Laboratório de Sistemas de Controle

Realimentação de Estados e Nyquist

**Professor**: Waldir Sampaio

**Assunto**: Experiência 10

Relatório apresentado como forma de obtenção de nota para a disciplina Laboratório de Sistemas de Controle 2008/2, ministrada pelo professor Waldir Sampaio na Universidade Federal do Amazonas.

Aluno / Matrícula: Adriano Mendes Gil\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20610326

Aluno / Matrícula: Gustavo Melo Medeiros\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20610250

Aluno / Matrícula: Jhony Braga da Silva\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20610052

Aluno / Matrícula: João Renato Aguiar\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20510051

Aluno / Matrícula: Rawlinson Gonçalves\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20610306

**Sumário**

1. Questão 1 – \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_4
2. Questão 2 –\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_15

### Laborátorio 10 – Realimentação de Estados e Nyquist

**Questão 1 -** Realimentação de Estados.

Planta:

Especificações:

* Overshoot < 10%;
* Erro zero ao degrau;
* Tempo de acomodação < 4s;

Simular para  (Degrau unitário).

* Obervar Overshoot => retirar efeitos de zeros => cancelando a instabilidade.
* **Simulação em malha aberta:**

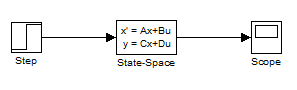


Figura 1. Planta feita no simulink do Matlab para simular o sistema em malha aberta.

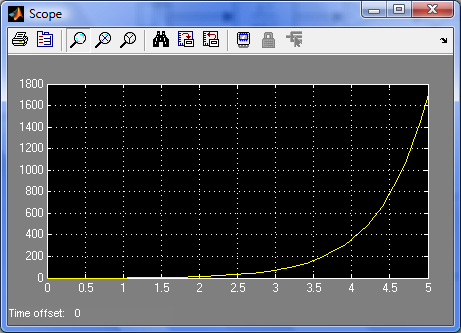


Figura 2. Resposta do sistema ao degrau unitário em Malha Aberta gerado pelo Matlab.

* **Simulação em malha fechada:**

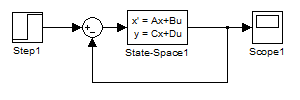


Figura 3. Planta feita no simulink do Matlab para simular o sistema em malha fechada.

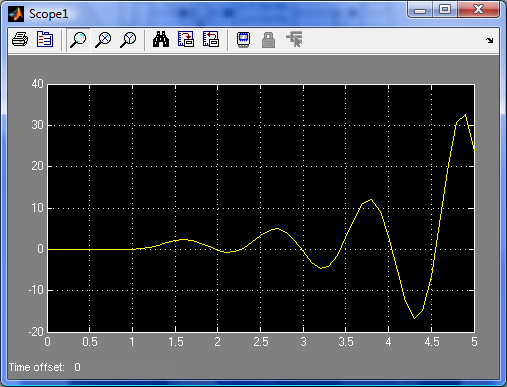


Figura 4. Resposta do sistema ao degrau unitário em Malha Fechada.

* **Cálculo do Controlador:**

Abaixo segue o código feito no Matlab para obter as informações do controlador:

clc; % limpa a linha de comando...

A = [1 1 0;-12 0 1; 20 0 0]

B = [0;1;1]

C = [21 -1 1]

D = [0]

P = ss(A, B, C, D);

Tf = tf(P)

ZPK\_TF = zpk(Tf)

pol\_caracteristico = poly(A)

D1 = pol\_caracteristico

D2 = conv([1 5],[1 (2\*0.6\*4) 16])

D3 = D2-D1

L\_BARRA = D3(2:4)

Vc = [B A\*B (A^2\*B)]

MUL = [1 -1 12;0 1 -1;0 0 1]

Q = Vc\*MUL

QINV = inv(Q)

A\_BARRA = QINV\*A\*Q

B\_BARRA = QINV\*B

C\_BARRA = C\*Q

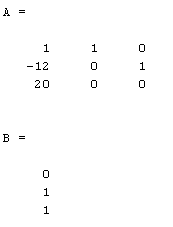
D\_BARRA = [0]

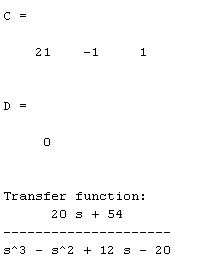
pol\_caracteristico\_barra = poly(A\_BARRA)

