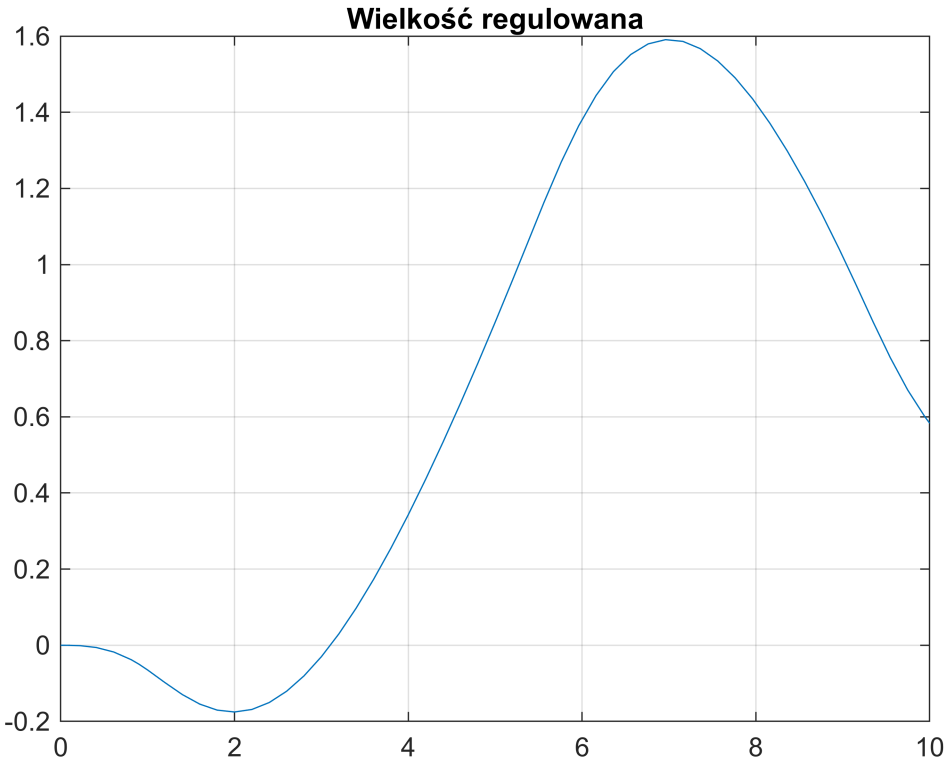


h = 0.05

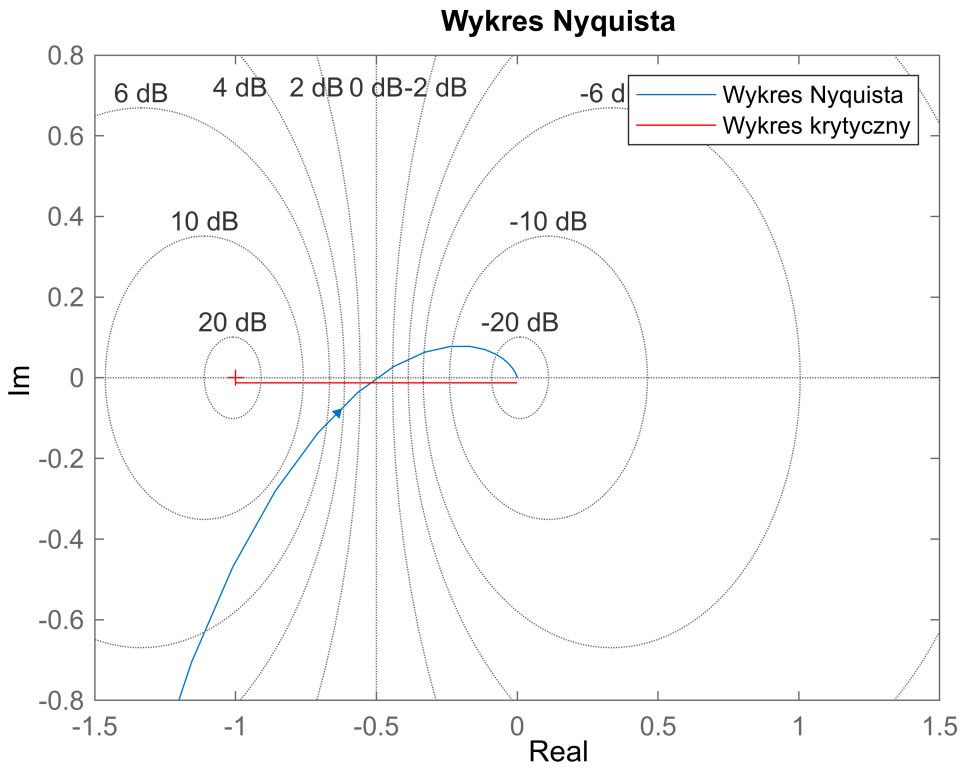
```
h = 0.05;
ym = 0.2*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



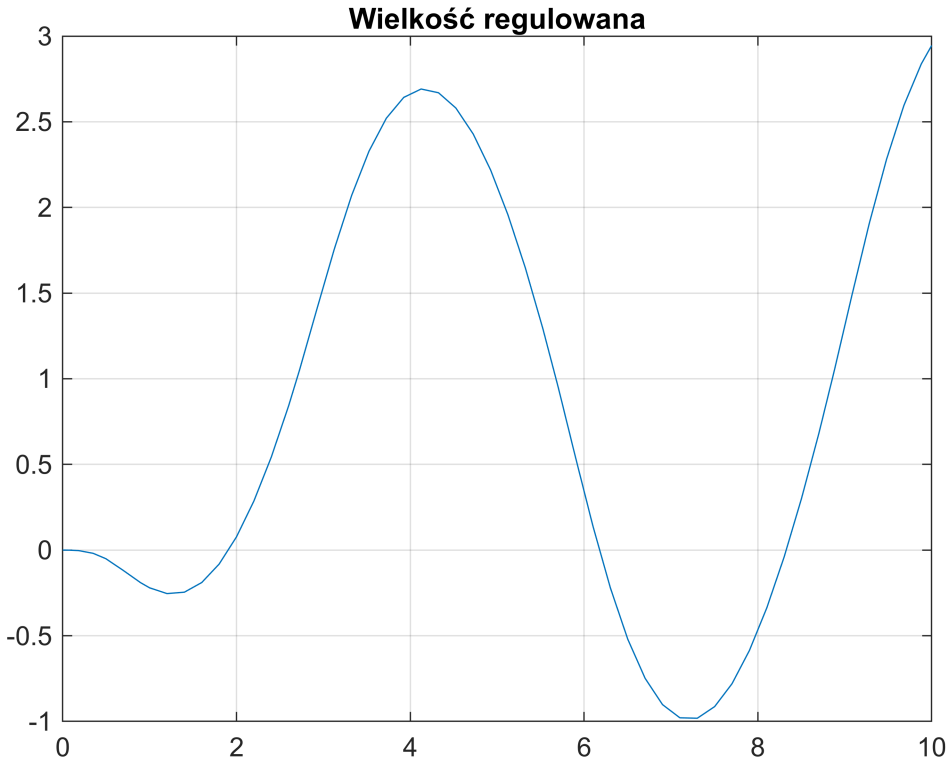

