



UNIVERSIDADE DA BEIRA INTERIOR

DEPARTAMENTO DE ENGENHARIA ELECTROMECHANICA

CONTROLO DE SISTEMAS

APONTAMENTOS DE MATLAB

CONTROL SYSTEM Toolbox

*Pedro Dinis Gaspar
António Espírito Santo
J. A. M. Felipe de Souza*

Edição Abril 2002

ÍNDICE

1. - INTRODUÇÃO	3
2. - REPRESENTAÇÃO DE SISTEMAS.....	4
2.1. REPRESENTAÇÃO DA FUNÇÃO DE TRANSFERÊNCIA DO SISTEMA	4
2.2. REPRESENTAÇÃO DE SISTEMAS EM ESPAÇO DE ESTADOS.....	4
2.3. CONVERSÃO DA REPRESENTAÇÃO DE SISTEMAS CONTÍNUOS	5
2.3.1. Conversão da representação de sistemas para função de transferência.....	5
2.3.2. Conversão da representação de sistemas para espaço de estados	6
2.3.3. Obtenção dos Pólos, Zeros e Ganho do sistema	7
3. - MODELAÇÃO DE SISTEMAS.....	8
3.1. BLOCOS EM CASCATA	8
3.2. SOMADOR DE BLOCOS	9
3.3. REALIMENTAÇÃO UNITÁRIA DE SISTEMAS	9
3.4. REALIMENTAÇÃO NÃO UNITÁRIA DE SISTEMAS.....	10
3.5. CONSTRUÇÃO DE SISTEMAS DE 2ª ORDEM A PARTIR DE ω_n E DE ζ	11
4. - RESPOSTA NO DOMÍNIO DO TEMPO	12
4.1. RESPOSTA A UMA ENTRADA EM DEGRAU UNITÁRIO.....	12
4.2. RESPOSTA A UMA ENTRADA EM RAMPA UNITÁRIA	15
4.3. RESPOSTA A UMA ENTRADA EM IMPULSO	16
4.4. RESPOSTA A ENTRADAS COM CONDIÇÕES INICIAIS NÃO NULAS.....	18
4.5. RESPOSTA A UMA ENTRADA ARBITRÁRIA	19
5. - ANÁLISE DO LUGAR GEOMÉTRICO DAS RAÍZES.....	21
5.1. MAPEAMENTO DE PÓLOS E ZEROS DE SISTEMAS EM MALHA ABERTA.....	21
5.2. LUGAR GEOMÉTRICO DAS RAÍZES (ROOT LOCUS)	22
5.3. ANÁLISE DE VALOR DO GANHO DE REALIMENTAÇÃO DO SISTEMA	24
5.4. CURVAS DE ω_n E DE ζ CONSTANTE.....	24
6. - RESPOSTA NO DOMÍNIO DA FREQUÊNCIA.....	25
6.1. DIAGRAMAS DE BODE	25
6.1.1. Diagrama de Bode : Módulo e Ângulo de Fase (BODE).....	31
6.1.2. Margem de Ganho e Margem de Fase	33
6.2. DIAGRAMA DE NYQUIST.....	35
6.3. DIAGRAMA DE NICHOLS (NICHOLS)	36
7. - REFERÊNCIAS BIBLIOGRÁFICAS	38

1. - INTRODUÇÃO

Estes apontamentos pretendem descrever e introduzir os comandos básicos da *CONTROL SYSTEM Toolbox* do MATLAB para permitir o estudo de sistemas dinâmicos lineares.

O documento tal como está estruturado, pressupõe alguma familiaridade de utilização do MATLAB, nomeadamente em termos de representação e manipulação de matrizes e polinómios e ainda de elaboração de gráficos.

O comando **help control** fornece uma lista das diversas funções existentes na caixa de ferramentas : *CONTROL SYSTEM Toolbox*. Para obter informação mais detalhada sobre cada uma das funções pode-se usar o comando **help nome_da_função**.

Este trabalho baseia-se em documentos públicos disponíveis na Internet, nos manuais do MATLAB, no livro *Solving Control Engineering Problems with MATLAB* e nas informações disponíveis na página da internet da companhia Mathworks.

2. - REPRESENTAÇÃO DE SISTEMAS

No MATLAB os sistemas dinâmicos podem representar-se por intermédio de uma *função de transferência* ou por um modelo em *espaço de estados*. Na representação por função de transferência definem-se os coeficientes dos polinómios do numerador e denominador. Na representação em espaço de estados definem-se as quatro matrizes que caracterizam o modelo.

2.1. REPRESENTAÇÃO DA FUNÇÃO DE TRANSFERÊNCIA DO SISTEMA

As chamadas *funções de transferência* são funções usadas para caracterizar as relações entre entrada e saída de componentes ou sistemas que possam ser descritos por equações diferenciais lineares invariantes no tempo.

A representação de sistemas em função de transferência pressupõe o emprego das transformadas de Laplace às equações diferenciais lineares que descrevem o modelo matemático do sistema dinâmico.

Assim, a equação diferencial linear pode ser transformada numa equação algébrica composta por um **numerador** e um **denominador**, em função de uma variável complexa s .

Por exemplo, o seguinte sistema que se encontra apresentado na forma de função de transferência deverá ser introduzido no MATLAB, pelos coeficientes das sucessivas potências dos polinómios que surgem ordenados por ordem significativa decrescente:

$$\frac{Y(s)}{R(s)} = \frac{3s^2 - 2}{s^2 + 2s + 4}$$

```
>> num = [ 3 - 2 ]  
>> den = [ 1 2 4 ]  
>> printsys ( num , den )
```

Esta última função **printsys** apresentará na *Janela de Comando* (Command Windows), a função de transferência que descreve o sistema :

$$\begin{array}{r} \text{num/den} = \\ 3 s - 2 \\ \hline s^2 + 2 s + 4 \end{array}$$

2.2. REPRESENTAÇÃO DE SISTEMAS EM ESPAÇO DE ESTADOS

Um sistema dinâmico que consiste num número finito de elementos concentrados pode ser escrito por equações diferenciais ordinárias em que o tempo é a variável independente. Fazendo uso de notação matricial-vectorial, uma equação diferencial de ordem n pode ser representada por uma equação matricial-vectorial de primeira ordem. Se n elementos do vector são um conjunto de variáveis de estado, então a equação diferencial matricial vectorial é denominada de *equação de estado*. Deste modo, um sistema representado na forma de equações de estado será dado por:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

Considerando o seguinte sistema representado no espaço de estado, a sua introdução no MATLAB efectua-se pelo método comum de introdução de matrizes na *Janela de Comando*:

$$\dot{x} = \begin{bmatrix} 2 & -1 \\ 3 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = [2 \quad -0.5]x$$

```
>> A = [ 2 -1 ; 3 -1 ]
>> B = [ 1 0 ]'
>> C = [ 2 -0.5 ]
>> D = [ 0 ]
```

2.3. CONVERSÃO DA REPRESENTAÇÃO DE SISTEMAS CONTÍNUOS

O MATLAB possui diversas funções que convertem a representação na qual o sistema está expresso. É possível realizar conversões de representação de sistemas *de* e *para*:

- Função de Transferência (*transfer function – tf*);
- Espaço de Estados (*state space – ss*);
- Zeros, Pólos e Ganho (*zero pole gain – zp*).

2.3.1. Conversão da representação de sistemas para função de transferência

É possível converter a representação de sistemas em equações de estados ou a partir dos pólos, zeros e ganho através do uso das seguintes funções:

ss2tf Conversão de representação em espaço de estados para função de transferência.
zp2tf Representação em função de transferência a partir dos Pólos, Zeros e Ganho do sistema.

Considere o seguinte sistema dado na forma de equações de estado:

$$\dot{x} = \begin{bmatrix} -2 & -1 \\ 1 & -2 \end{bmatrix} x + \begin{bmatrix} 1 \\ 2 \end{bmatrix} u$$

$$y = [1 \quad 0]x$$

```
>> A = [ -2 -1 ; 1 -2 ]
>> B = [ 1 2 ]'
>> C = [ 1 0 ]
>> D = [ 0 ]
>> [ num,den ] = ss2tf( A, B, C, D )
```

$$\begin{array}{rccccccc} \text{num} = & & & & & & \\ & 0 & & 1 & & & 0 \\ \\ \text{den} = & & & & & & \\ & 1 & & 4 & & & 5 \end{array}$$

Isto é, a função de transferência do sistema será:

$$\frac{\text{num}}{\text{den}} = \frac{s}{s^2 + 4s + 5}$$

No caso do sistema estar descrito pelos seus Pólos, Zeros e Ganho, a conversão para a representação em função de transferência será dada por:

```
>> z = 0                                % Zero do sistema.
>> p1 = -2 + i                          % Pólo do sistema.
>> p2 = -2 - i                          % Pólo do sistema.
>> k = 1                                % Ganho.
>> [ num,den ] = zp2tf( z , [ p1 p2 ] , k)
```

Obtendo-se a mesma função de transferência do sistema que anteriormente.

2.3.2. Conversão da representação de sistemas para espaço de estados

Tal como no item anterior, é possível converter a representação de sistemas em função de transferência ou a partir dos pólos, zeros e ganho para a representação em espaço de estados através do uso das seguintes funções:

tf2ss Conversão da função de transferência para a representação em modelo de espaço de estados.
zp2ss Representação do modelo de espaço de estados a partir dos Pólos, Zeros e Ganho do sistema.

Considere a função de transferência do seguinte sistema :

$$\frac{Y(s)}{R(s)} = \frac{3s^2 - 2s}{s^2 + 2s + 4}$$

```
>> num = [ 3 -2 ]
>> den = [ 1 2 4 ]
>> [ A, B, C, D ] = tf2ss ( num, den )
```

A sua representação em espaço de estados será :

$$\begin{array}{rcc} A = & & \\ & -4 & -5 \\ & 1 & 0 \\ \\ B = & & \\ & 1 & \\ & 0 & \end{array}$$

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 \end{bmatrix}$$

Se fossem dados os Pólos, Zeros e Ganho do sistema, a conversão para a representação em espaço de estados seguiria a mesma metodologia apresentada para o caso da função de transferência, tendo em consideração que se pretende obter as matrizes **A**, **B**, **C**, **D** e não o **numerador** e **denominador**.

2.3.3. Obtenção dos Pólos, Zeros e Ganho do sistema

Dado qualquer sistema representado no espaço de estados ou descrito por uma função de transferência, é possível extrair os seus os Pólos, Zeros e o Ganho através do uso das seguintes funções:

- tf2zp** Obtenção dos Pólos, Zeros e Ganho do sistema a partir da função de transferência.
- ss2zp** Obtenção dos Pólos, Zeros e Ganho do sistema a partir da sua representação em espaço de estados.

Os Pólos, Zeros e o Ganho dos sistemas anteriores representados no espaço de estados e em forma de função de transferência são dados por :

```
>> % Sistema representado na forma de equação de transferência
>> num = [ 0 1 0 ]
>> den = [ 1 2 5 ]
>> [ z, p, k ] = tf2zp ( num, den )
```

```
>> % Sistema representado no espaço de estados
>> A = [ -2 -1 ; 1 -2 ]
>> B = [ 1 2 ]'
>> C = [ 1 0 ]
>> D = [ 0 ]
>> [ z, p, k ] = ss2zp ( A, B, C, D )
```

$$z = \begin{bmatrix} 0 \end{bmatrix}$$

$$p = \begin{bmatrix} -1 + 2i \\ -1 - 2i \end{bmatrix}$$

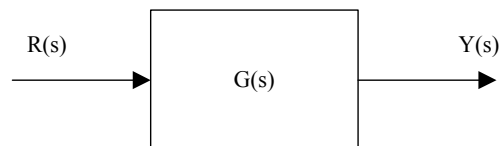
$$k = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

NOTA : Em todos os tipos de conversões apenas é necessário ter em consideração que a função de transferência é dada por um **numerador** e **denominador**, enquanto a representação em espaço de estado pelas matrizes **A**, **B**, **C**, **D**. A representação do sistema pode ainda ser expressa pelos seus Pólos (**p**), Zeros (**z**) e Ganho (**k**).

3. - MODELAÇÃO DE SISTEMAS

A *CONTROL SYSTEM Toolbox* do MATLAB possui um conjunto de funções que permitem obter representações de sistemas constituídos por diversos subsistemas interligados.

O *diagrama de blocos* de um sistema é uma representação ilustrativa das funções desempenhadas por cada um dos componentes e fluxos de sinais. Tal diagrama indica as inter-relações que existem entre os vários componentes.



3.1. BLOCOS EM CASCATA

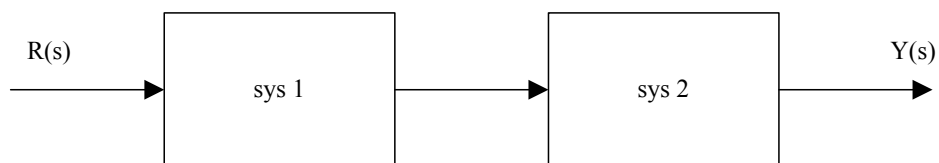
Esta função permite a associação de blocos em cascata, tendo em consideração que os dois sistemas deverão ser do mesmo tipo (contínuos ou discretos).