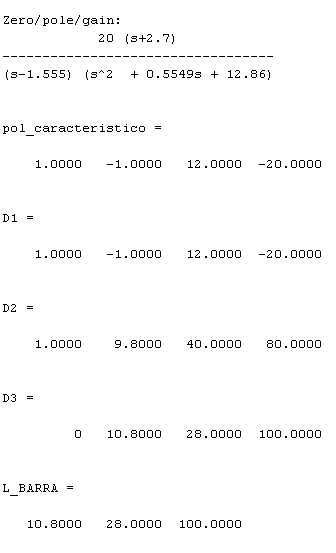
L = L\_BARRA\*QINV

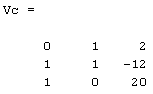
%TES = (A\_BARRA - B\_BARRA\*L\_BARRA)

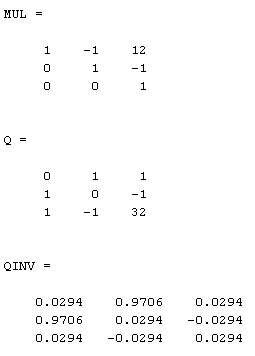
Abaixo segue o resultado gerado pelo Matlab ao executar o código acima:

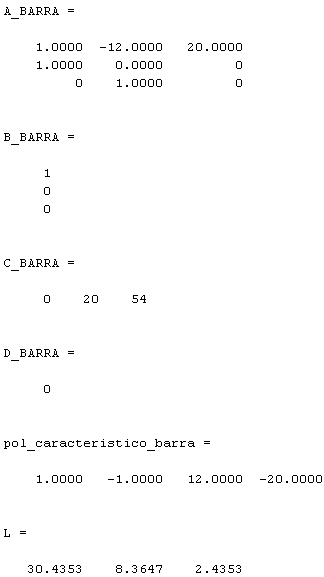












Com o valor L, obtido acima, e considerando que o ganho de M é igual a 1. Montamos a planta abaixo e simulamos para (Degrau unitário). Logo:

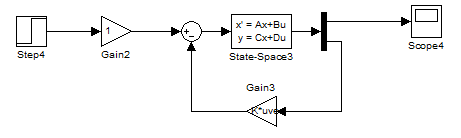


Figura 5. Planta feita no simulink do Matlab para simular o sistema em malha fechada, com controlador L e ganho M.

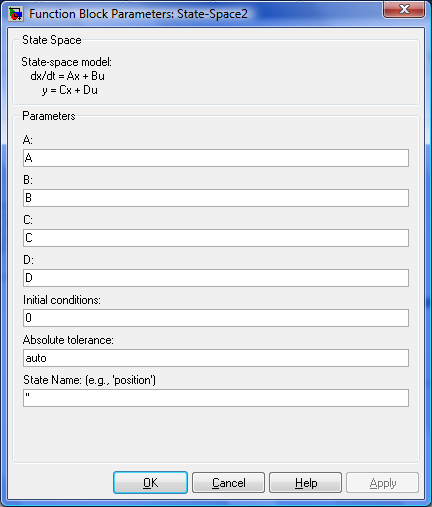


Figura 6. Mostra as definições do bloco State-space3 acima.

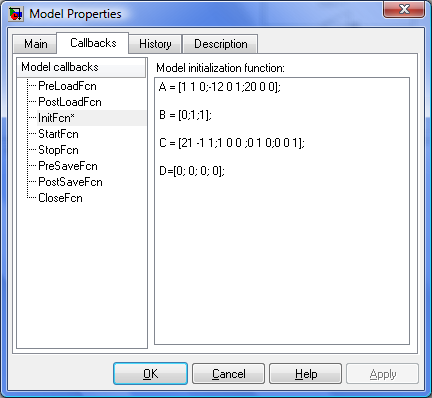


Figura 7. Mostra as definições do sistema da planta anterior.

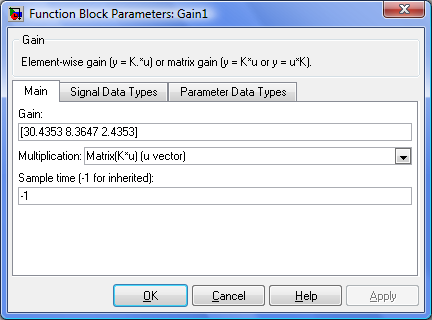


Figura 8. Mostras as definições do controlador L com os valores obtidos no Matlab.