```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```



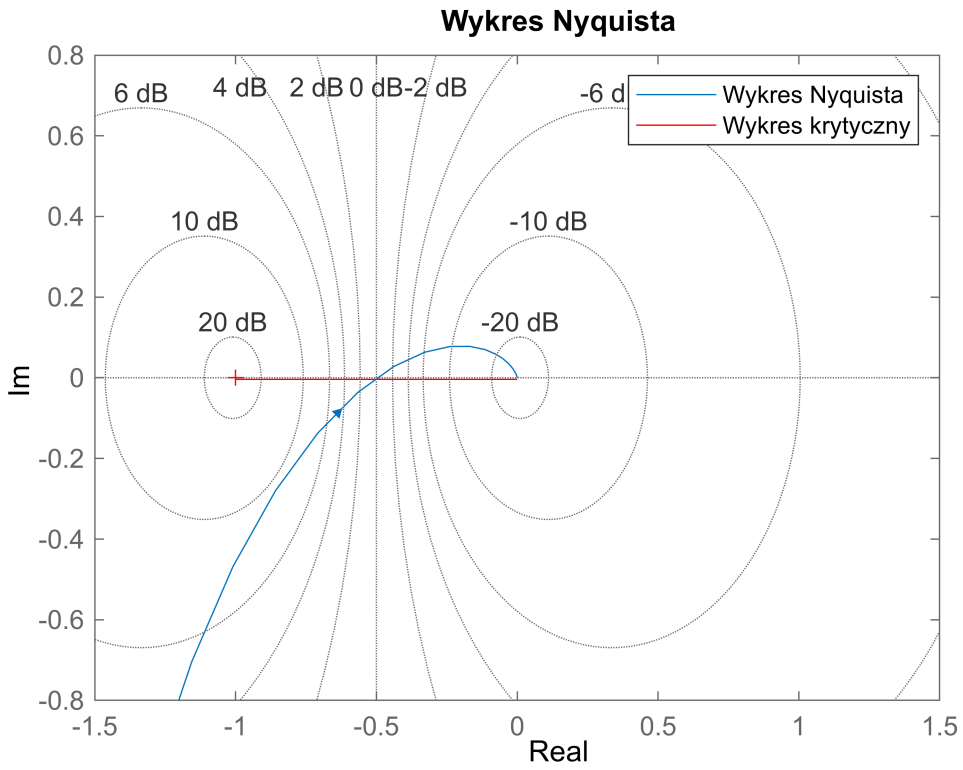

```
h = 0.05;
ym = pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



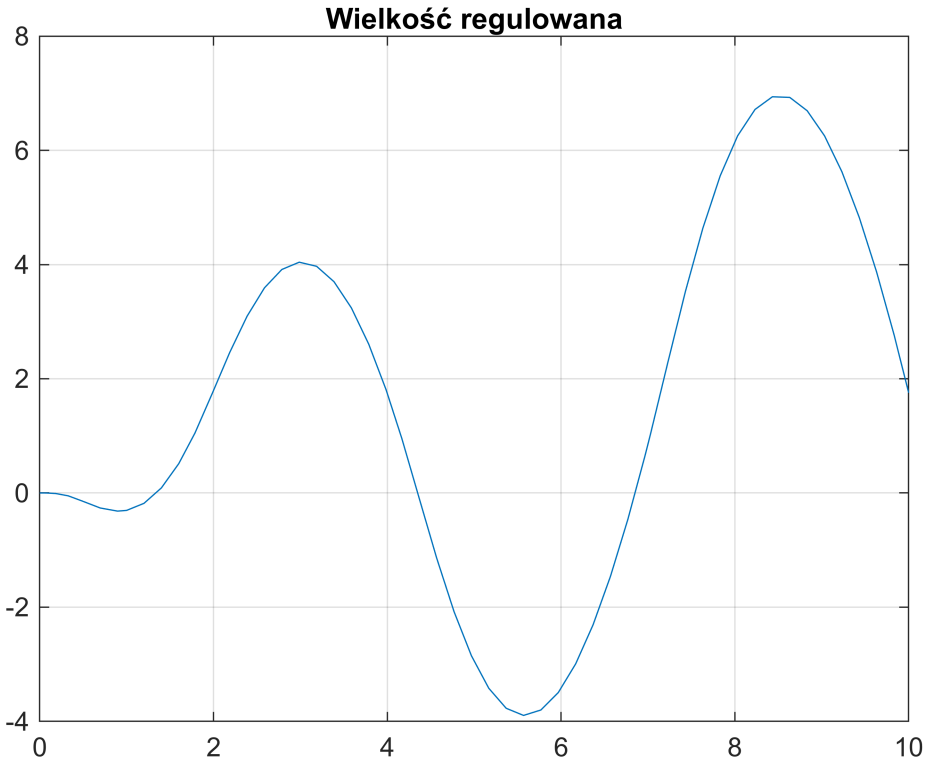

```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.05;
ym = 10;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```

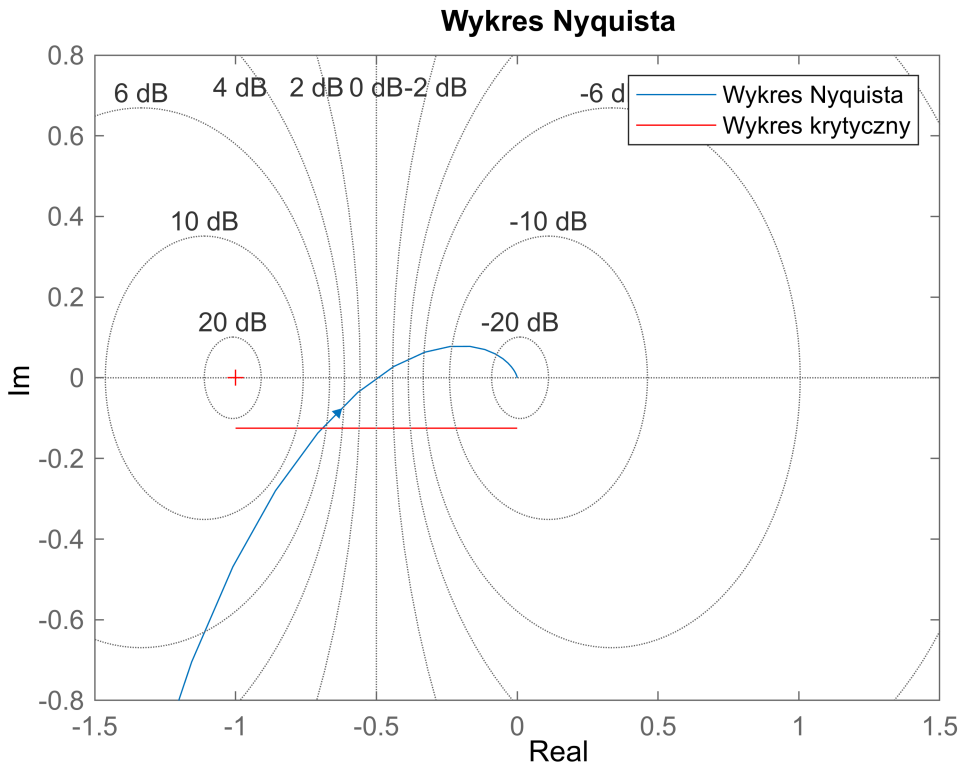



```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```

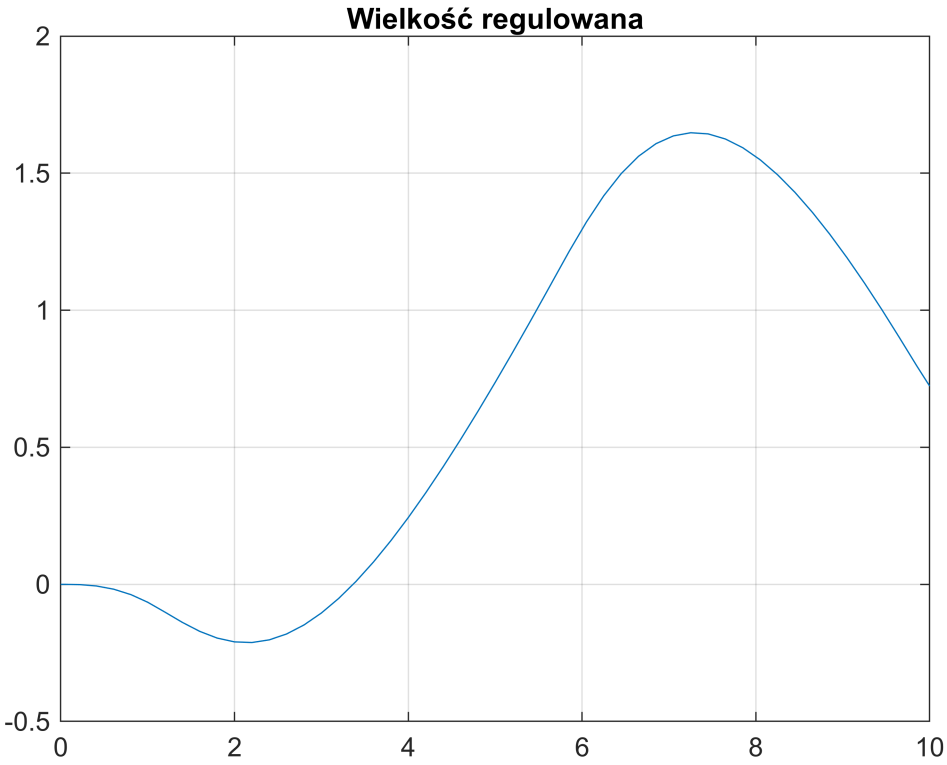


h = 0.1

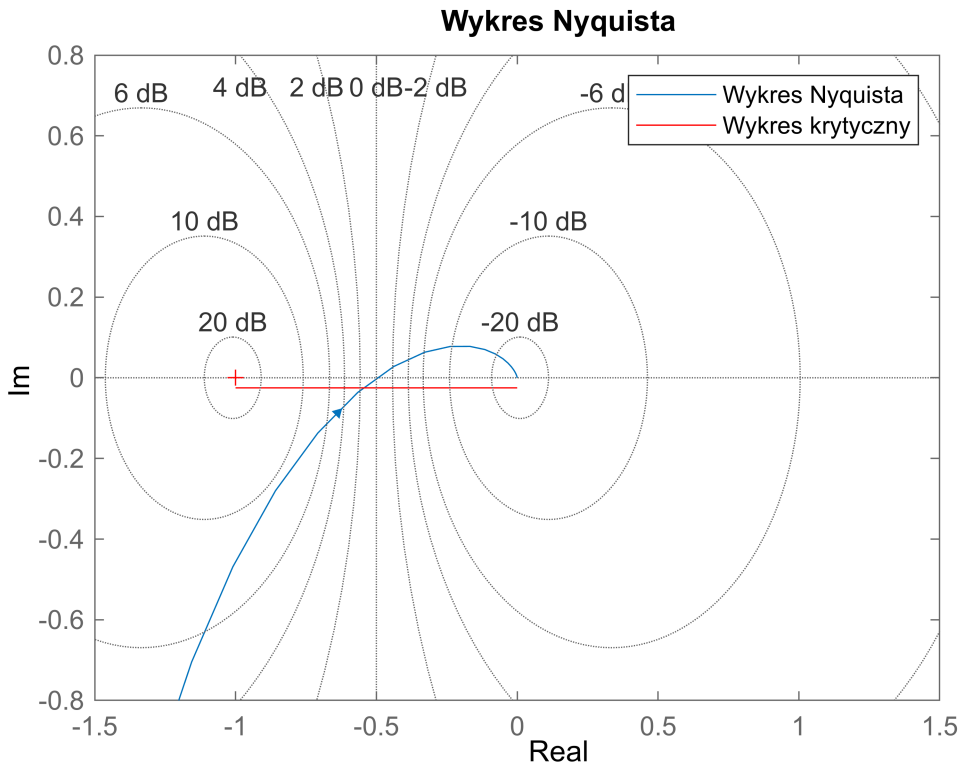
```
h = 0.1;
ym = 0.2*pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