Aqui e doravante, considera-se que um sistema *sys* pode estar representado :

- Na forma de equação de transferência : $\text{sys} = \text{num}, \text{den}$
- No modelo de espaço de estado : $\text{sys} = A, B, C, D$

A seguinte função implementa a associação de blocos em cascata :

```
>> sys = series ( sys1 , sys2 )
```



Este comando é equivalente à multiplicação de polinómios no caso dos sistemas estarem representados na forma de equação de transferência que faz uso da função *conv*. Por exemplo, dados os polinómios :

$$\text{sys1} : \frac{\text{num1}}{\text{den1}} = \frac{2}{s^2 + 2s + 3}$$
$$\text{sys2} : \frac{\text{num2}}{\text{den2}} = \frac{s + 1}{4s^2 + 5s + 6}$$

A sua multiplicação é obtida a partir de:

```
>> num1 = [ 0 0 2 ]  
>> den1 = [ 1 2 3 ]  
>> num2 = [ 0 1 1 ]  
>> den2 = [ 4 5 6 ]
```



```
>>num = conv ( num1 , num2 )
>>den = conv ( den1 , den2 )
```

num =

2 2

den =

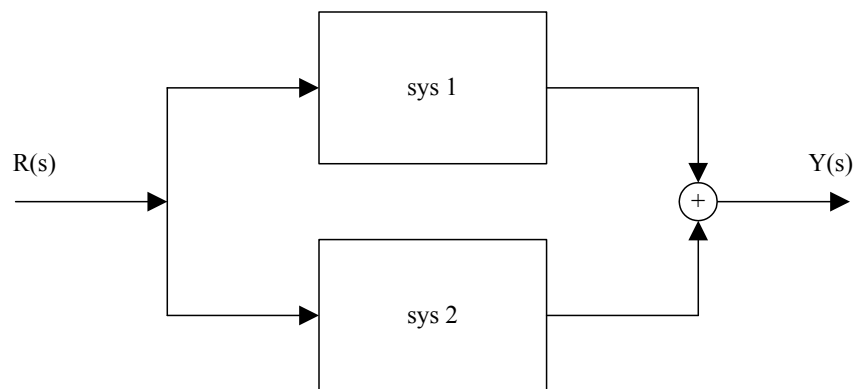
4 13 28 27 18

Isto é : sys : $\frac{Y(s)}{R(s)} = \frac{num}{den} = \frac{2s + 2}{4s^4 + 13s^3 + 28s^2 + 27s + 18}$

3.2. SOMADOR DE BLOCOS

Esta função permite a concepção de blocos em paralelo, tendo em consideração que os dois sistemas deverão ser do mesmo tipo (contínuos ou discretos).

```
>> sys = parallel ( sys1 , sys2 )
```

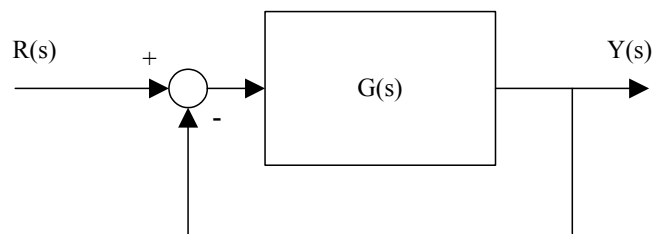


Este comando é equivalente à soma directa de polinómios no caso dos sistemas estarem representados na forma de equação de transferência.

3.3. REALIMENTAÇÃO UNITÁRIA DE SISTEMAS

A obtenção de um sistema de malha fechada com realimentação unitária (*closed loop*) é dada pela utilização do seguinte comando :

```
>> sys = loop ( G )
```



Considerando o sistema de malha aberta $G(s)$, o sistema de malha fechada com realimentação unitária $Y(s)/R(s)$ será dado por :

$$G(s) = \frac{num}{den} = \frac{s - 6}{s^2 - s + 5}$$

$$\frac{Y(s)}{R(s)} = \frac{n}{d} = \frac{G(s)}{1 + G(s)} = \frac{s - 6}{s^2 - 1}$$

```
>> num = [ 0  1 - 6 ]
>> den = [ 1 - 1  5 ]
>> [n,d] = cloop ( num , den )
```

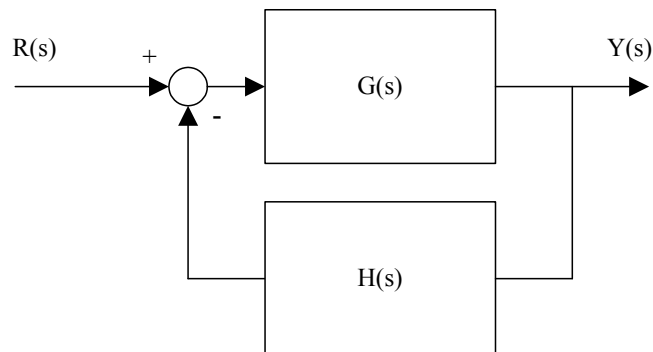
```
n =
      1      - 6

d =
      1      0      - 1
```

3.4. REALIMENTAÇÃO NÃO UNITÁRIA DE SISTEMAS

A obtenção de um sistema de malha fechada com realimentação não unitária é dada pela utilização do seguinte comando :

```
>> sys = feedback ( G, H )
```



Considerando o sistema constituído pelos blocos $G(s)$ e $H(s)$:

$$G(s) = \frac{num1}{den1} = \frac{1}{s + 2}$$

$$H(s) = \frac{num2}{den2} = \frac{1}{s}$$

A função de transferência de malha aberta do sistema será dada por : $G(s)H(s)$:

$$G(s)H(s) = \frac{num3}{den3} = \frac{1}{s(s + 2)}$$

E a função de transferência de malha fechada do sistema $Y(s)/R(s)$ será dada por :

$$\frac{Y(s)}{R(s)} = \frac{num}{den} = \frac{G(s)}{1 + G(s)H(s)} = \frac{s}{s^2 + 2s + 1}$$

```
>> num1 = [ 0 1]
>> den1 = [ 1 2]
>> num2 = [ 0 1]
>> den2 = [ 1 0]
>> [ num , den ] = feedback ( num1 , den1 , num2 , den2 )
```

```
num =
      1      0

den =
      1      2      1
```

NOTA : Por defeito, assume-se que **feedback(sys1,sys2)** representa realimentação não unitária negativa. Para aplicar realimentação não unitária positiva utilizar o comando : **feedback(sys1,sys2, 1)**.

3.5. CONSTRUÇÃO DE SISTEMAS DE 2ª ORDEM A PARTIR DE ω_n E DE ζ

Considerando um sistema de 2ª Ordem expresso por :

$$\frac{Y(s)}{R(s)} = \frac{num}{den} = \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

A sua construção na forma de função de transferência ou no espaço de estados pode ser realizada a partir da frequência natural : ω_n e do coeficiente de amortecimento : ζ .

No caso de se pretender gerar um sistema na forma de equação de transferência com uma frequência natural de $\omega_n = 2.4$ [rad/seg] e com um coeficiente de amortecimento : $\zeta = 0.4$ deverá introduzir-se na *Janela de Comando* :

```
>> [ num , den ] = ord2 ( 2.4 , 0.4 )
```

```
num =
      0      0      1

den =
      1.0000      1.9200      5.7600
```

Que corresponde à função de transferência :

$$\frac{Y(s)}{R(s)} = \frac{num}{den} = \frac{1}{s^2 + 1.92s + 5.76}$$

4. - RESPOSTA NO DOMÍNIO DO TEMPO

Na análise e projecto de sistemas de controlo é necessário possuir uma base de comparação do desempenho de vários desses sistemas. Esta base pode ser obtida especificando-se sinais de teste de entrada particulares e comparando-se as respostas dos vários sistemas.

Os sinais de entrada de teste típicos usualmente utilizados são as funções *degrau*, *rampa* e *impulso*.

A determinação de qual ou quais destes sinais de entrada típicos devem ser usados para analisar as características do sistema depende da forma da entrada a que o sistema será sujeito mais frequentemente durante a operação normal.

Durante o projecto de um sistema de controlo, deverá ser previsto o comportamento dinâmico do sistema a partir do conhecimento dos componentes. A característica mais importante do comportamento dinâmico de um sistema de controlo é a *Estabilidade Absoluta*, isto é, se o sistema é estável ou instável.

Um sistema de controlo está em equilíbrio se, na ausência de qualquer perturbação ou entrada, a saída permanece no mesmo estado. Mais especificamente, um determinado sistema de controlo invariante no tempo e linear é estável se a saída voltar ao seu estado de equilíbrio quando o sistema é sujeito a uma perturbação.

A *CONTROL SYSTEM Toolbox* do MATLAB possui um conjunto de funções que permitem obter a resposta de um sistema no domínio do tempo. Cada uma destas funções será objecto de uma descrição detalhada nos itens que se seguem.

4.1. RESPOSTA A UMA ENTRADA EM DEGRAU UNITÁRIO

A resposta de sistemas a degrau unitário, apresenta características totalmente diferentes entre sistemas de 1ª Ordem e sistemas de Ordem superior (Neste item será dada atenção especial aos sistemas de 2º Ordem).

Considerando que a função de transferência de malha fechada de um determinado sistema é dada por :

$$\frac{Y(s)}{R(s)} = \frac{num}{den} = \frac{K_0 \omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2}$$

O comportamento dinâmico deste sistema de 2ª Ordem pode ser descrito pela frequência natural: ω_n e pelo coeficiente de amortecimento : ζ .

$0 < \zeta < 1$: Sistema sub-amortecido	$\Rightarrow$ Resposta transiente oscilatória.
$\zeta = 1$: Sistema criticamente amortecido	$\Rightarrow$ Resposta transiente não oscilatória.
$\zeta > 1$: Sistema sobre-amortecido	$\Rightarrow$ Resposta transiente não oscilatória.

É necessário efectuar uma definição das especificações da resposta transiente de sistemas de 2ª Ordem a uma entrada em degrau unitário, já que frequentemente as características de desempenho desejadas para o sistema de controlo são especificadas em termos de grandezas no domínio do tempo para um tipo de entrada que seja simples, mas severa (caso da entrada em degrau unitário):

Tempo de Atraso : t_d

Corresponde ao tempo necessário para que a resposta alcance (pela primeira vez) metade do valor final.

Tempo de Subida : t_r

Por norma, corresponde ao tempo necessário para que a resposta passe de 0% a 100% do seu valor final.

Sabendo que a *Frequência Natural Amortecida* é dada por :

$$\omega_d = \omega_n \sqrt{1 - \zeta^2}$$

O *Tempo de Subida* virá :

$$t_r = \frac{1}{\omega_d} \tan^{-1} \left(-\frac{\omega_d}{\zeta \omega_n} \right)$$

Instante de pico : t_p

Corresponde ao tempo necessário para que a resposta alcance o primeiro pico do *Sobre-sinal (Overshoot)* :

$$t_p = \frac{\pi}{\omega_d}$$

Sobre-sinal Máximo (Overshoot) : M_p

Valor de pico da curva da resposta medido a partir do valor final de regime estacionário da resposta. Geralmente é definido em termos percentuais :

$$M_p = e^{-\frac{\zeta \pi}{\sqrt{1 - \zeta^2}}}$$

Tempo de Estabelecimento : t_{ac}

Corresponde ao tempo necessário para que a curva da resposta alcance e permaneça dentro de uma faixa em torno do valor final. Por norma, esta faixa é especificada com uma magnitude dada por uma percentagem absoluta de 2% ou 5% do valor final.

$$t_{ac(2\%)} = \frac{4}{\zeta \omega_n} \quad t_{ac(5\%)} = \frac{3}{\zeta \omega_n}$$

O MATLAB não possui funções para retirar automaticamente os valores destas especificações da resposta de sistemas de 2ª Ordem no domínio do tempo, logo terão que ser calculados aritmeticamente após a obtenção da curva da resposta do sistema.

A função **step** calcula a resposta de um sistema no domínio do tempo a uma entrada em degrau unitário. É considerado um estado inicial nulo quando o sistema está representado no espaço de estado.