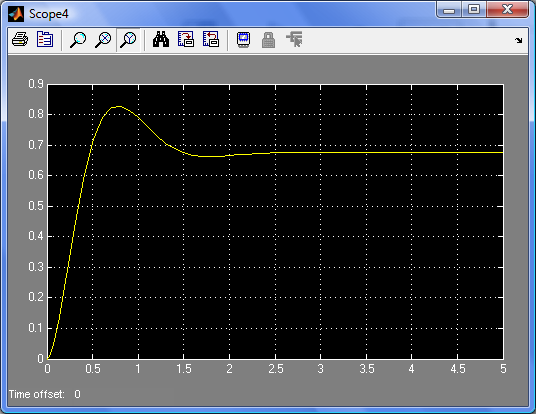


Figura 9. Gráfico gerado pelo Matlab ao simular a planta acima com o controlador.

Overshoot:

 => 





No gráfico da figura 9 é possível verificar que será necessário retirar o efeito do zero, pois ele está fazendo com que o overshoot seja muito grande. Para resolver este problema colocamos um pólo sobre o zero.

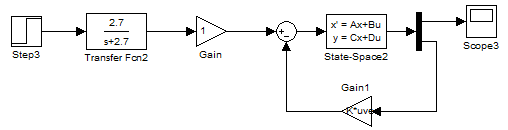


Figura 10. Planta feita no Matlab para simular o sistema em malha fechada, com um posicionamento de um pólo para anular o efeito de um zero.

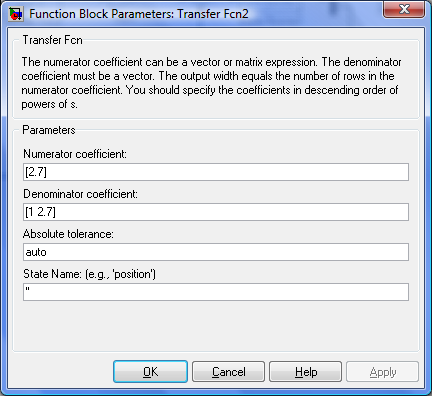


Figura 11. Mostra a definição do pólo para cancelar os efeitos do zero.

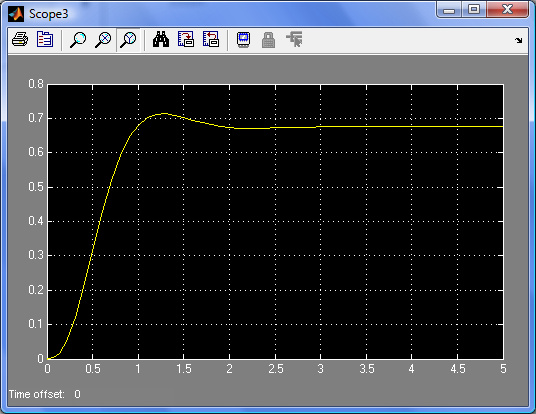


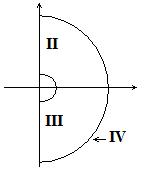
Figura 12. Gráfico gerado pelo Matlab ao simular a planta acima com o controlador e o posicionamento do pólo para anular os efeitos do zero.

**Questão 2** - Nyquist.

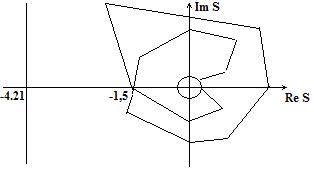
**Função de Transferência:**

|  |
| --- |
|  |







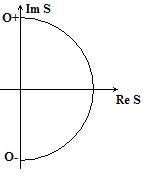
**Corte no Eixo Real**

**Corte no Eixo Imaginário**

=>

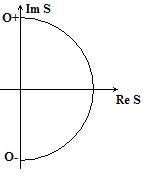
|  |
| --- |
|  |

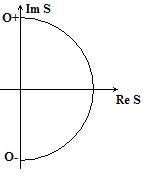
**Assíntota**

**Trecho II**

|  |  |
| --- | --- |
|  |  |
|  |  |
| 0 | 0 |
|  |  |



**Trecho III**

**Trecho IV**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| 0 | 0 |

**Malha Aberta**

**Malha Fechada**

Sistema Instável

P = 0

Z = 0

Logo, é estável neste intervalo

P = 0

Z = 1+0=1

Instável => pólo em

Abaixo segue o código feito no Matlab para simular Nyquist:

clc; % limpa a linha de comando...