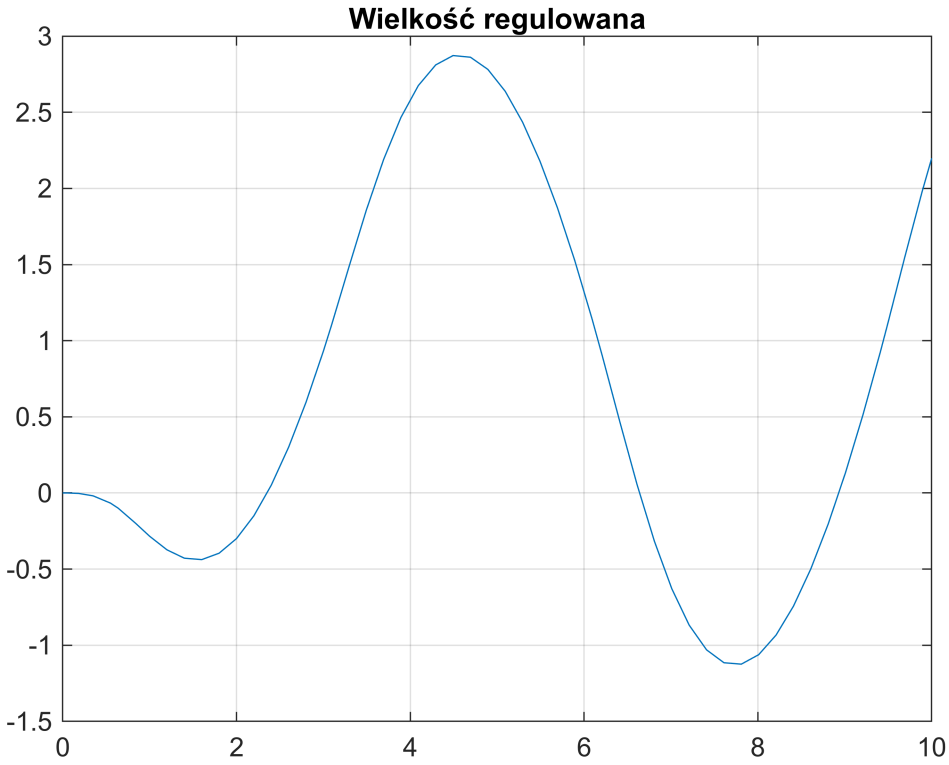

```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```



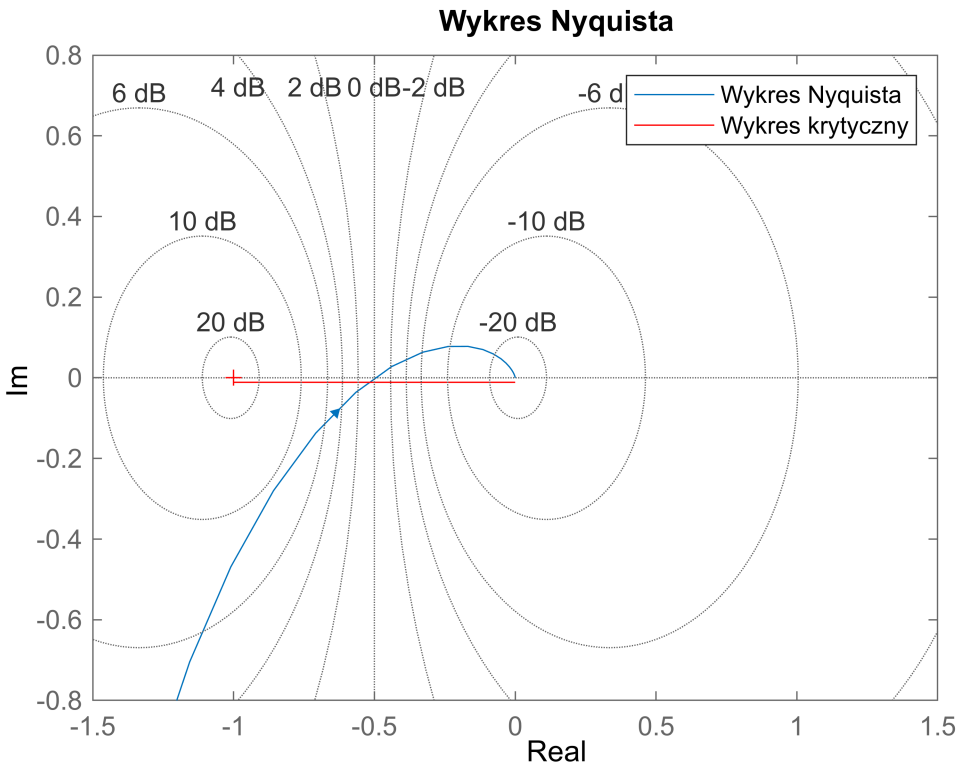

```
h = 0.1;
ym = pi;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



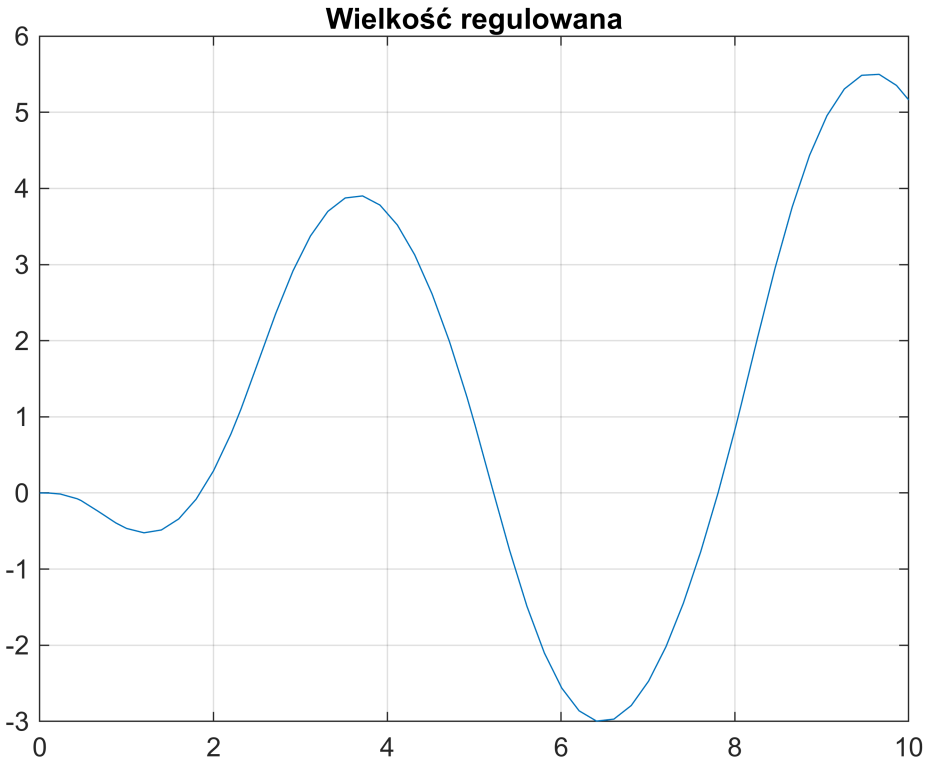

```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.1;
ym = 7;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian2)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1.5 1.5 -0.8 0.8]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```