Quando o comando é utilizado sem argumentos, esta função gera o traçado da resposta ao degrau unitário na *Janela Gráfica*. A duração da simulação é determinada automaticamente baseada nos *Pólos e Zeros* do sistema.

```
>> step ( sys )
```

No entanto, no caso de se pretender a obtenção dos valores da resposta do sistema (da saída: **y**, tempo: **t** e a trajetória de estado: **x**), deverão ser utilizados os argumentos no comando. O traçado da resposta do sistema poderá ser obtido através da utilização das função básica do MATLAB para a criação de gráficos: **plot** :

sys = num , den : Função de transferência ou **sys = A, B, C, D** : Modelo em espaço de estados.

```
>> [ y, t ] = step ( sys )           % Sistema representado por função de transferência.
>> [ y, x, t ] = step ( sys )       % Sistema representado por um modelo em espaço de estado.
>> plot ( t, y )
```

Do mesmo modo, o utilizador poderá estabelecer a duração da simulação através da imposição do tempo:

```
>> t = 0 : 0.1 : 10
>> step ( sys,t )
```

A função **step** permite a introdução de distintos argumentos que definam vários sistemas, de modo a que sejam apresentados os traçados da resposta sobrepostos na mesma *Janela Gráfica*, podendo os tipos, cores e marcadores das linhas de cada sistema serem definidos como nas funções básicas de criação de gráficos:

```
>> step ( sys1 , 'y:' , sys2 , 'g-' )
```

Considere a função de transferência de malha fechada do seguinte sistema :

$$\frac{Y(s)}{R(s)} = \frac{num}{den} = \frac{25}{s^2 + 4s + 25}$$

A resposta do sistema no domínio do tempo a uma entrada em degrau unitário (**R(s)=1/s**), consiste em :

$$Y(s) = \frac{num}{den} = \frac{25}{s^2 + 4s + 25} \cdot R(s) \quad \Leftrightarrow \quad Y(s) = \frac{num}{den} = \frac{25}{s^2 + 4s + 25} \cdot \frac{1}{s}$$

As linhas de comando a introduzir para a visualização da resposta do sistema, serão :

```
>> num = [ 0 0 25 ]
>> den = [ 1 4 25 ]
```

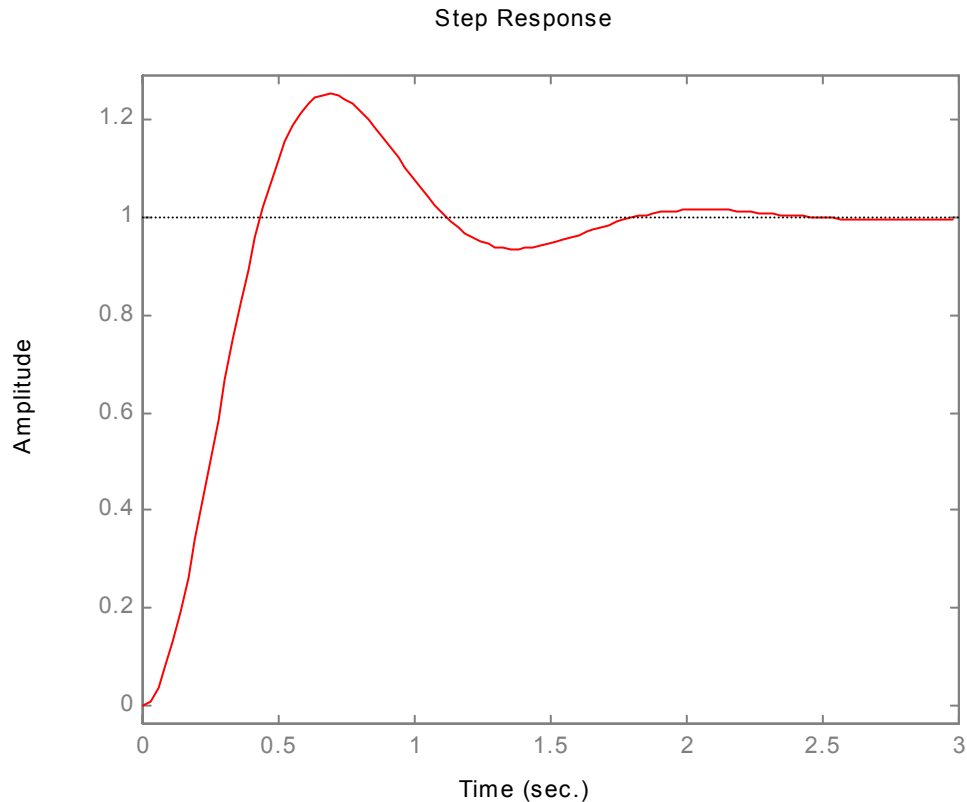
```
>> step ( num , den )
```

```
num =
```

```
0      0      25
```

```
den =
```

```
1      4      25
```



NOTA : O sistema também poderá estar representado no espaço de estados, sendo a resposta a um degrau unitário obtida com a função expressa do seguinte modo : ***step(A,B,C,D)***

4.2. RESPOSTA A UMA ENTRADA EM RAMPA UNITÁRIA

O MATLAB não dispõe de nenhuma função para obter a resposta de um sistema no domínio do tempo a uma entrada em rampa unitária. Para se obter a resposta temporal a este sinal de entrada particular, utiliza-se a função ***step*** aumentando um grau os coeficientes das sucessivas potências do polinómio que traduzem o ***denominador*** da função de transferência do sistema. A metodologia de utilização do comando permanece igual ao caso anterior. Considerando a mesma função de transferência do sistema, a resposta do sistema no domínio do tempo a uma entrada em rampa unitária ($R(s)=1/s^2$), consiste em :

$$Y(s) = \frac{num}{den} = \frac{25}{s^2 + 4s + 25} \cdot R(s) \quad \Leftrightarrow \quad Y(s) = \frac{num}{den} = \frac{25}{s^2 + 4s + 25} \cdot \frac{1}{s^2}$$

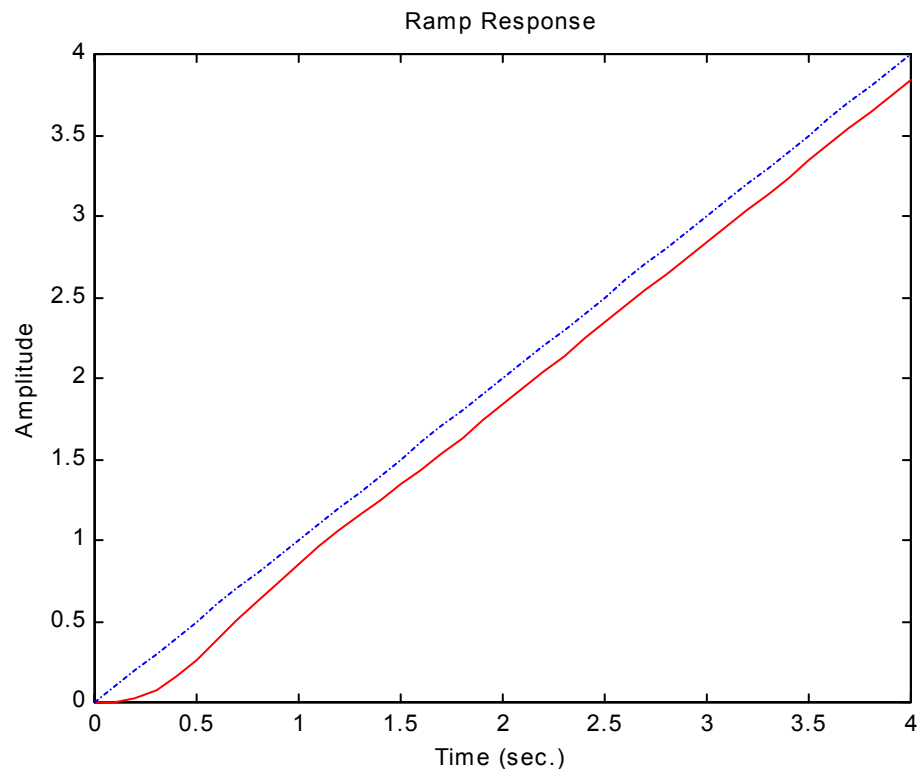
$$Y(s) = \frac{num}{den} = \frac{25}{s(s^2 + 4s + 25)} \cdot \frac{1}{s}$$

Tal que as funções a introduzir na *Janela de Comando* para a visualização da resposta do sistema, serão :

```
>> num = [ 0 0 25 ]
>> den1 = [ 1 4 25 ]
>> den2 = [ 0 1 0 ]
>> den = conv ( den1 , den2 )
>> step ( num , den )
```

```
num =
    0         0         0        25

den =
    1         4        25         0
```



O **Erro Estacionário** para a resposta no domínio do tempo de um sistema de 2ª Ordem sujeito a uma entrada em rampa unitária é dado por :

$$e_{ss} = \frac{2\zeta}{\omega_n}$$

4.3. RESPOSTA A UMA ENTRADA EM IMPULSO

A função ***impulse*** calcula a resposta de um sistema no domínio do tempo a uma entrada em impulso.

A sua aplicação está sujeita aos mesmos procedimentos e restrições que a função ***step***.

A resposta do sistema anterior no domínio do tempo a uma entrada em impulso de Dirac ($R(s)=I$), consiste em :

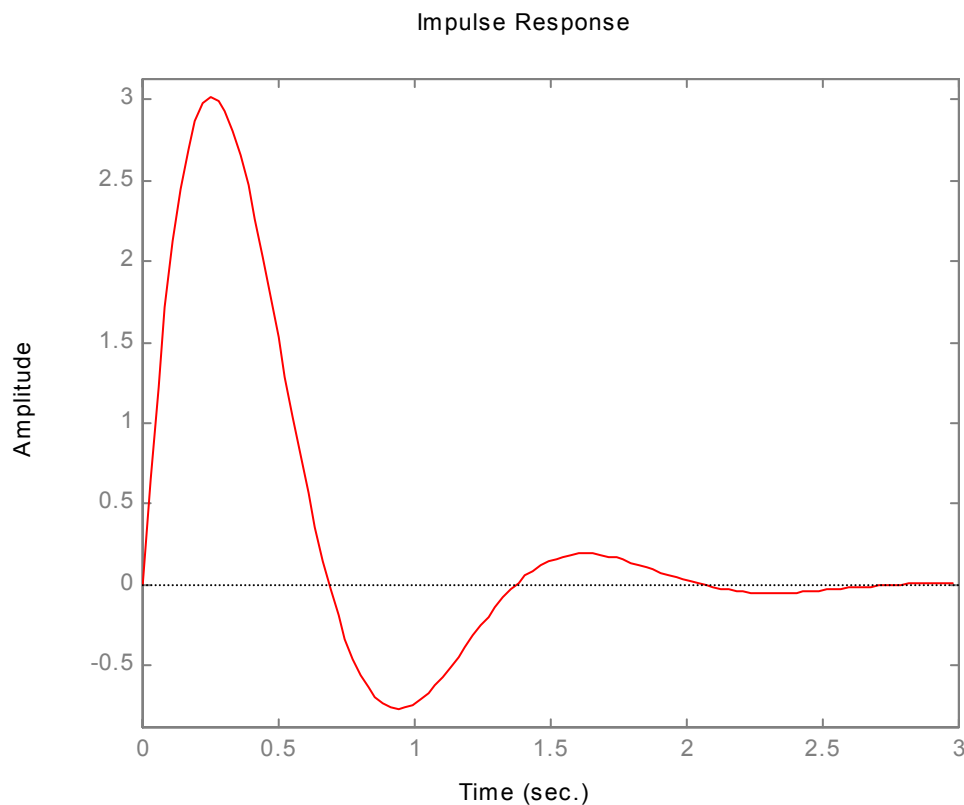
$$Y(s) = \frac{num}{den} = \frac{25}{s^2 + 4s + 25} \cdot R(s) \quad \Leftrightarrow \quad Y(s) = \frac{num}{den} = \frac{25}{s^2 + 4s + 25} \cdot 1$$

As linhas de comando a introduzir para a visualização da resposta do sistema, serão :

```
>> num = [ 0 0 25 ]
>> den = [ 1 4 25 ]
>> impulse ( num , den )
```

```
num =
      0      0     25

den =
      1      4     25
```



Como a resposta a uma entrada em impulso corresponde à derivada temporal da resposta a uma entrada em degrau unitário, o ***Sobre-sinal Máximo (Overshoot)*** para a resposta a degrau unitário pode ser determinado a partir da correspondente resposta ao impulso, já que a área sob a curva de resposta ao impulso de Dirac desde $t=0$ até t_p (tempo do primeiro cruzamento com zero) é dada por :

$$1 + M_p$$

Onde M_p corresponde ao ***Sobre-sinal Máximo (Overshoot)*** para a resposta a degrau unitário.

4.4. RESPOSTA A ENTRADAS COM CONDIÇÕES INICIAIS NÃO NULAS.

A função ***initial*** gera a resposta no domínio do tempo de um sistema no espaço de estados com condições iniciais não nulas.