N = 30;

D = [1 2 9 8 0];

w = 0.1:0.1:1000;

P = tf(N, D)

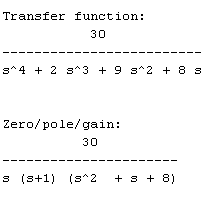
sys = zpk(P)

nyquist(P);

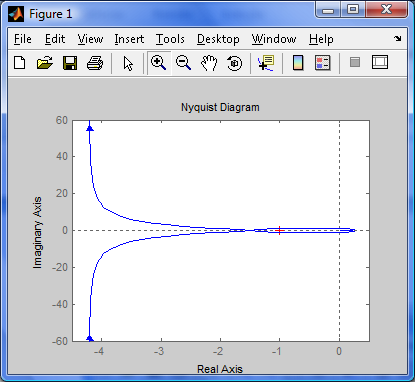
axis([-20 20 -20 20]);

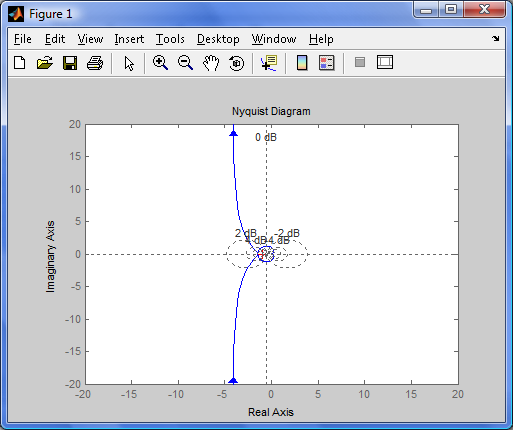
grid on;

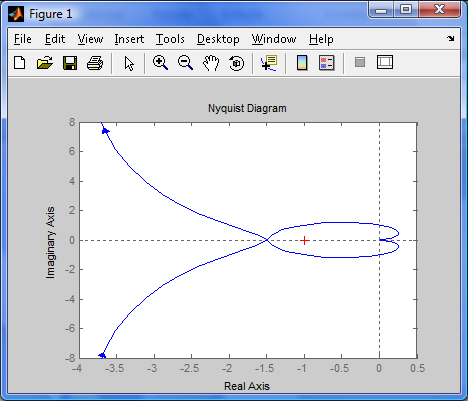
Resultado gerado pelo Matlab ao executar o código acima:

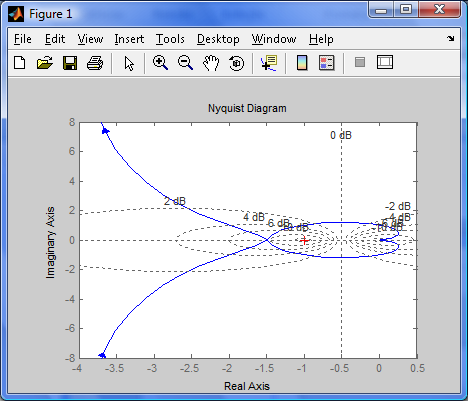


Abaixo segue o gráfico de Nyquist gerado pelo Matlab:



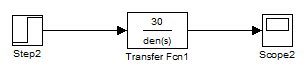




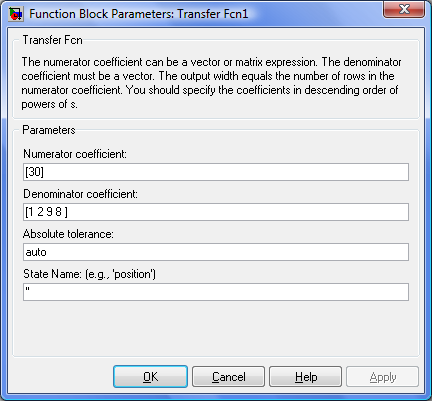




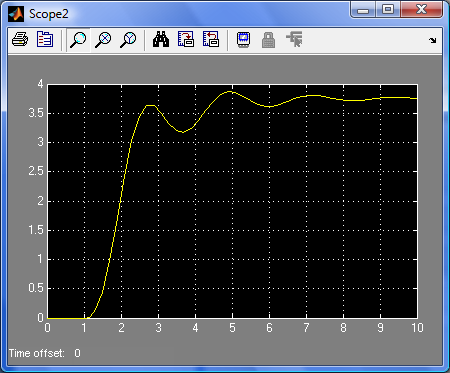
A planta abaixo está simulando a função G(S) acima:



A figura acima mostra a planta feita no Matlab para simular a função de transferência acima (dada na questão) para realização do critério de Nyquist.



A figura mostra as definições do bloco da função de transferência da planta acima.



A figura mostra a saída gerada pelo Matlab ao simular a planta acima.