```
out2 = sim("ukl_reg_II.slx");
figure;
plot(out2.y.time, out2.y.signals.values)
grid on;
title('Wielkość regulowana');
```

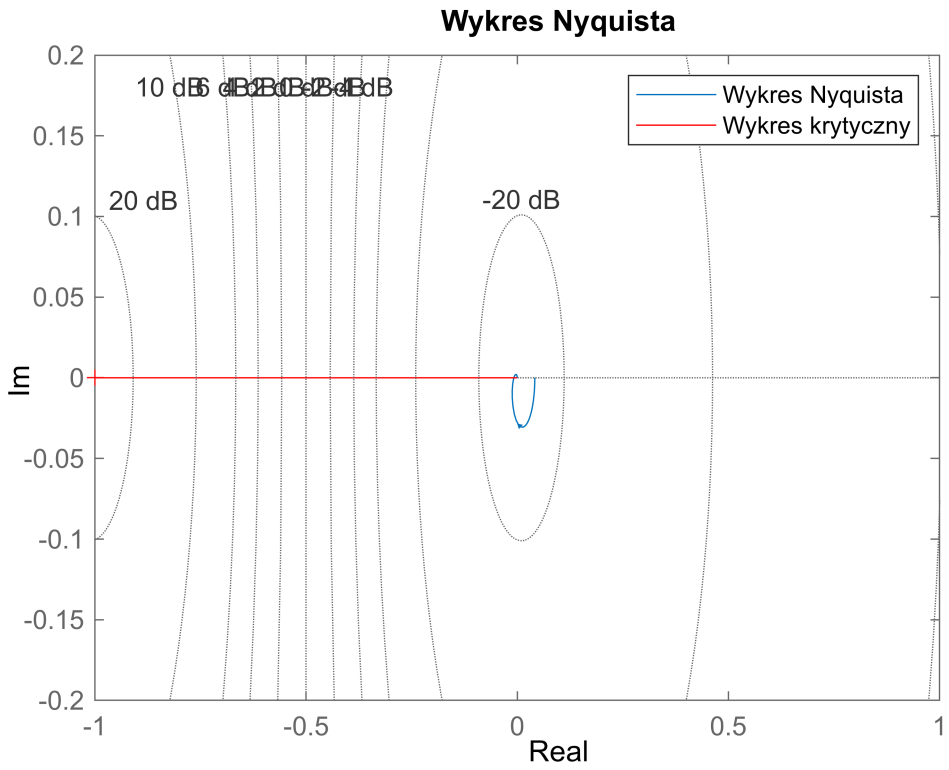


2.3

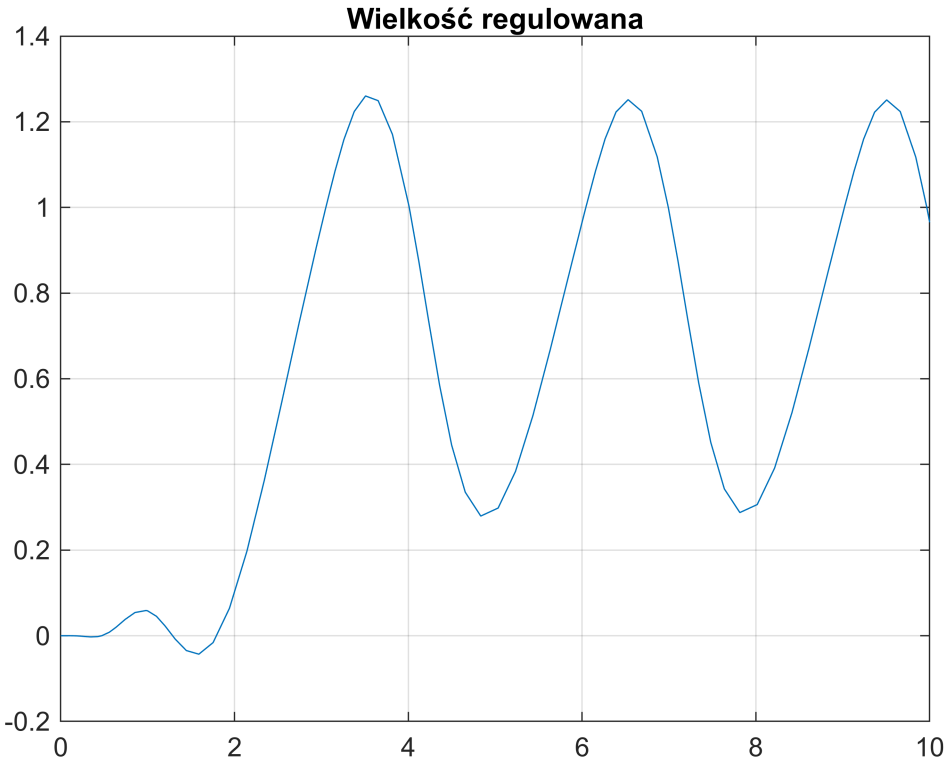
$$G(s) = \frac{1}{(s+1)(s+2)(s+3)(s+4)}.$$

h = 0.0

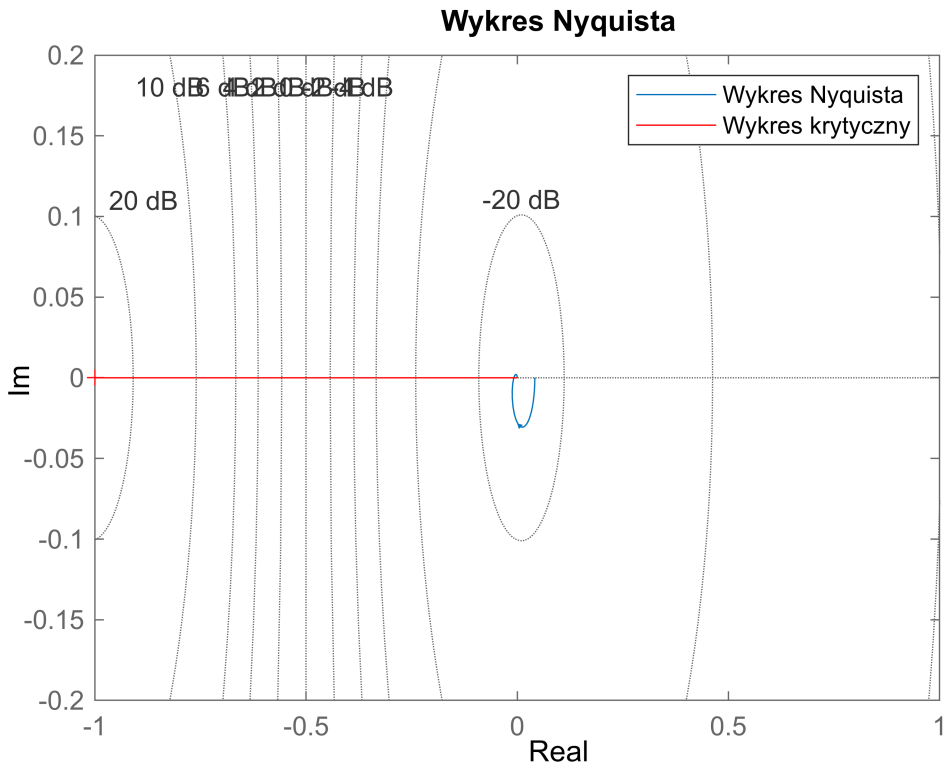
```
h = 0.0;
ym = 50;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