Considerando um sistema representado no espaço de estados com condições iniciais não nulas :

$$\begin{cases} \dot{x} = A x + B u \\ y = C x \end{cases}$$

$$x(0) = x_0$$

As linhas de comando que possibilitam a obtenção da resposta do sistema a este tipo particular de entrada são:

```
>> initial ( sys , x0 )
```

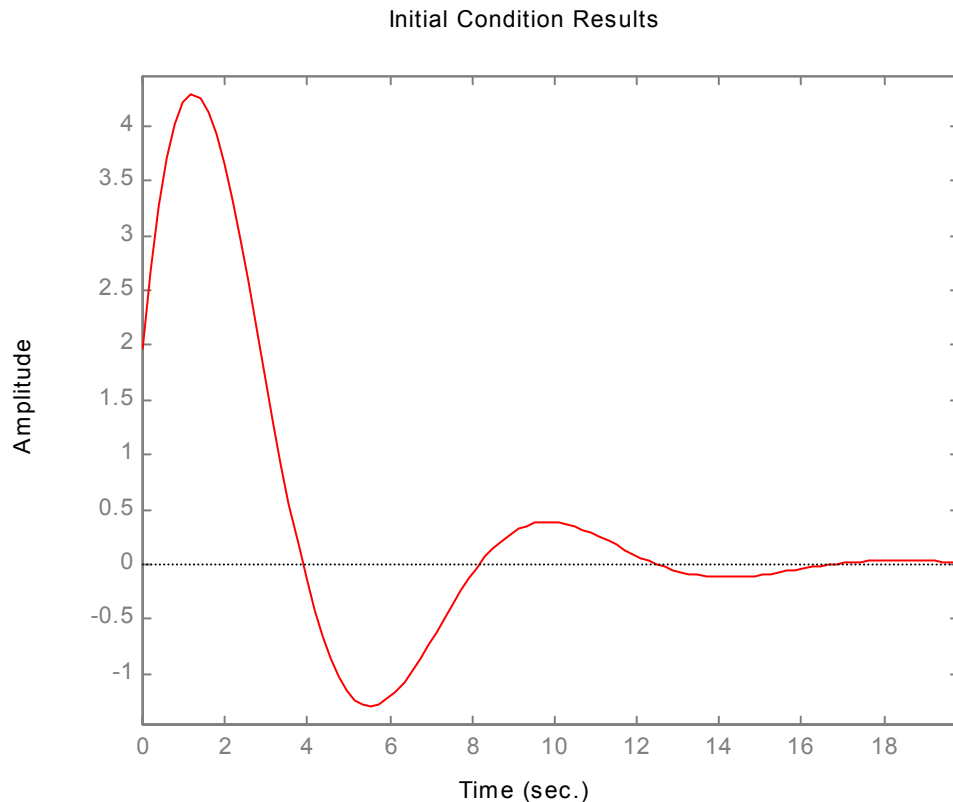
Considerando o seguinte sistema representado no espaço de estados :

$$\begin{cases} \dot{x} = \begin{bmatrix} -0.5572 & -0.7814 \\ 0.7814 & 0 \end{bmatrix} x \\ y = \begin{bmatrix} 1.9691 & 6.4493 \end{bmatrix} x \end{cases}$$

Sujeito às condições iniciais : $x(0) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

A resposta do sistema anterior no domínio do tempo consistirá na introdução das seguintes funções na *Janela de Comando* :

```
>> A = [ - 0.5572  - 0.7814 ; 0.7814  0 ]  
>> B = [ 0  0 ]'  
>> C = [ 1.9691  6.4493 ]  
>> D = [ 0 ]  
>> x0 = [ 1  0 ]'  
>> initial ( A , B , C , D , x0 )
```



NOTA : A aplicação desta função está sujeita aos mesmos procedimentos e restrições que as funções anteriores.

4.5. RESPOSTA A UMA ENTRADA ARBITRÁRIA

A função **lsim** (*linear simulation*) simula a resposta de um sistema no domínio do tempo a uma entrada arbitrária. Faz uso da mesma sintaxe que as funções anteriores, está sujeita à mesma metodologia de formulação e possui as mesmas limitações.

```
>> lsim ( sys , u , t )
```

A matriz representativa da função de entrada : **u** deverá possuir tantas colunas como a dimensão do vector representativo da amostragem de tempo : **t** (**length(t)**).

Além destas particularidades, a função **lsim** poderá gerar a resposta a entrada arbitrária de sistemas representados no espaço de estados com condições iniciais não nulas. Para tal, a sintaxe a utilizar será dada por :

```
>> lsim ( sys , u , t , x0 )
```

Considerando a função de transferência do sistema :

$$\frac{Y(s)}{R(s)} = \frac{num}{den} = \frac{s - 1}{s^2 + s + 5}$$

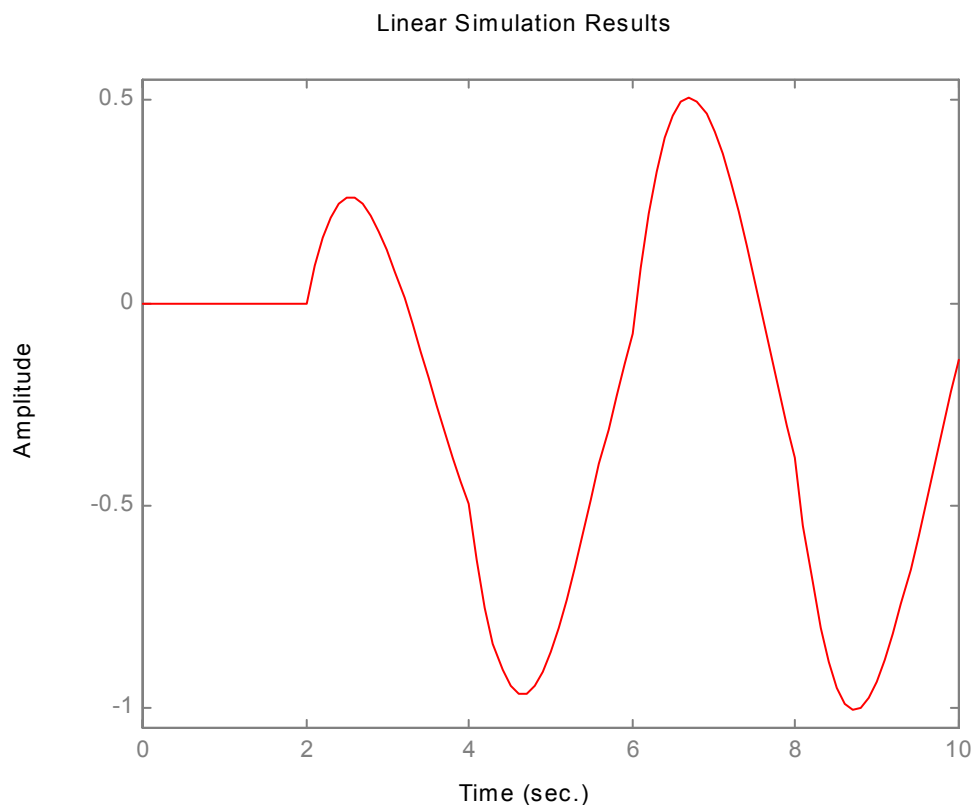
Pretende-se obter a resposta a uma entrada correspondente a uma onda quadrada com um período de 4 [seg].

Inicialmente, gera-se a onda quadrada com a função **gensig** (*generate signal*) considerando uma amostragem cada 0.1 [seg] durante 10 [seg] e posteriormente simula-se a resposta do sistema:

```
>> [u, t] = gensig ( 'square', 4, 10, 0.1 )
>> num = [ 1 - 1 ]
>> den = [ 1 1 2 ]
>> lsim ( num, den, u, t )
```

```
num =
      0      1     -1

den =
      1      1      2
```



NOTA : A função **gensig** gera um sinal periódico escalar **u** da classe *type* e período *tau*, para simulações no domínio do tempo através do uso da função **lsim**.

A função suporta as seguintes classes de sinal :

type = 'sin' : Onda sinusoidal.
type = 'square' : Onda quadrada.
type = 'pulse' : Impulso periódico.

```
>> [u, t] = gensig ( type, tau )
```

5. - ANÁLISE DO LUGAR GEOMÉTRICO DAS RAÍZES

A característica básica da resposta transiente de um sistema em malha fechada está intimamente relacionada com a localização dos pólos de malha fechada. Se o sistema tiver um ganho de malha variável, então a localização dos pólos de malha fechada depende do valor do ganho de malha escolhido. Assim, é importante durante o projecto de um sistema de controlo saber qual a movimentação dos pólos malha fechada no plano s em função do ganho. Em alguns sistemas, basta efectuar ajustes do ganho de modo a mover os pólos de malha fechada para os locais desejados, pelo que o projecto do sistema de controlo reside na selecção de um ganho apropriado.

Os pólos de malha fechada são as raízes da equação característica ($p(s) = 1 + G(s) \cdot H(s)$). O método do Lugar Geométrico das Raízes consiste na determinação das raízes da equação característica, colocadas num gráfico para todos os valores de um parâmetro do sistema. Note-se que o parâmetro usualmente variado é o ganho da função de transferência de malha aberta.

5.1. MAPEAMENTO DE PÓLOS E ZEROS DE SISTEMAS EM MALHA ABERTA

A função **pzmap** gera um gráfico com o mapeamento dos *Pólos* e dos *Zeros* de um sistema contínuo ou discreto. Os *Pólos* são representados por **X** e os *Zeros* representados por **O**.

Quando o comando é utilizado sem argumentos, esta função gera o mapeamento dos *Pólos* e dos *Zeros* do sistema na *Janela Gráfica*. Em caso contrário, apresenta duas colunas correspondentes aos *Pólos* : **p** e aos *Zeros* : **z**, sem gerar qualquer gráfico :

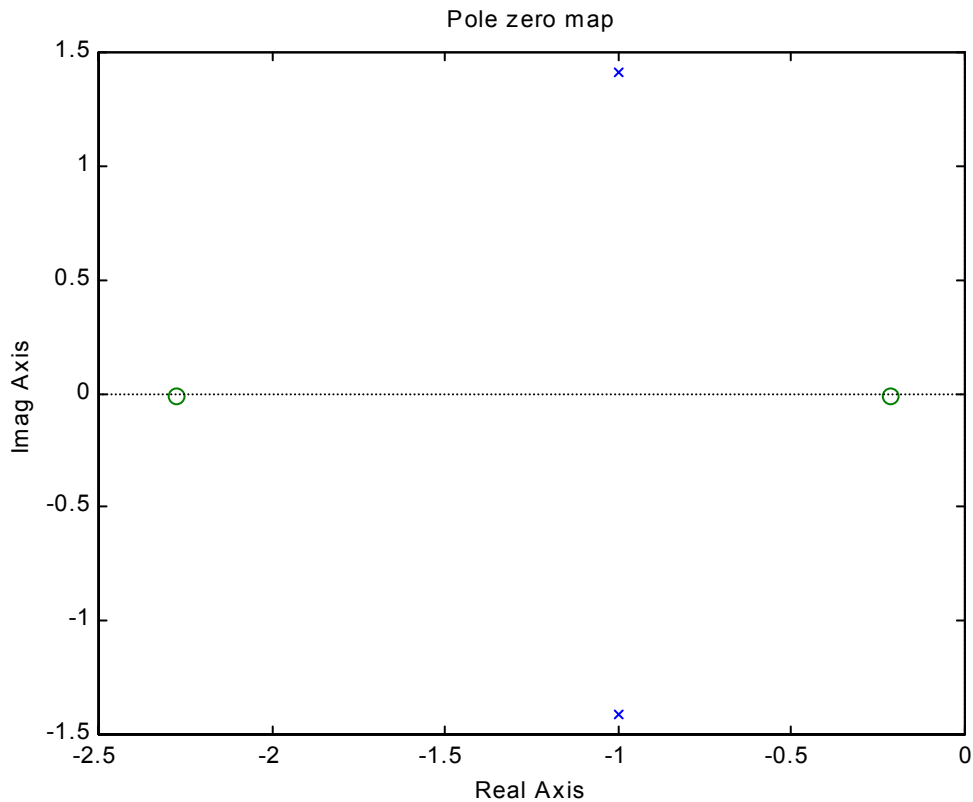
```
>> pzmap ( sys )           % Geração do gráfico com o mapeamento dos pólos e dos zeros do sistema.  
>> [ p , z ] = pzmap ( sys ) % Obtenção dos valores dos pólos e dos zeros.
```

Considerando a função de transferência de malha aberta do sistema, pretende-se obter o mapeamento dos *Pólos* e dos *Zeros* :

$$G(s) = \frac{num}{den} = \frac{2s^2 + 5s + 1}{s^2 + 2s + 3}$$

```
>> num = [ 2 5 1 ]  
>> den = [ 1 2 3 ]  
>> pzmap ( num , den )
```

```
num =  
      2      5      1  
  
den =  
      1      2      3
```

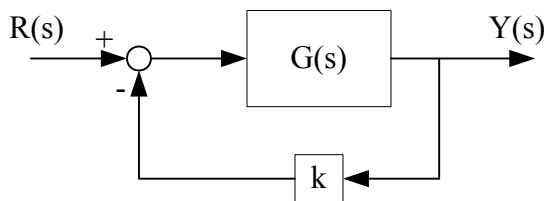


5.2. LUGAR GEOMÉTRICO DAS RAÍZES (ROOT LOCUS)

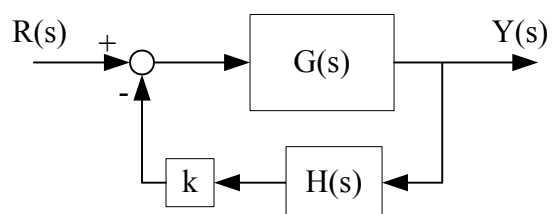
A função ***rlocus*** calcula o *Lugar Geométrico das Raízes* (Root Locus) para um sistema de malha aberta em função da variação do ganho k desde zero até infinito. O *Lugar Geométrico das Raízes* indica a trajetória dos *Pólos* de malha fechada do sistema em função do ganho de realimentação k (assumindo realimentação negativa).