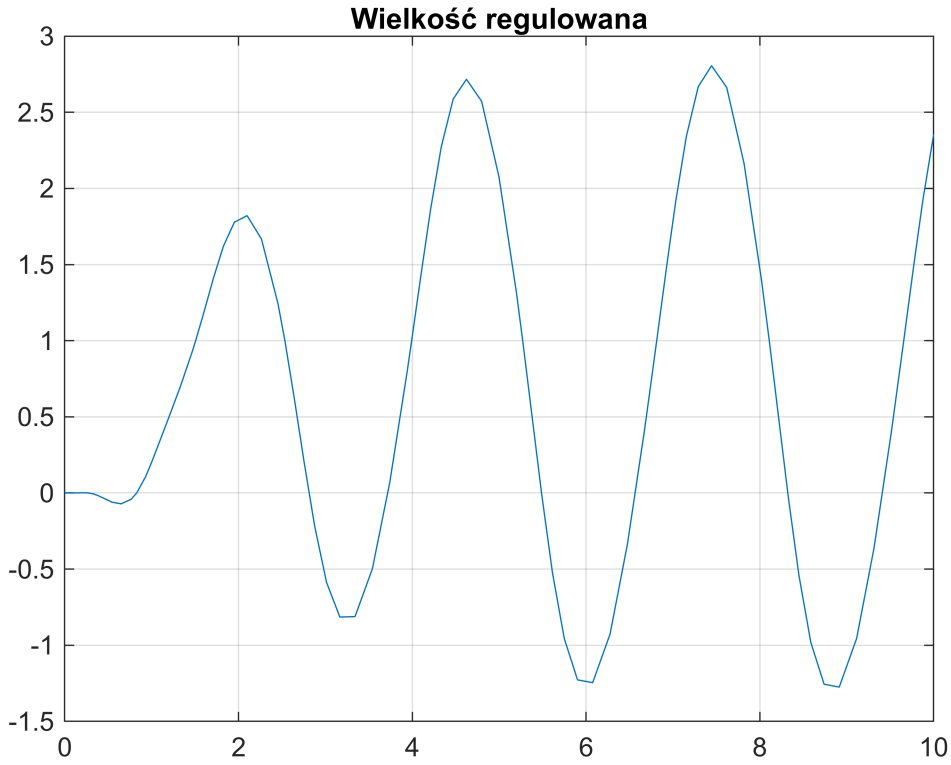

```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```



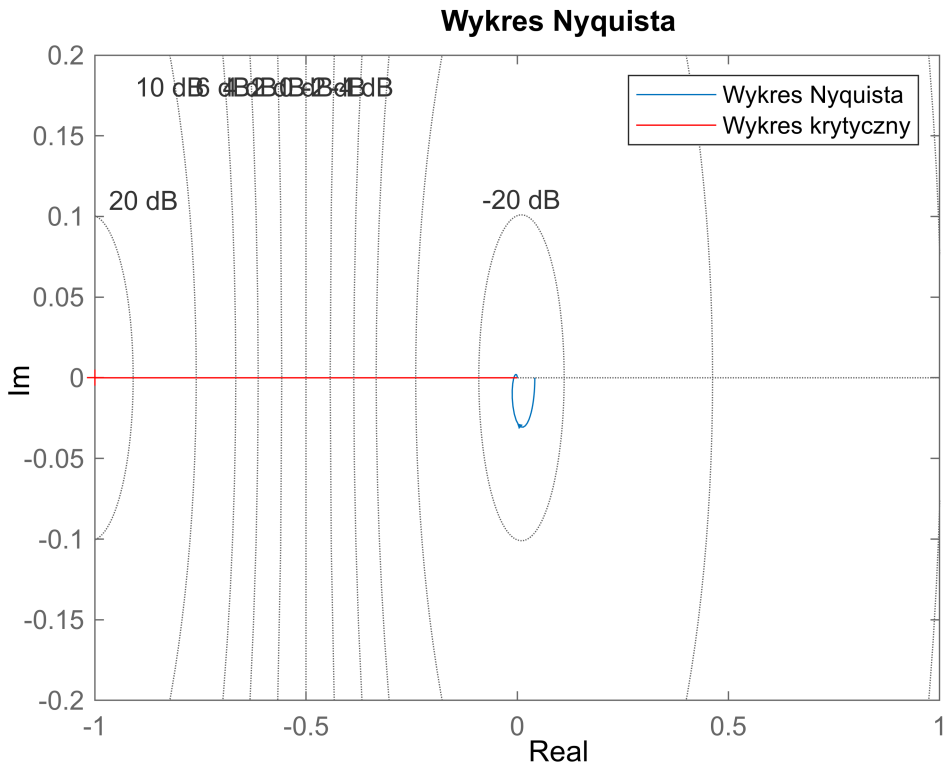

```
h = 0.0;
ym = 200;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



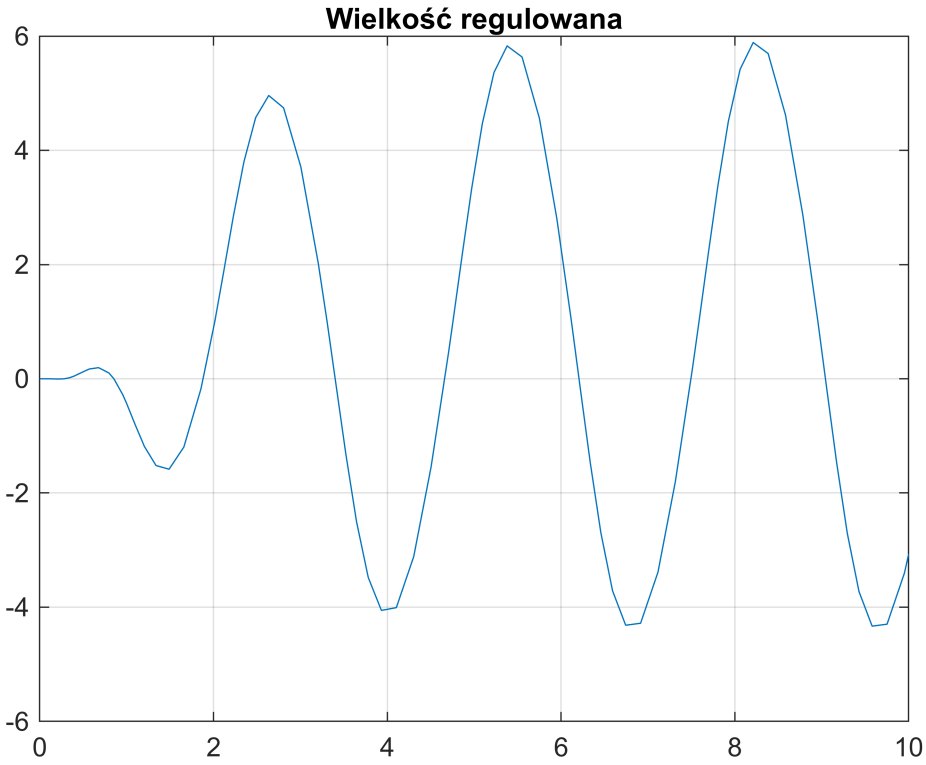

```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.0;
ym = 500;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```

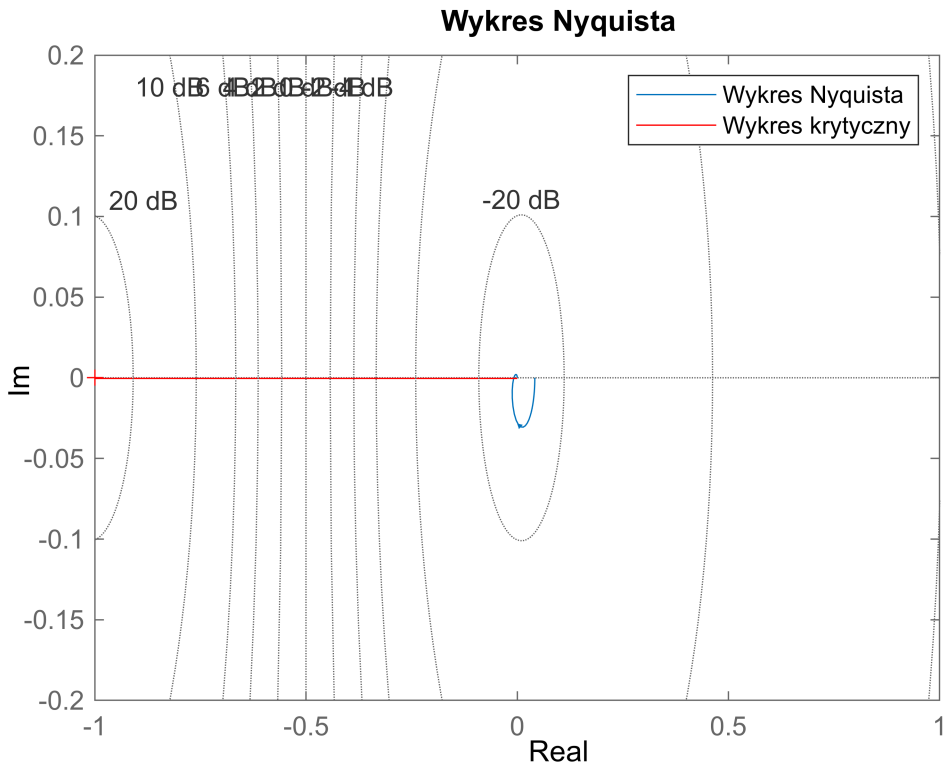



```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```

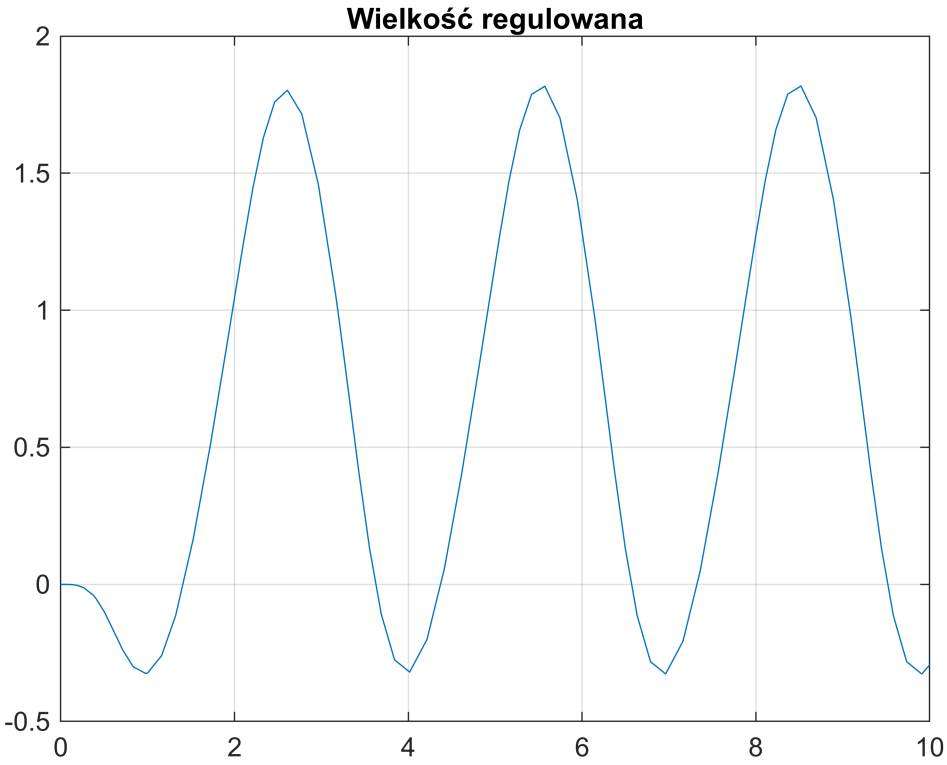


h = 0.05

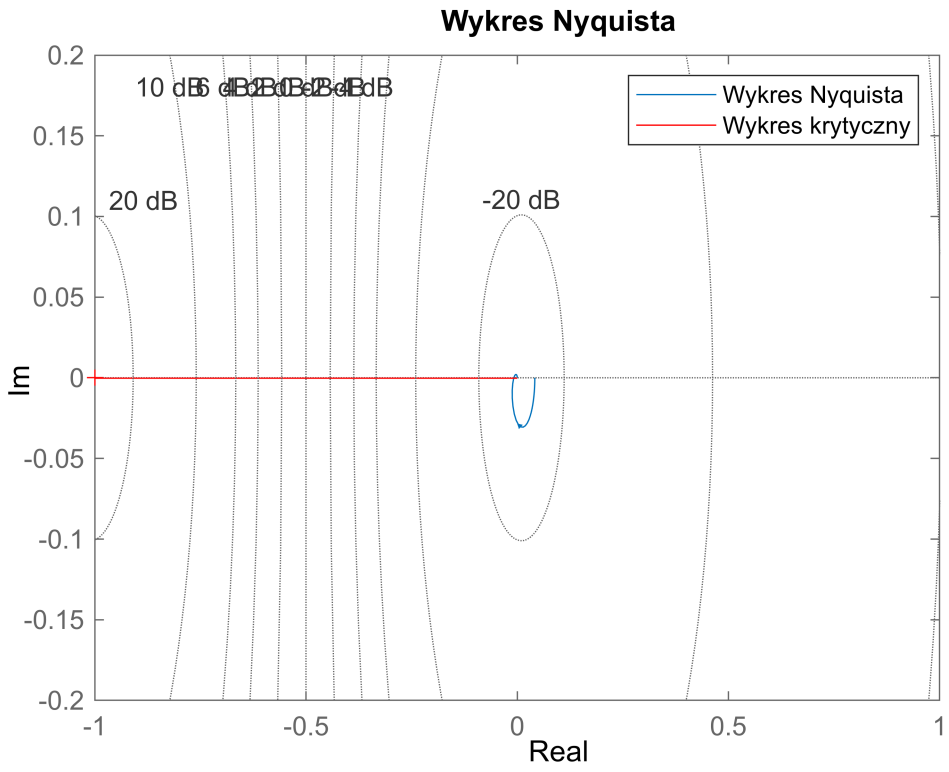
```
h = 0.05;
ym = 100;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



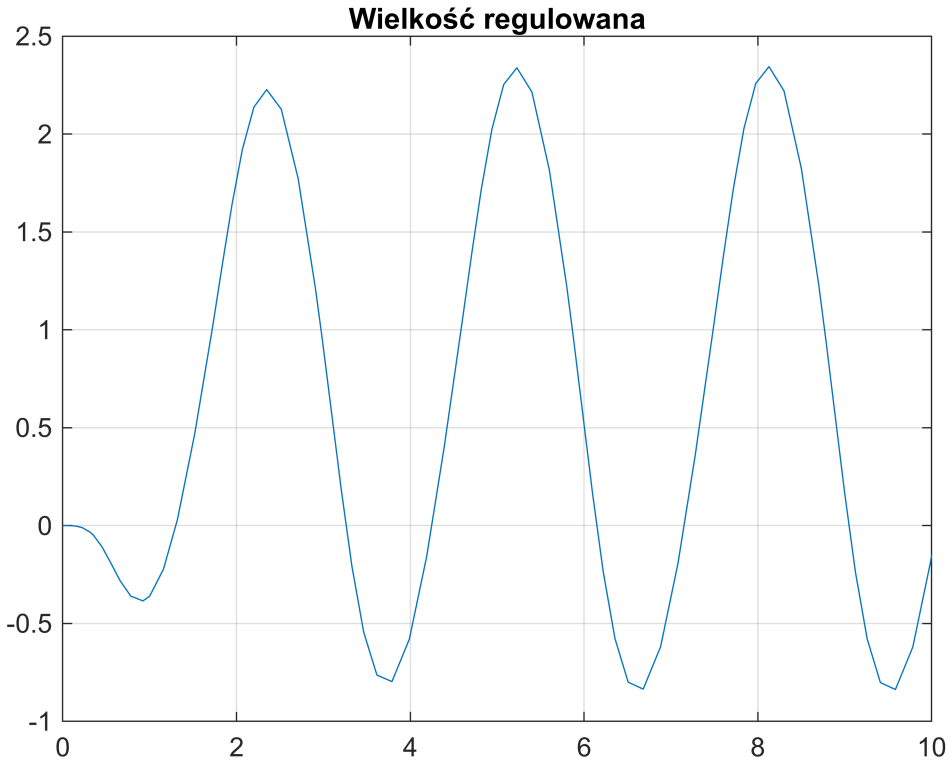

```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```



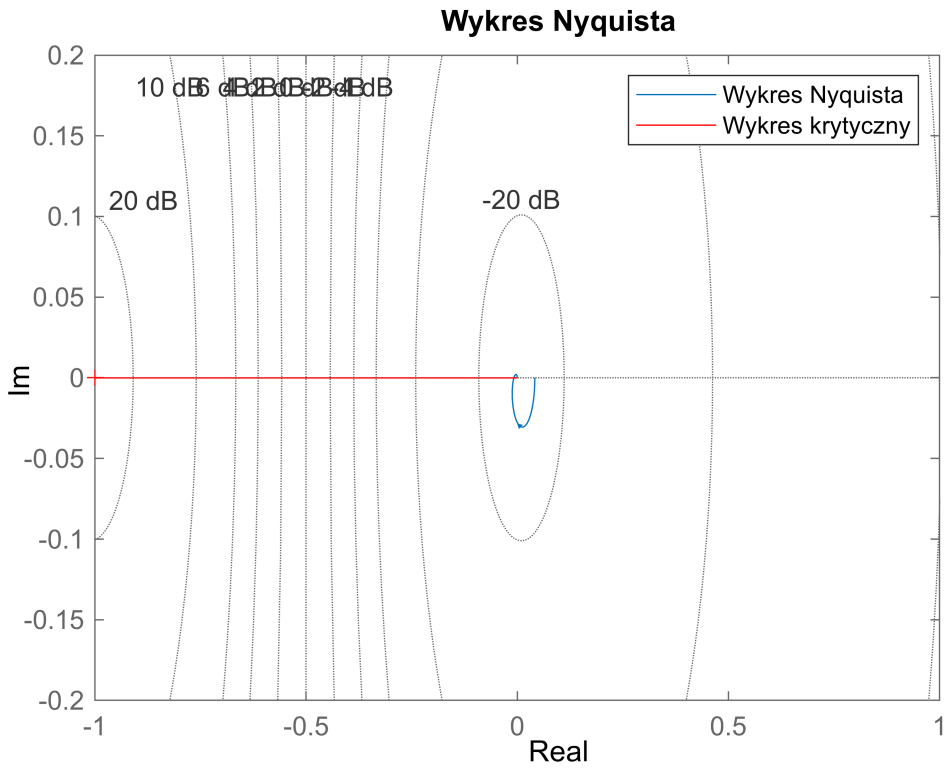

```
h = 0.05;
ym = 150;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



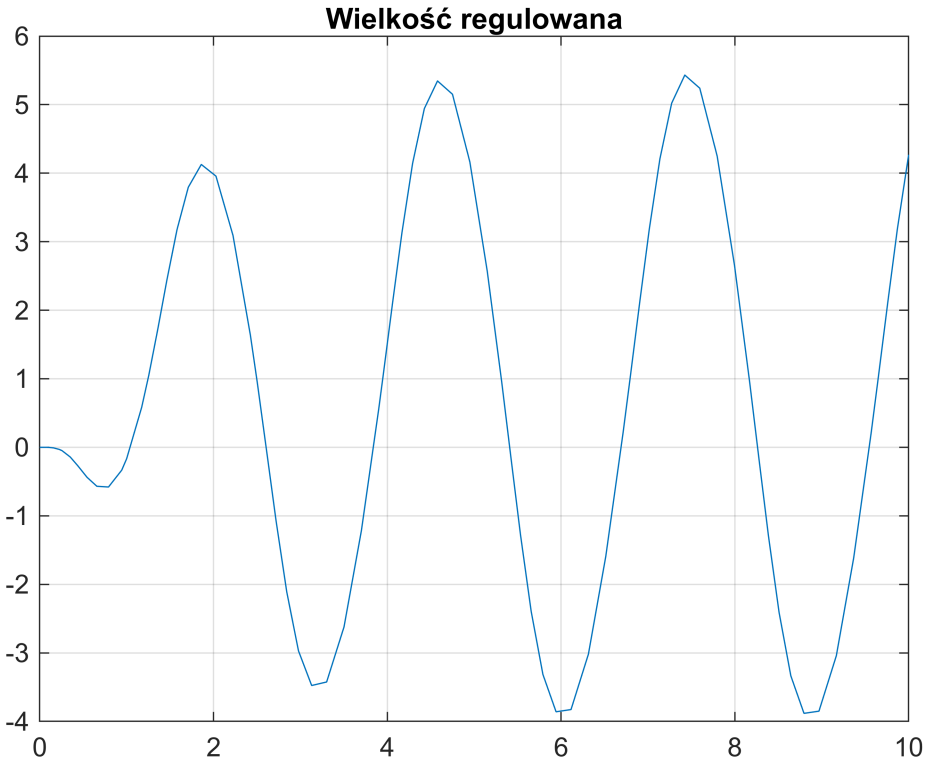