O *Lugar Geométrico das Raízes* é usado no estudo dos efeitos da variação do ganho de realimentação na localização dos *Pólos* de malha fechada.

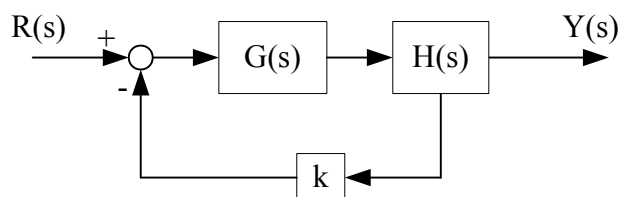
As diferentes topologias de funções de transferência de malha aberta de sistemas para implementação da função ***rlocus***, são dadas por :



F.T.M.A. : $\text{sys} = G(s)$



F.T.M.A. : $\text{sys} = H(s) G(s)$



F.T.M.A. : $\text{sys} = G(s) H(s)$

Se o sistema possui uma função de transferência de malha aberta dada por :

$$G(s) = \frac{num}{den} = \frac{n(s)}{d(s)}$$

Os *Pólos* de malha fechada do sistema são dados pelas raízes de: $d(s) + k n(s) = 0$

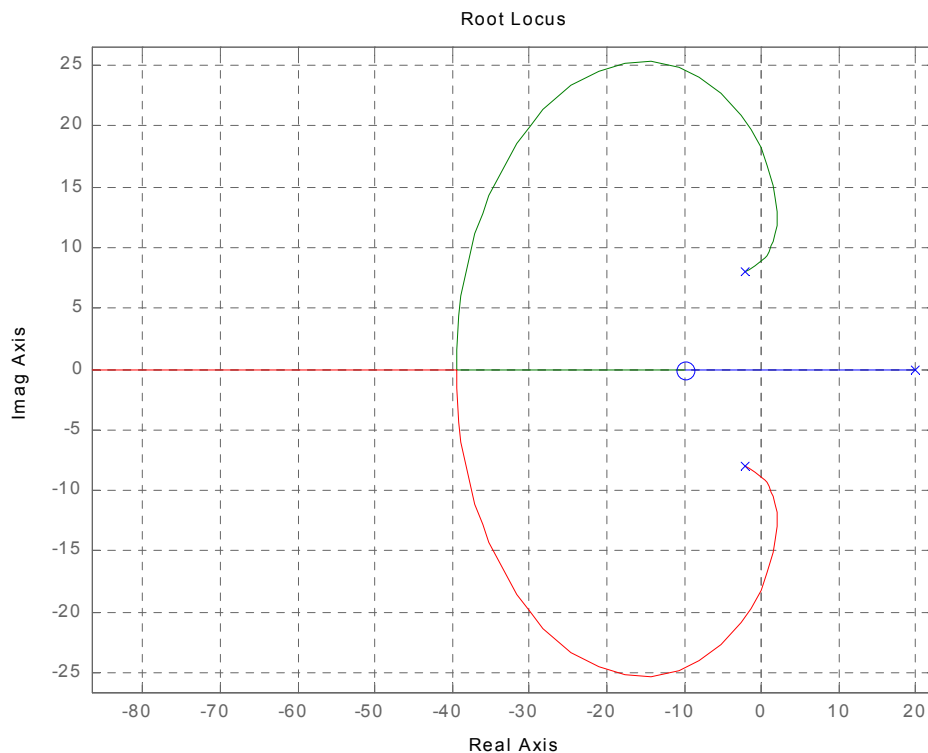
Quando o comando é utilizado sem argumentos, esta função gera o traçado da trajectória dos *Pólos* em função do ganho na *Janela Gráfica*. Em caso contrário, apresenta duas colunas correspondentes à localização das raízes complexas: *r* e respectivo *Ganho : k*, sem gerar qualquer gráfico :

```
>> rlocus ( sys )                % Geração do traçado da trajectória dos pólos.
>> rlocus ( sys , k )            % Geração do traçado da trajectória dos pólos para um determinado ganho.
>> [ r , k ] = rlocus ( sys )    % Obtenção da localização das raízes complexas e respectivo ganho.
>> r = rlocus ( sys , k )        % Obtenção da localização das raízes complexas para um ganho fixo.
```

No caso de se pretender determinar e visualizar a trajectória do sistema dado pela seguinte função de transferência de malha aberta :

$$G(s) = \frac{num}{den} = \frac{K(s + 10)^2}{(s - 20)(s^2 + 4s + 68)}$$

```
>> num = conv ( [ 1 10 ] , [ 1 10 ] )
>> den = conv ( [ 1 -20 ] , [ 1 4 68 ] )
>> rlocus ( num , den )
```



5.3. ANÁLISE DO VALOR DO GANHO DE REALIMENTAÇÃO DO SISTEMA

A função **rlocfind** utiliza a regra da magnitude do *Lugar Geométrico das Raízes* para determinar o *Ganho* para uma localização particular das raízes. Trata-se por defeito de uma função interactiva, já que permite ao utilizador seleccionar a localização das raízes no traçado do Root Locus para as quais pretende determinar o ganho de realimentação. No entanto, pode ser utilizada com argumentos de modo a calcular o ganho de realimentação para uma localização específica das raízes.

```
>> [ k, poles ] = rlocfind ( sys )      % Obtenção do ganho de realimentação interactivamente.  
>> [ k, poles ] = rlocfind ( sys , p )  % Obtenção do ganho de realimentação para uma localização específica.
```

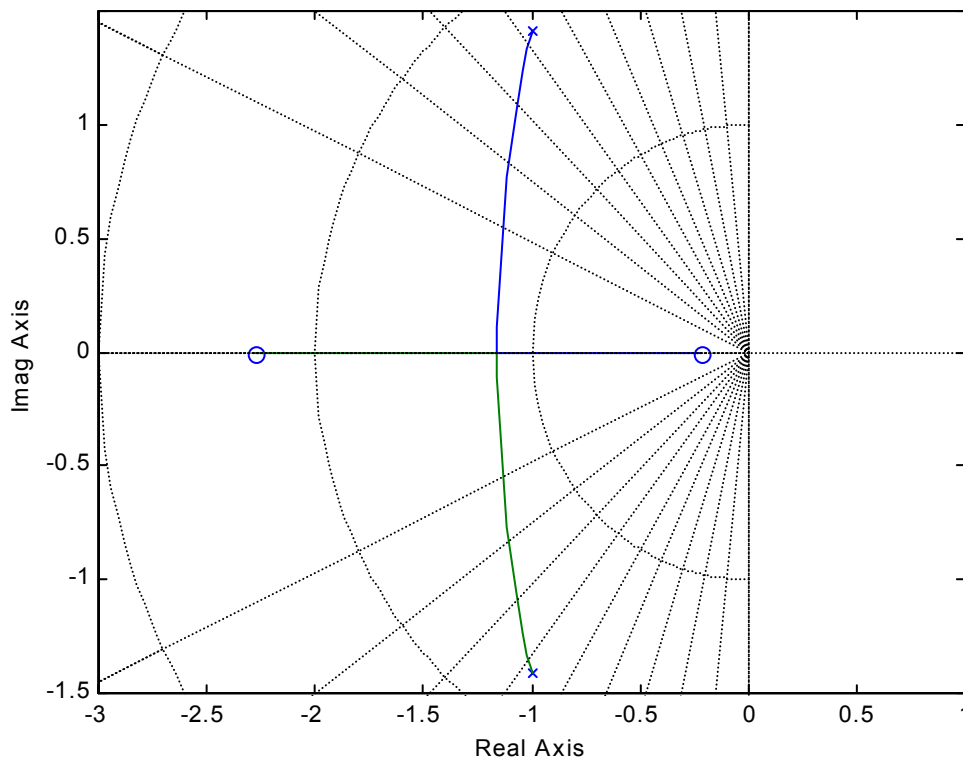
5.4. CURVAS DE ω_n CONSTANTE E DE ζ CONSTANTE

A função **sgrid** tem como objectivo gerar uma grelha no plano s de curvas de frequência natural constante e de coeficiente de amortecimento constante.

A grelha é gerada sobre o traçado do *Lugar Geométrico das Raízes* ou sobre o *Mapeamento dos Pólos e Zeros* obtido anteriormente, com um espaçamento de 0.1 desde 0 até 1 para as curvas de coeficiente de amortecimento constante, e com um espaçamento de 1 [rad/seg] desde 0 até 10 [rad/seg] para as curvas de frequência natural constante. Pode-se especificar como argumento as linhas de coeficiente de amortecimento constante : ζ (*zeta*) e as linhas frequência natural constante : ω_n que se pretendem visualizar.

```
>> sgrid  
>> sgrid ( zeta , wn )
```

As linhas de grelha no plano s de coeficiente de amortecimento constante e de frequência natural constante do sistema para o qual foi obtido traçado da trajectória dos *Pólos* (Root Locus) adicionando a função **sgrid** :



6. - RESPOSTA NO DOMÍNIO DA FREQUÊNCIA

Pelo termo resposta em frequência entende-se a resposta em regime estacionário de um sistema sujeito a uma entrada sinusoidal. Na análise e projecto de sistemas de controlo são utilizados os métodos convencionais de resposta em frequência, que consistem no estudo da resposta no domínio da frequência resultante em função da variação numa faixa de interesse, da frequência do sinal de entrada.

A *CONTROL SYSTEM Toolbox* do MATLAB possui um conjunto de funções que permitem analisar a resposta no domínio da frequência de um dado sistema. Os traçados de *Bode*, *Nyquist* e *Nichols* representam os métodos convencionais mencionados anteriormente, sabendo que em qualquer dos casos o sistema pode ser descrito por uma função de transferência ou por um modelo em espaço de estados.

6.1. DIAGRAMAS DE BODE

Uma função de transferência sinusoidal é uma função complexa da frequência ω , sendo caracterizada pelo seu *Módulo* e *Ângulo de Fase*. Assim, um *Diagrama de Bode* é definido por dois gráficos separados : Gráfico do logaritmo do *Módulo* de uma função de transferência sinusoidal e o Gráfico do *Ângulo de Fase*, sendo ambos construídos em função da frequência em escala logarítmica.

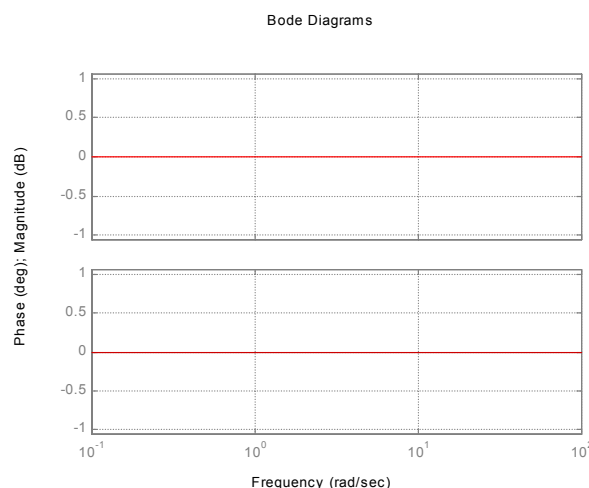
A principal vantagem em utilizar um gráfico com escalas logarítmicas reside na facilidade de traçado das curvas da resposta no domínio da frequência. As evoluções em frequências dos factores básicos que mais frequentemente se encontram numa função de transferência arbitrária : $G(j\omega)$ $H(j\omega)$ são evolutivos para as várias frequências de corte ω :

- **Ganho : k_B .**

$$\text{Módulo : } |k_B| = 20 \log(k) \quad [dB]$$

$$\text{Inclinação da recta : } 0 \text{ [dB/dec]}$$

$$\text{Ângulo de Fase : } \underline{k_B} = 0^\circ$$

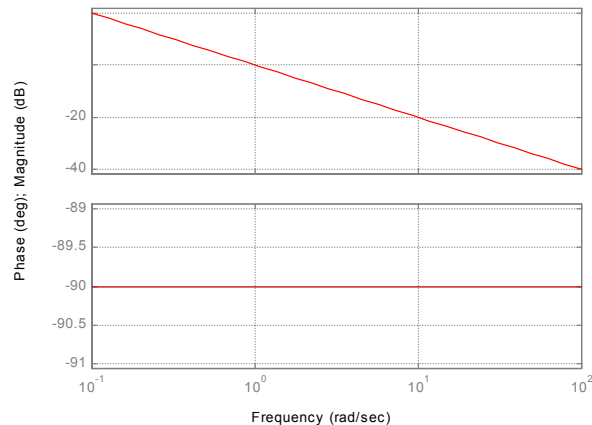


- **Factor integral (Pólos na Origem) : $(j\omega)^{-1}$.**

Módulo : $|G(j\omega)| = 20 \log \left| \frac{1}{j\omega} \right| = -20 \log(\omega) \quad [dB]$