```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.05;
ym = 450;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```

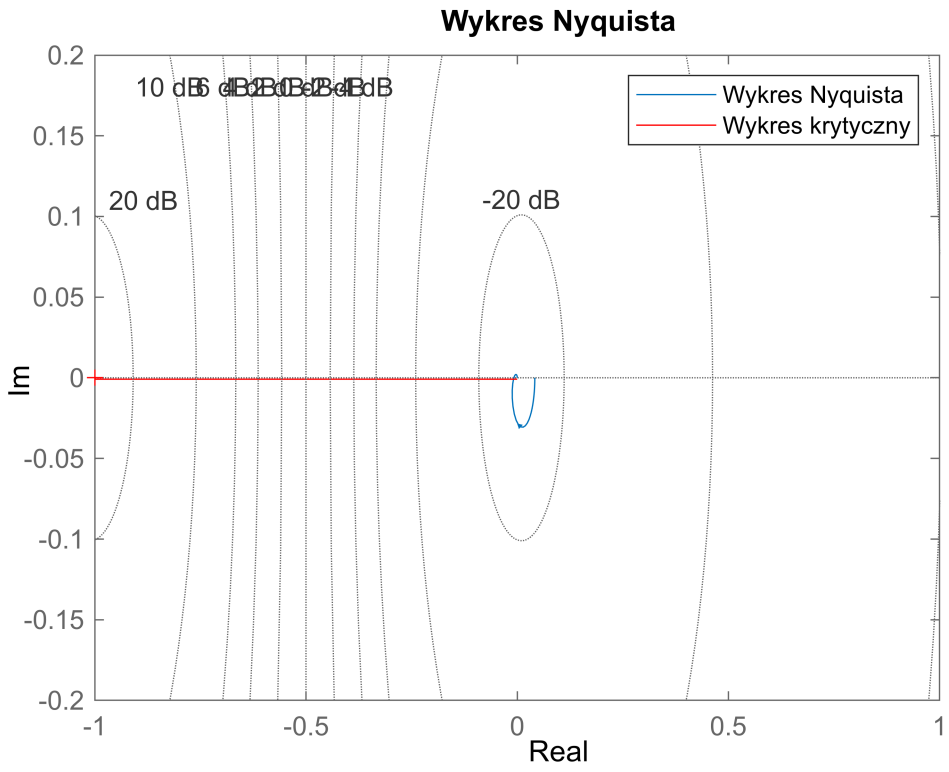



```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```

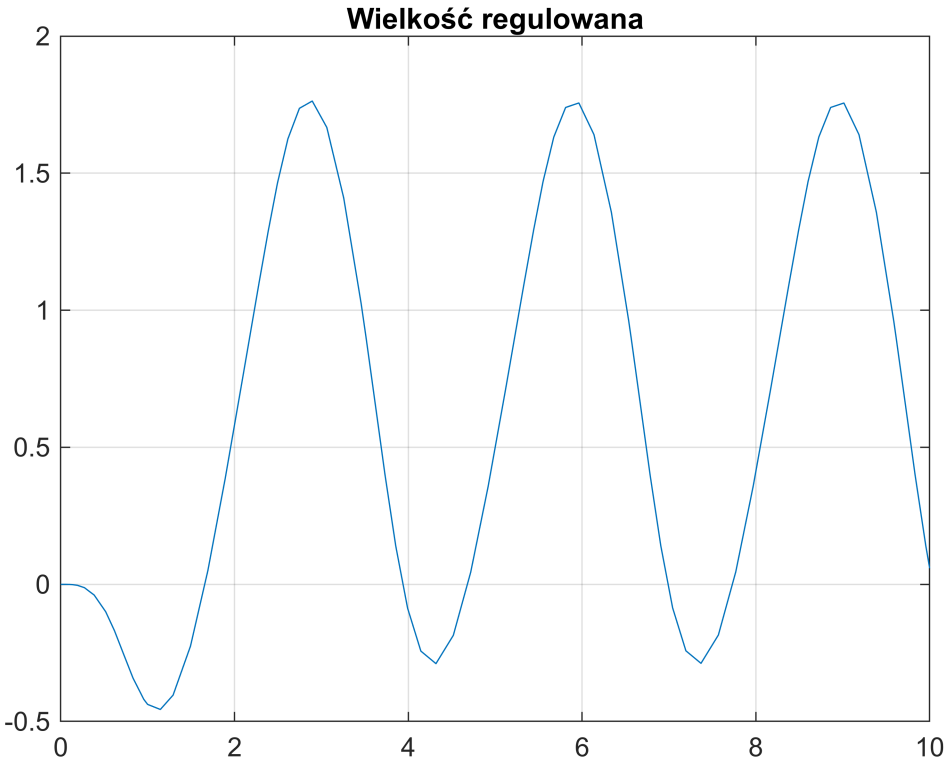


h = 0.1

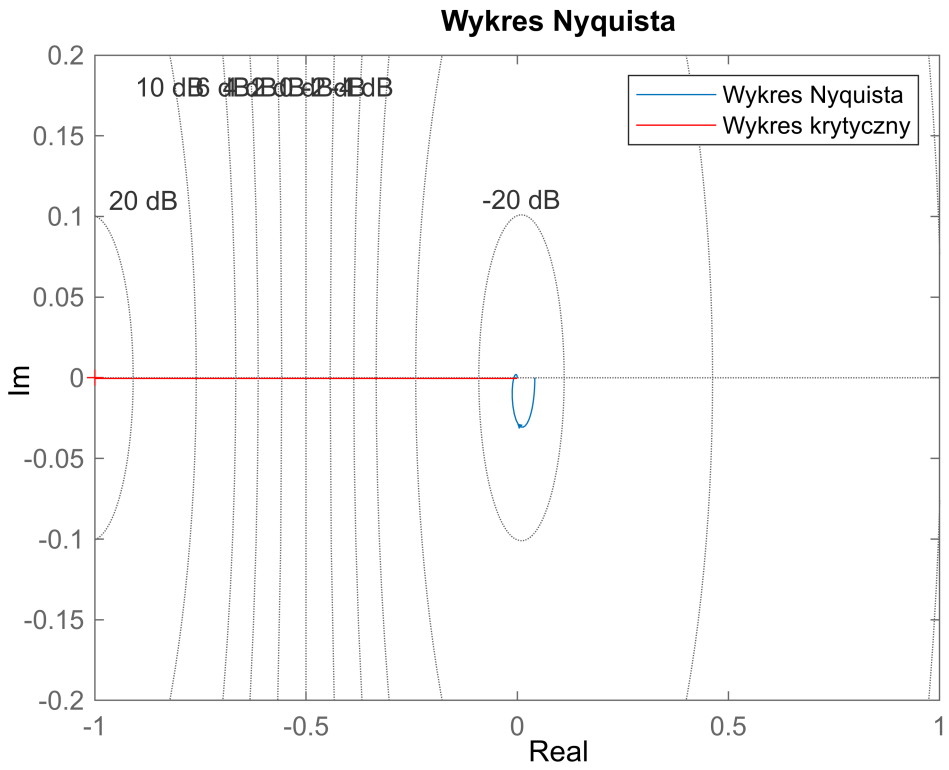
```
h = 0.1;
ym = 90;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