Inclinação da recta : - 20 [dB/dec]

Ângulo de Fase : $\angle G(j\omega) = -90^\circ$



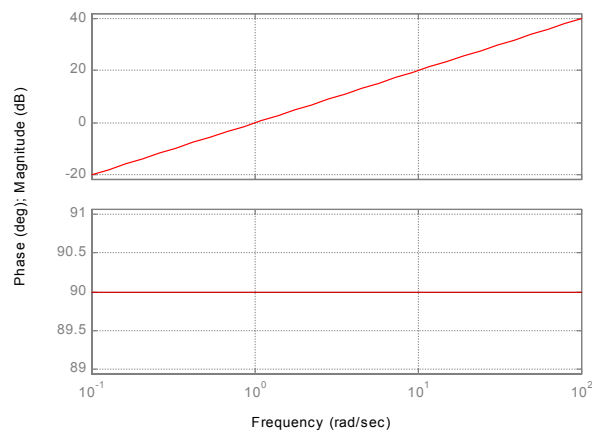
- **Factor derivativo (Zeros na Origem) : $(j\omega)$.**

Módulo : $|G(j\omega)| = 20 \log |j\omega| = 20 \log(\omega) \quad [dB]$

Inclinação da recta : 20 [dB/dec]

Ângulo de Fase : $\angle G(j\omega) = 90^\circ$

Bode Diagrams



NOTA : A representação destes factores nos Gráficos do logaritmo do *Módulo* e do *Ângulo de Fase*, acarreta uma inclinação da recta e uma variação do *Ângulo de Fase* com os valores especificados a partir da frequência de corte ω .

- **Factor de primeira ordem (Pólos reais) : $(1 + j\omega T)^{-1}$.**

Módulo : $|G(j\omega)| = 20 \log \left| \frac{1}{1 + j\omega T} \right| = -20 \log \left(\sqrt{1 + \omega^2 T^2} \right) \quad [dB]$

$$\omega \ll \frac{1}{T} \Rightarrow |G(j\omega)| = -20 \log \left(\sqrt{1 + \omega^2 T^2} \right) = -20 \log(1) = 0 [dB]$$

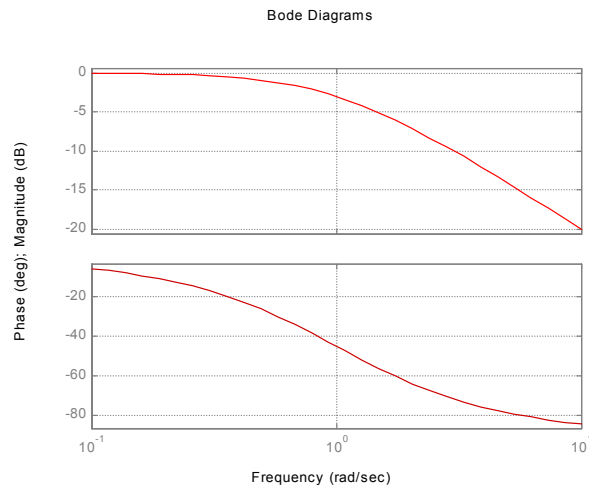
$$\omega \gg \frac{1}{T} \Rightarrow |G(j\omega)| = -20 \log \left(\sqrt{1 + \omega^2 T^2} \right) = -20 \log(\omega T) \quad [dB]$$

Inclinação da recta : -20 [dB/dec]

Ângulo de Fase : $\angle G(j\omega) = -\tan^{-1}(\omega T)$

$$\omega \ll \frac{1}{T} \Rightarrow \angle G(j\omega) = -\tan^{-1} \left(\frac{\omega T}{1} \right) = -\tan^{-1}(0) = 0^\circ$$

$$\omega \gg \frac{1}{T} \Rightarrow \angle G(j\omega) = -\tan^{-1} \left(\frac{\omega T}{1} \right) = -\tan^{-1}(\infty) = -90^\circ$$



- **Factor de primeira ordem (Zeros reais) : $(1 + j\omega T)$.**

Módulo : $|G(j\omega)| = 20 \log |1 + j\omega T| = 20 \log \left(\sqrt{1 + \omega^2 T^2} \right) \quad [dB]$

$$\omega \ll \frac{1}{T} \Rightarrow |G(j\omega)| = 20 \log \left(\sqrt{1 + \omega^2 T^2} \right) = 20 \log(1) = 0 [dB]$$

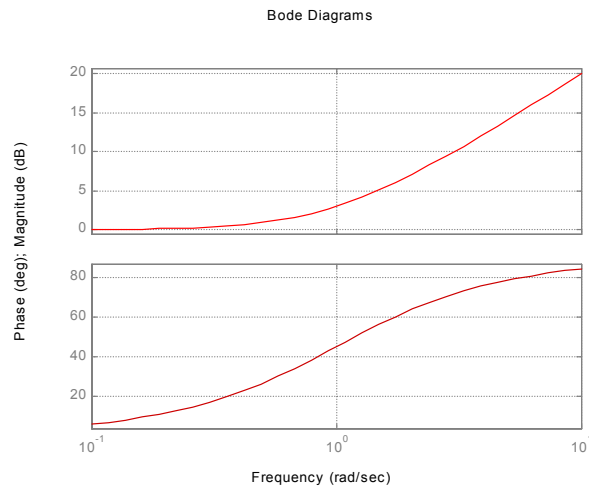
$$\omega \gg \frac{1}{T} \Rightarrow |G(j\omega)| = 20 \log \left(\sqrt{1 + \omega^2 T^2} \right) = 20 \log(\omega T) \quad [dB]$$

Inclinação da recta : 20 [dB/dec]

Ângulo de Fase : $\angle G(j\omega) = \tan^{-1}(\omega T)$

$$\omega \ll \frac{1}{T} \Rightarrow \angle G(j\omega) = \tan^{-1}\left(\frac{\omega T}{1}\right) = \tan^{-1}(0) = 0^\circ$$

$$\omega \gg \frac{1}{T} \Rightarrow \angle G(j\omega) = \tan^{-1}\left(\frac{\omega T}{1}\right) = \tan^{-1}(\infty) = 90^\circ$$



NOTA : A representação destes factores nos Gráficos do logaritmo do *Módulo* e do *Ângulo de Fase*, acarreta uma inclinação da recta e uma variação do *Ângulo de Fase* com os valores especificados a partir da frequência de corte ω .

- **Factor quadrático (Pólos complexos) :** $[1 + 2\zeta(j\omega/\omega_n) + (j\omega/\omega_n)^2]^{-1}$.

$$\begin{aligned} \text{Módulo : } \quad |G(j\omega)| &= 20 \log \left| \frac{1}{1 + 2\zeta \left(j \frac{\omega}{\omega_n} \right) + \left(j \frac{\omega}{\omega_n} \right)^2} \right| \\ &= -20 \log \left(\sqrt{\left(1 - \frac{\omega^2}{\omega_n^2} \right)^2 + \left(2\zeta \frac{\omega}{\omega_n} \right)^2} \right) \quad [dB] \end{aligned}$$

$$\omega \ll \omega_n \Rightarrow |G(j\omega)| = -20 \log(1) = 0 [dB]$$

$$\omega \gg \omega_n \Rightarrow |G(j\omega)| = -20 \log \left(\frac{\omega^2}{\omega_n^2} \right) = -40 \log \left(\frac{\omega}{\omega_n} \right) [dB]$$

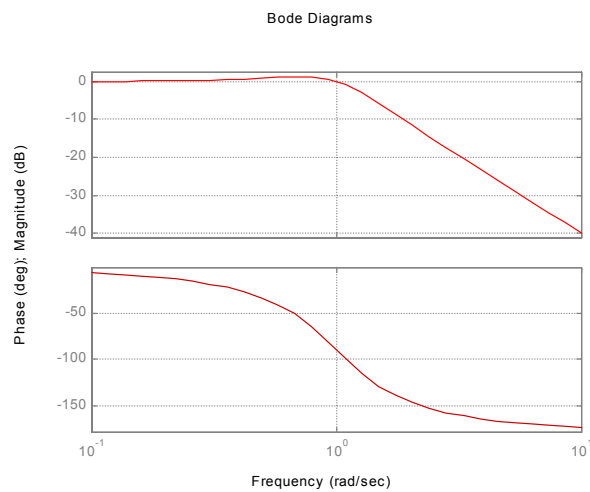
Inclinação da recta : - 40 [dB/dec]

$$\hat{\text{Ângulo de Fase :}} \quad \left| \underline{G(j\omega)} \right| = -\tan^{-1} \left(\frac{2\zeta \frac{\omega}{\omega_n}}{1 - \left(\frac{\omega}{\omega_n} \right)^2} \right)$$

$$\omega = 0 \Rightarrow \left| \underline{G(j\omega)} \right| = 0^\circ$$

$$\omega = \omega_n \Rightarrow \left| \underline{G(j\omega)} \right| = -90^\circ$$

$$\omega = \infty \Rightarrow \left| \underline{G(j\omega)} \right| = -180^\circ$$



- **Factor quadrático (Zeros complexos) : $[1 + 2\zeta(j\omega/\omega_n) + (j\omega/\omega_n)^2]$.**

$$\text{Módulo :} \quad \left| \underline{G(j\omega)} \right| = 20 \log \left| 1 + 2\zeta \left(j \frac{\omega}{\omega_n} \right) + \left(j \frac{\omega}{\omega_n} \right)^2 \right|$$

$$= 20 \log \left(\sqrt{\left(1 - \frac{\omega^2}{\omega_n^2} \right)^2 + \left(2\zeta \frac{\omega}{\omega_n} \right)^2} \right) [dB]$$

$$\omega \ll \omega_n \Rightarrow \left| \underline{G(j\omega)} \right| = 20 \log(1) = 0 [dB]$$

$$\omega \gg \omega_n \Rightarrow \left| \underline{G(j\omega)} \right| = 20 \log \left(\frac{\omega^2}{\omega_n^2} \right) = 40 \log \left(\frac{\omega}{\omega_n} \right) [dB]$$

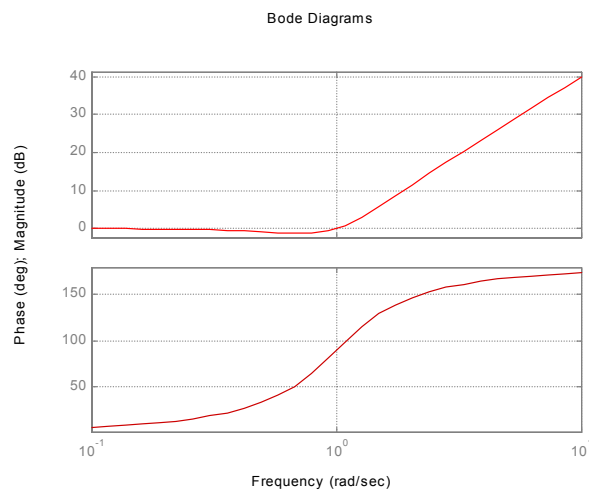
Inclinação da recta : 40 [dB/dec]

$$\hat{\text{Ângulo de Fase}} : \quad \angle G(j\omega) = \tan^{-1} \left(\frac{2\zeta \frac{\omega}{\omega_n}}{1 - \left(\frac{\omega}{\omega_n} \right)^2} \right)$$

$$\omega = 0 \Rightarrow \angle G(j\omega) = 0^\circ$$

$$\omega = \omega_n \Rightarrow \angle G(j\omega) = 90^\circ$$

$$\omega = \infty \Rightarrow \angle G(j\omega) = 180^\circ$$



- Frequência de Ressonância ω_r e Pico de Ressonância M_r .

Denomina-se por **Frequência de Ressonância**, a frequência em que $|G(s)|$ possui um valor de pico :

$$\omega_r = \omega_n \sqrt{1 - 2\zeta^2} \quad 0 \leq \zeta \leq 0.707$$

O valor de pico de $|G(s)|$, ou **Pico de Ressonância** é dado por :

$$M_r = \frac{1}{2\zeta \sqrt{1 - \zeta^2}} \quad 0 \leq \zeta \leq 0.707$$

$$M_r = 1 \quad \zeta \geq 0.707$$

NOTA : A representação destes factores nos Gráficos do logaritmo do *Módulo* e do *Ângulo de Fase*, acarreta uma inclinação da recta e uma variação do *Ângulo de Fase* com os valores especificados a partir da frequência de corte ω .

É necessário ter em consideração que a amplitude do *Pico de Ressonância* depende do valor do coeficiente de amortecimento, o que se vai reflectir em erros nas curvas assintóticas do gráfico do *Módulo*, bem como no gráfico do *Ângulo de Fase*.

6.1.1. Diagrama de Bode : Módulo e Ângulo de Fase

A função **bode** exibe o traçado de Bode (Gráficos de *Módulo* e de *Ângulo de Fase*) da resposta no domínio da frequência de um dado sistema pela avaliação da função de transferência : $G(s)$ no eixo imaginário : $s = j\omega$. Apenas são consideradas frequências positivas.

Para modelos em espaço de estados, a resposta em frequência é dada por :

$$D + C (j\omega I - A)^{-1} B, \quad \omega \geq 0$$

Quando o comando é utilizado sem argumentos, esta função gera o traçado do *Módulo* e do *Ângulo de Fase* da resposta em frequência. Os diagramas de Bode são utilizados para analisar propriedades do sistema, como sejam as Margem de Ganho, Margem de Fase, Largura de Banda, Estabilidade, etc.

A gama de frequências [rad/seg] é determinada automaticamente baseada nos *Pólos* e *Zeros* do sistema. As diversas sintaxes para a utilização desta função são dadas por :

```
>> bode ( sys )                % Geração do traçado do diagrama de Bode de Módulo e Ângulo de Fase.
>> bode ( sys , w )            % Geração do traçado de Bode com gama de frequências especificadas.
>> bode ( sys1 , sys2 , ..., sysN ) % Geração dos traçados de Bode sobrepostos de diversos sistemas.
```

Quando o comando é usado com os argumentos, *Módulo* : **mod**, *Ângulo de Fase* : **fase** [°] e *Frequência* : **w**, são apresentados os valores correspondentes sem que seja exibido o traçado do diagrama de Bode :

```
>> [ mod , fase , w ] = bode ( sys )    % Obtenção dos valores do Módulo, Ângulo de Fase e Frequência.
>> [ mod , fase ] = bode ( sys , w )    % Obtenção dos valores do Módulo, Ângulo de Fase para a gama de frequências especificadas.
```

Considerando o seguinte sistema dado pela função de transferência em malha aberta, pretende-se obter o traçados dos diagramas de Bode de *Módulo* e *Ângulo de Fase* :

$$G(s)H(s) = \frac{num}{den} = \frac{s^2 + 0.1s + 7.5}{s^4 + 0.12s^3 + 9s^2}$$

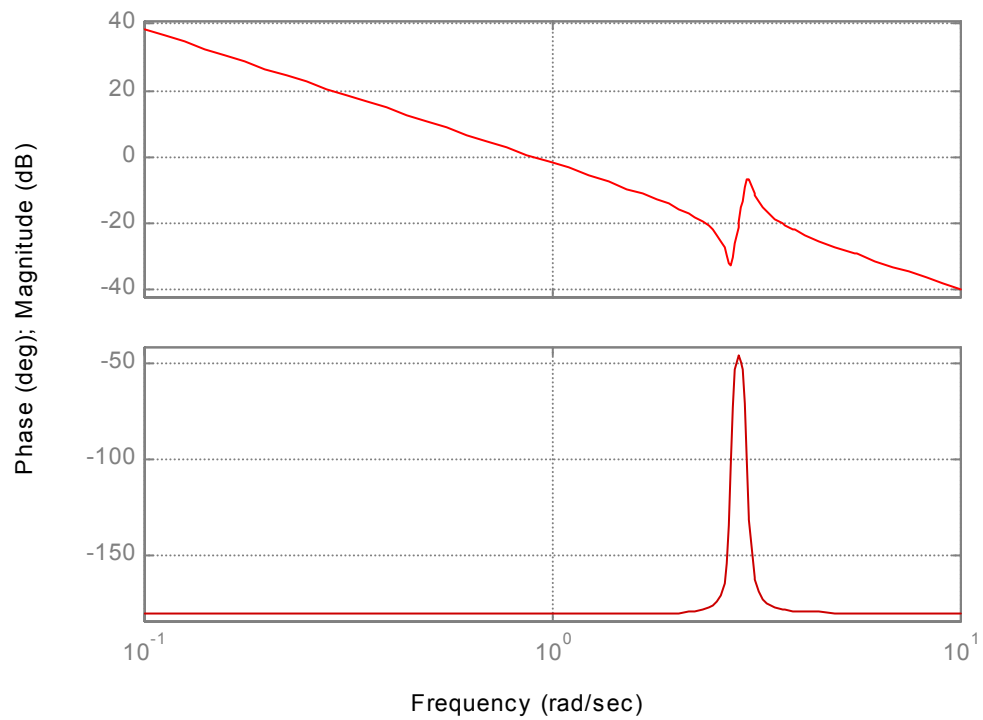
Este sistema é composto por uma função quadrática no **numerador** (*Zeros* complexos) e no **denominador** apresenta o quociente entre dois factores integrais de primeira ordem (*Pólos* reais) e uma função quadrática (*Pólos* complexos).

```
>> num = [ 1 0.1 7.5 ]
>> den = [ 1 0.12 9 0 0 ]
>> bode ( num , den )
```

```
num =
      0      0      1      0.1      7.5

den =
      1      0.12      9      0      0
```

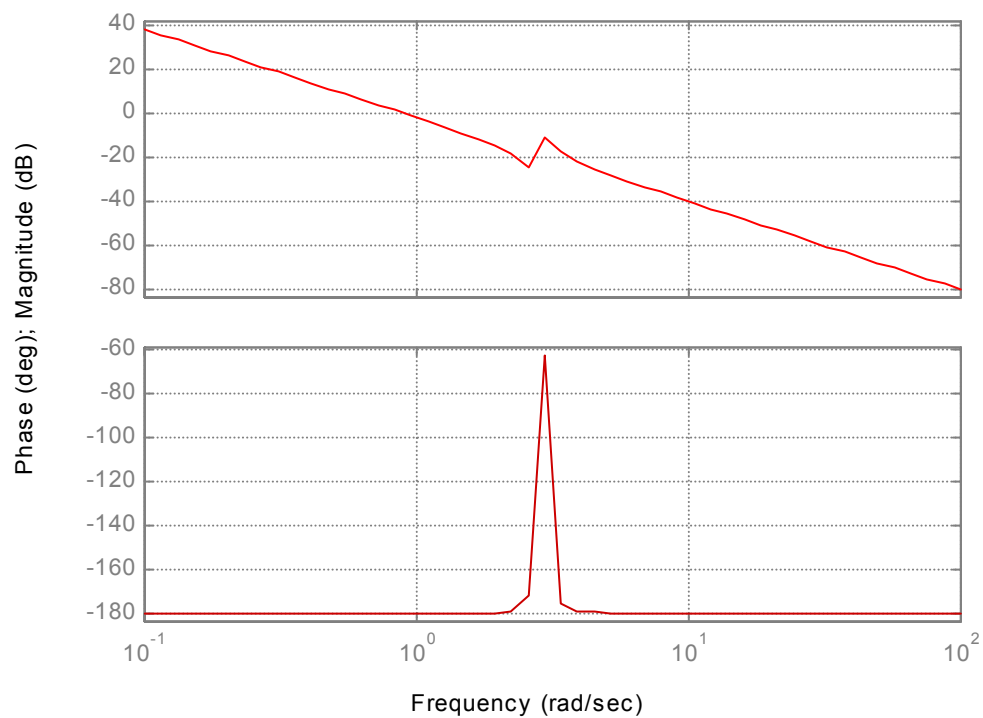
Bode Diagrams



Para obter o traçado da resposta do sistema no domínio da frequência para uma gama mais alargada de frequência, por exemplo desde 0.1 até 100 [rad/seg], basta introduzir o comando :

```
>> w = logspace ( 0.1 , 100 )
>> bode ( num , den , w )
```

Bode Diagrams



6.1.2. Margem de Ganho e Margem de Fase

Na análise da resposta de sistemas no domínio da frequência, as *Margens de Ganho* e de *Fase* permitem estudar a estabilidade do sistema.

Margem de Fase – Atraso de fase adicional na frequência de cruzamento do ganho, necessário para levar o sistema ao limiar da instabilidade. A frequência de cruzamento do ganho é a frequência na qual $G(j\omega)$, o módulo da função de transferência de malha aberta, é unitário. A *Margem de Fase* : γ é 180° mais o *Ângulo de Fase* : ϕ da função de transferência de malha aberta na frequência de cruzamento de ganho : $\gamma = 180 + \phi$

Para um sistema de fase mínima (Todos os *Pólos* e *Zeros* no semiplano s da esquerda) ser estável, a *Margem de Fase* deve ser positiva.

Margem de Ganho – Inverso do módulo $|G(j\omega)|$ na frequência onde o *Ângulo de Fase* : ϕ é -180° . Definindo a frequência de cruzamento de fase : ω_r , como sendo a frequência na qual o *Ângulo de Fase* : ϕ da função de transferência de malha aberta é igual a -180° , a Margem de Ganho resulta : $K_g = -20 \log |G(j\omega_r)|$ [dB]

Uma *Margem de Ganho* positiva significa que o sistema é estável, e o simétrico que o sistema é instável. Para um sistema de fase mínima estável, a *Margem de Ganho* indica quanto o ganho pode ser aumentado antes que o sistema se torne instável. Para um sistema instável, a *Margem de Ganho* é indicativa de quanto o ganho deve ser diminuído para tornar o sistema estável.

A *Margem de Ganho* e a *Margem de Fase* e respectivas frequências de cruzamento de um sistema, são calculadas a partir da função de transferência em malha aberta pela função **margin**. Indicam a estabilidade relativa do sistema em malha fechada.

Quando o comando é usado sem argumentos, exibe os diagramas de Bode de *Módulo* e *Ângulo de Fase* com as referidas margens. As sintaxes possíveis para a utilização da função são dadas por :

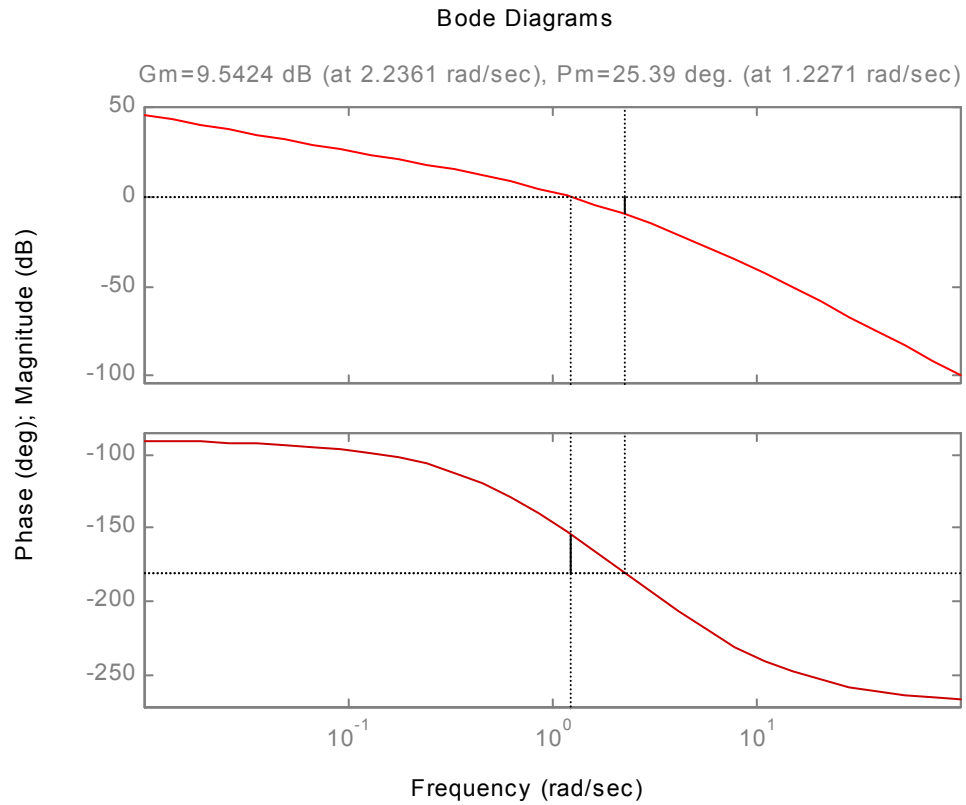
```
>> margin ( mod , fase , w )                                % Exibição do diagrama de Bode com as
                                                             Margens.
>> [ Gm , Pm , Wcg , Wcp ] = margin ( mod , fase , w )      % Obtenção das Margens e frequências de
                                                             cruzamento.
```

Considerando o seguinte sistema dado pela função de transferência em malha aberta, pretende-se determinar a estabilidade do sistema de malha fechada para $K = 10$ e $K = 100$ através da análise da *Margem de Ganho* e da *Margem de Fase* :