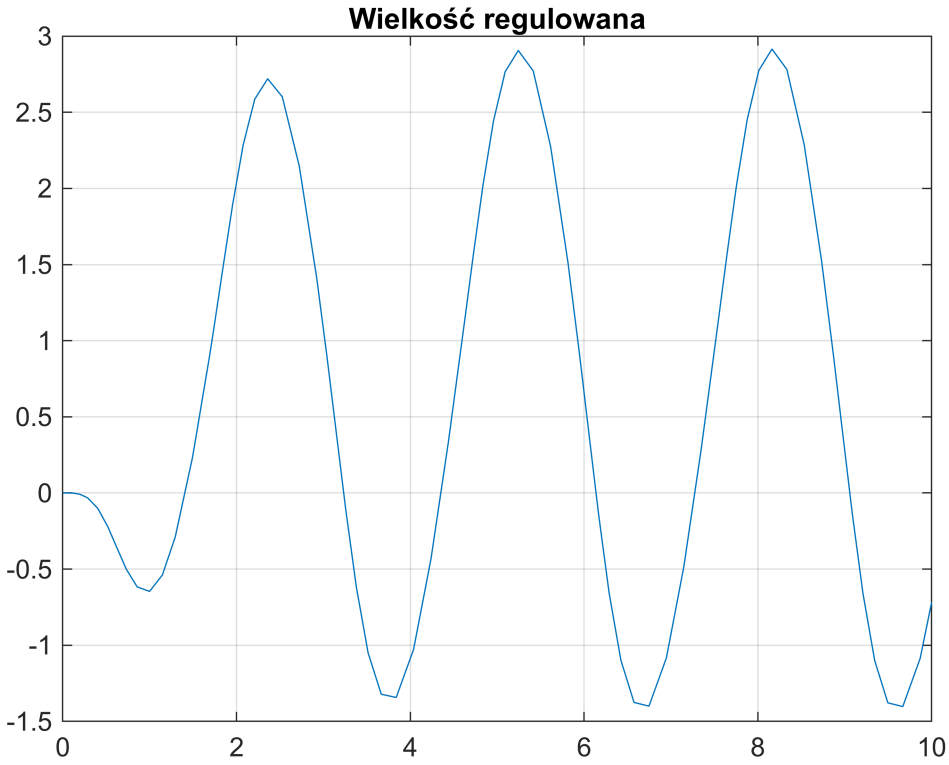

```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```



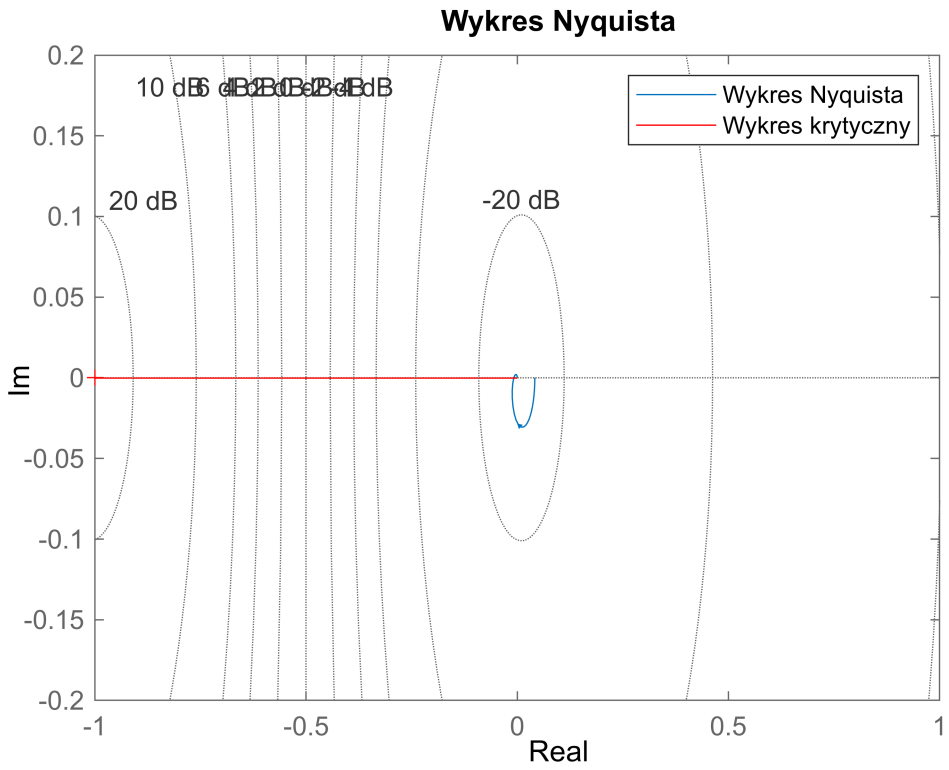

```
h = 0.1;
ym = 200;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```



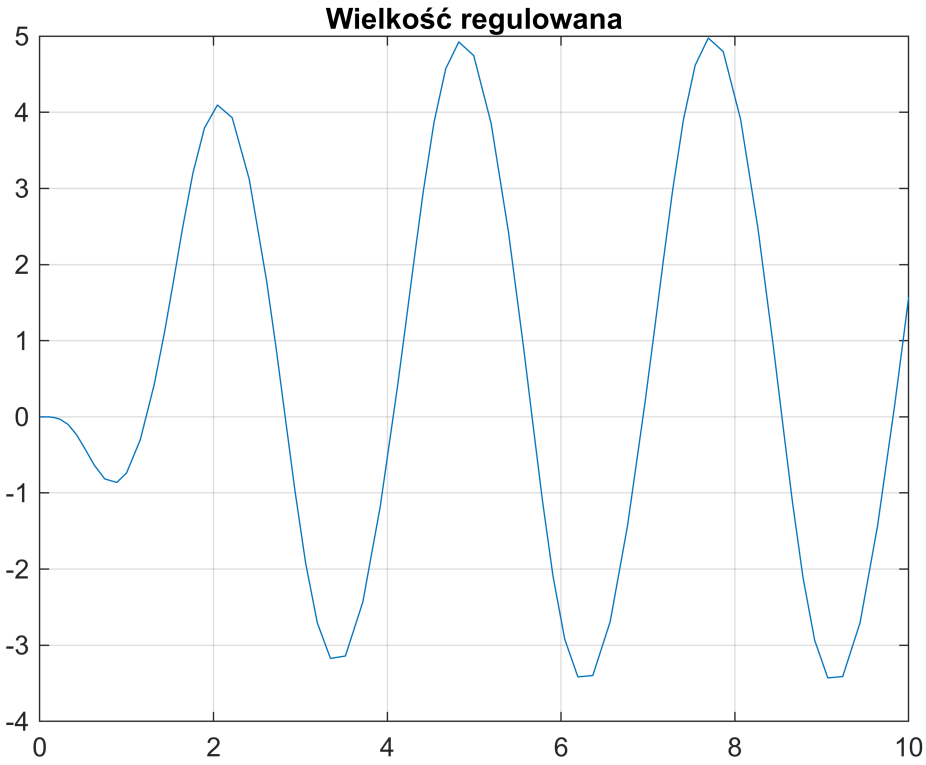

```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
title('Wielkość regulowana');
```




```
h = 0.1;
ym = 400;
c = (-pi * h)/(4*ym);
n = nyquistplot(tf((licz), (mian3)));
opt = getoptions(n);
opt.ShowFullContour = 'off';
setoptions(n, opt);

hold on;
plot([-1 0], [c, c], 'r');
axis([-1 1 -0.2 0.2]);
grid on;
title('Wykres Nyquista');
xlabel('Real');
ylabel('Im');
legend('Wykres Nyquista', 'Wykres krytyczny');
hold off
```




```
out3 = sim("ukl_reg_II.slx");
figure;
plot(out3.y.time, out3.y.signals.values)
grid on;
```




```
title('Wielkość regulowana');
```

4. Wnioski

Część I i II ćwiczenia przybliżyła nam działanie regulatorów trójpołożeniowych oraz rozumienie stabilności układu.

Regulatory bez histerezy nie posiadają części urojonej, a zwiększanie jej powoduje, że wykres przemieszcza się po części rzeczywistej. Amplituda sygnału jest związana ze zmienną ym.