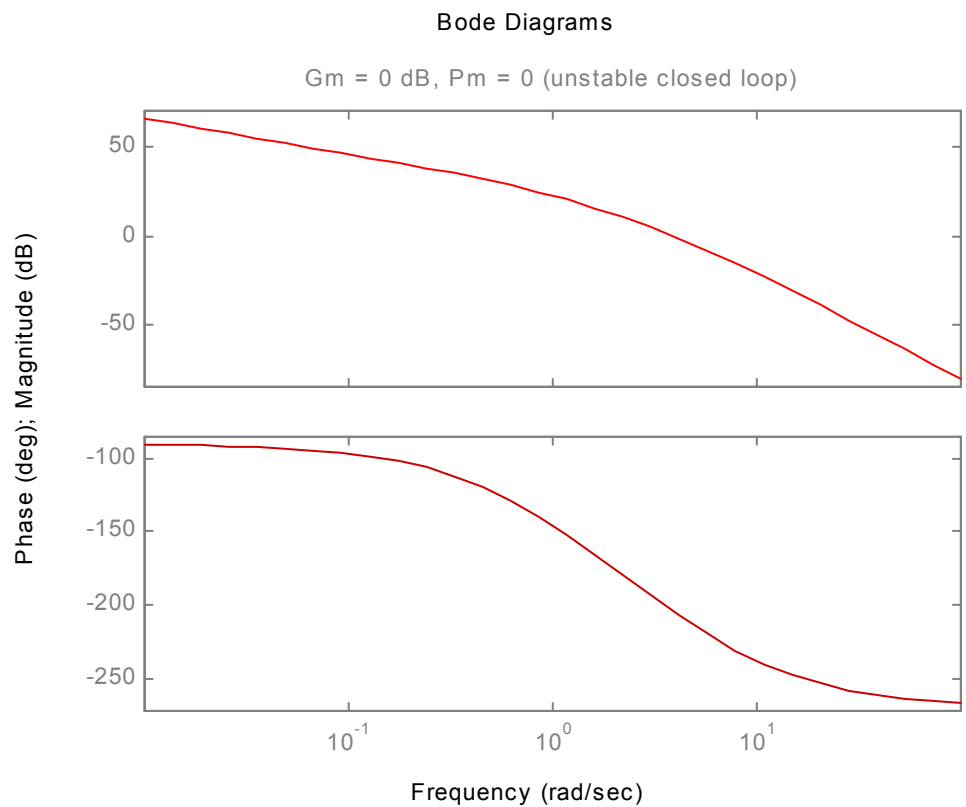
$$G(s)H(s) = \frac{num}{den} = \frac{K}{s(s+1)(s+5)}$$

```
>> % Para K = 10
>> num = [ 10 ]
>> den = [ 1 6 5 0 ]
>> [ mod , fase , w ] = bode ( num , den )
>> margin ( mod , fase , w )

>> % Para K = 100
>> num = [ 100 ]
>> [ mod , fase , w ] = bode ( num , den )
>> margin ( mod , fase , w )
```



$K = 10$: Margem de Ganho e Margem de Fase positivas : Sistema Estável.



$K = 100$: Margem de Ganho e Margem de Fase negativas : Sistema instável.

6.2. DIAGRAMA DE NYQUIST

O critério de estabilidade de Nyquist permite investigar a estabilidade absoluta e a estabilidade relativa de sistemas lineares de malha fechada a partir do conhecimento das características da sua resposta no domínio do tempo em malha aberta.

Considerando o seguinte sistema representado pela sua função de transferência de malha fechada :

$$\frac{Y(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

Para que o sistema seja estável, todas as raízes da equação característica : $1 + G(s)H(s) = 0$, deverão permanecer no semiplano s da esquerda.

A vantagem do critério de Nyquist reside na particularidade do sistema de malha fechada poder ser avaliado quanto à estabilidade absoluta através da análise gráfica das curvas da resposta no domínio da frequência em malha aberta.

O critério de Nyquist pode ser expresso por :

$$Z = N + P$$

Z : Número de Zeros de $1 + G(s)H(s)$ no semiplano direito do plano s .

N : Número de circulações em torno do ponto $-1 + j0$, no sentido horário.

P : Número de Pólos de $G(s)H(s)$ no semiplano direito do plano s .

Isto é, para que um sistema de controlo seja estável sabendo que:

$P \neq 0$: É necessário impor : $Z = 0$ ou $N = -P$.

$P = 0$: É necessário impor : $Z = N$.

A função **Nyquist** corresponde à análise e exibição das curvas da resposta do domínio da frequência do sistema em malha fechada, através da função de transferência de malha aberta (ou pelo modelo no espaço de estados). As curvas da resposta do sistema no domínio da frequência permitem estudar a *Margem de Ganho* e *Margem de Fase* e a *Estabilidade*.

Quando o comando é usado com argumentos calcula as partes real : **re** e imaginária : **im** da resposta em frequência. As sintaxes possíveis para a utilização da função são dadas por :

>> <i>nyquist (num , den)</i>	% Exibição do traçado de Nyquist.
>> <i>nyquist (num , den , w)</i>	% Exibição do traçado de Nyquist para a gama de frequências especificada.
>> <i>[re , im , w] = nyquist (num , den)</i>	% Obtenção das partes Real e Imaginária da resposta no domínio da frequência.

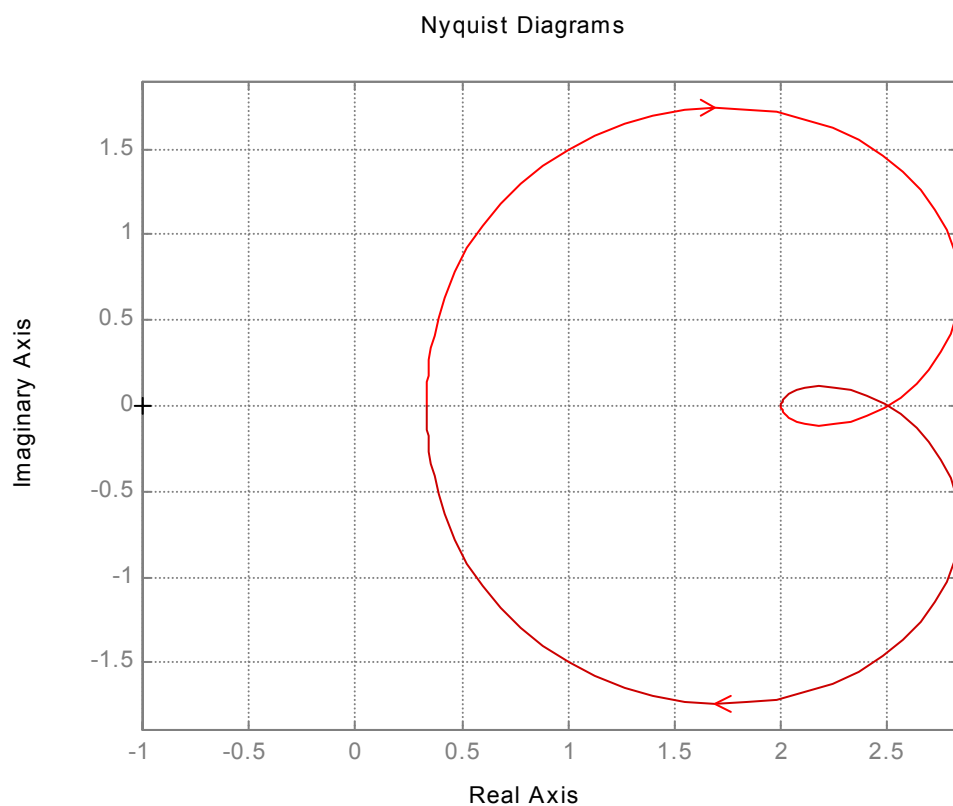
Considerando o seguinte sistema dado pela função de transferência em malha aberta, pretende-se obter o traçado de Nyquist da resposta no domínio da frequência do sistema de malha fechada :

$$G(s)H(s) = \frac{num}{den} = \frac{2s^2 + 5s + 1}{s^2 + 2s + 3}$$

```
>> num = [ 2 5 1 ]
>> den = [ 1 2 3 ]
>> nyquist ( num , den )
```

```
num =
      2      5      1

den =
      1      2      3
```



6.3. DIAGRAMA DE NICHOLS

A função **Nichols** calcula a resposta de um sistema no domínio da frequência e exibe o traçado nas coordenadas de Nichols. Esta função é útil para analisar as propriedades de sistema em malha aberta e em malha fechada.

Quando o comando é usado sem argumentos produz a exibição do traçado de Nichols.

A função **ngrid** gera as linhas de grelha para o traçado de Nichols que correspondem às linhas de *Módulo* e *Ângulo de Fase* constantes da função de transferência de malha aberta.

A grelha e a gama de frequências é determinada automaticamente pela análise dos *Pólos* e *Zeros* do sistema.

```
>> nichols ( sys ) % Geração do traçado de Nichols.
>> nichols ( sys, w ) % Geração do traçado de Nichols com gama de frequências especificada.
>> nichols ( sys1 , sys2,..., sysN ) % Geração dos traçados de Nichols sobrepostos de diversos sistemas.
```

Quando o comando é usado com os argumentos, *Módulo* : **mod**, *Ângulo de Fase* : **fase** [°] e *Frequência* : **w**, são apresentados os valores correspondentes sem que seja exibido o traçado de Nichols :

```
>> [ mod , fase, w ] = nichols ( sys ) % Obtenção dos valores do Módulo, Ângulo de Fase e Frequência.
>> [ mod , fase ] = nichols ( sys , w ) % Obtenção dos valores do Módulo, Ângulo de Fase para a gama de frequências especificada.
```

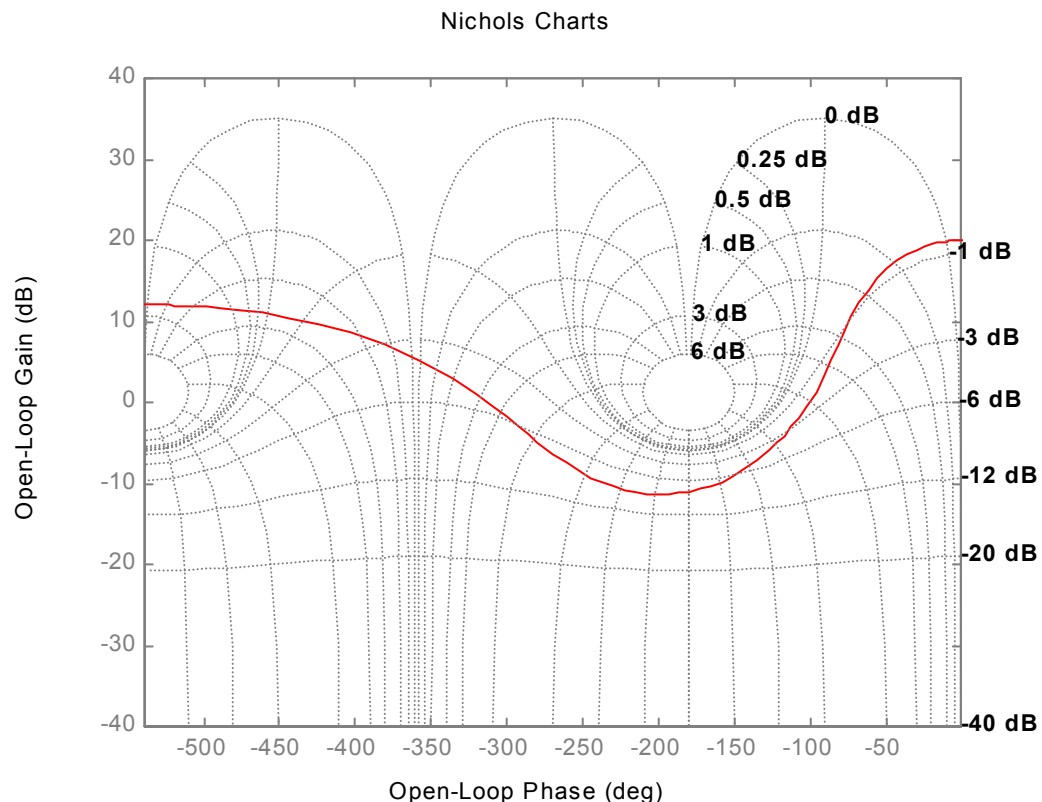
Considerando o seguinte sistema dado pela função de transferência em malha aberta, pretende-se obter o traçados de Nichols :

$$G(s)H(s) = \frac{-4s^4 + 48s^3 - 18s^2 + 250s + 600}{s^4 + 30s^3 + 282s^2 + 525s + 60}$$

```
>> num = [ -4 48 -18 250 600 ]
>> den = [ 1 30 282 525 60 ]
>> nichols ( num , den )
>> ngrid
```

```
num =
    -4    48   -18   250   600

den =
     1    30   282   525    60
```



REFERÊNCIAS BIBLIOGRÁFICAS

- [1] *The Student Edition of MATLAB*, The Math Works Inc., Prentice Hall, 1992.
- [2] Ogata, Katsuhiko - *Solving Control Engineering Problems with MATLAB*, Prentice Hall, 1994.
- [3] Ogata, Katsuhiko - *Engenharia de Controle Moderno – 2ª Edição*, Prentice Hall.