EA722 - Laboratório de Controle e Servomecanismos

Fundamentos de Realimentação: Simulação dos Modelos do ECP em Malha Fechada

Faculdade de Engenharia Elétrica e de Computação
Universidade Estadual de Campinas

2º Semestre 2017

Obs.: Nas configurações adotadas nesta experiência

I. Modelos Lineares

- Emulador Industrial

$$J_d^* \ddot{\theta}_1 + c_d^* \dot{\theta}_1 = T_d$$

$$G_p(s) = \frac{k_{hw}}{s(J_d^* s + c_d^*)}$$

- Sistema Retilíneo

$$m_1 \ddot{x}_1 + c_1 \dot{x}_1 + k_1 x_1 = F(t)$$

$$G_p(s) = \frac{k_{hw}}{m_1 s^2 + c_1 s + k_1}, \quad m_1 = m_{c1} + m_w$$

- Sistema Torcional

$$J_1 \ddot{x}_1 + c_1 \dot{x}_1 + k_1 x_1 = T(t)$$

$$G_p(s) = \frac{k_{hw}}{J_1 s^2 + c_1 s + k_1}, \qquad J_1 = J_{d1} + J_w$$

- Pêndulo com Haste Rotacional Travada

$$m_1 \ddot{x}_1 + c_1 \dot{x}_1 = F(t)$$

$$G_p(s) = \frac{X(s)}{F(s)} = \frac{k_{hw}}{m_1^* s^2 + c_1 s}$$

II. Modelos Não-Lineares

- Pêndulo Invertido

- Duas Hastes Livres:

$$\bar{J}\left(\ddot{x}+\frac{c_1}{m_1}\dot{x}\right)-J^*x\dot{\theta}^2-2m_1\ell_o x\dot{x}\dot{\theta}+(m_2\ell_o\ell_c-\bar{J})g\,\mathrm{sen}\,\theta+$$

$$+m_1\ell_o gx\cos\theta=\frac{J^*}{m_1}F(t)$$

$$\bar{J}\,\ddot{\theta}+c_r\dot{\theta}+2m_1x\dot{x}\dot{\theta}+m_1\ell_o x\dot{\theta}^2-m_2\ell_c g\,\mathrm{sen}\,\theta-m_1gx\cos\theta=-\ell_o F(t)$$

- Duas Hastes Livres. Modelo Linearizado (Taylor):

$$\bar{J}\left(\ddot{x}+\frac{c_1}{m_1}\dot{x}\right)+m_1\ell_o gx+(m_2\ell_o\ell_c-\bar{J})g\theta=\frac{J^*}{m_1}F(t)$$

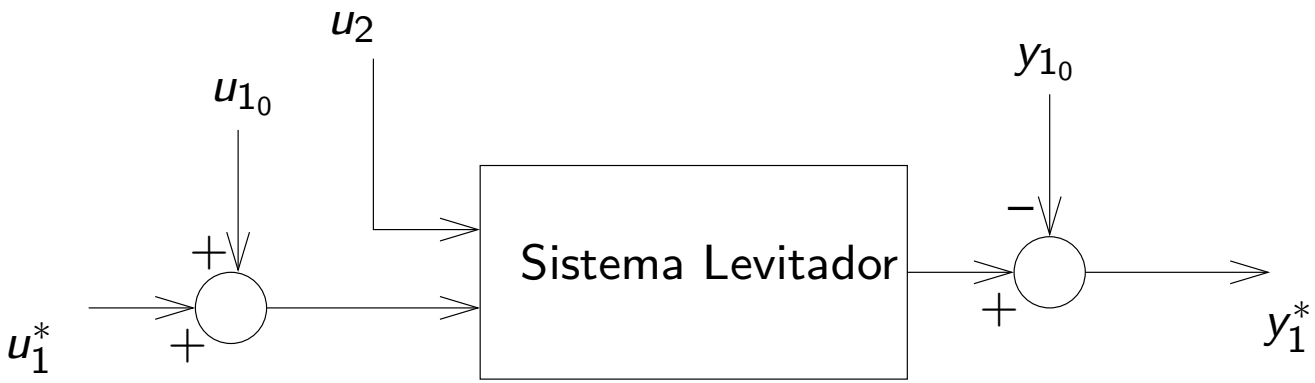
$$\bar{J}\,\ddot{\theta}+c_r\dot{\theta}-m_1gx-m_2\ell_c g\theta=-\ell_o F(t)$$

- Levitador Magnético

$$m_1\ddot{y}_1 + c_1\dot{y}_1 = \frac{u_1}{a(k_s y_1 + b)^4} - m_1 g$$

escolhe-se o ponto de operação y_{1_0} (distância da bobina), obtendo-se a corrente u_{1_0} necessária:

$$u_{1_0} = a\, m_1\, g\, (k_s y_{1_0} + b)^4$$



Sistema levitador com ajuste de operação no ponto de equilíbrio y_{1_0}. Variáveis: u_1^* entrada incremental, y_1^* saída incremental.

- Levitador Magnético. Modelo Linearizado (Taylor):

$$m_1\ddot{y}_1 + c_1\dot{y}_1 + k_1 = k_{u_1}u_1$$

$$G_p(s) = \frac{Y_1(s)}{U_1(s)} = \frac{k_{u_1}}{m_1 s^2 + c_1 s + k_1}$$

“Viscosidade Adicional” I

- “Viscosidade Adicional” via Realimentação de velocidade

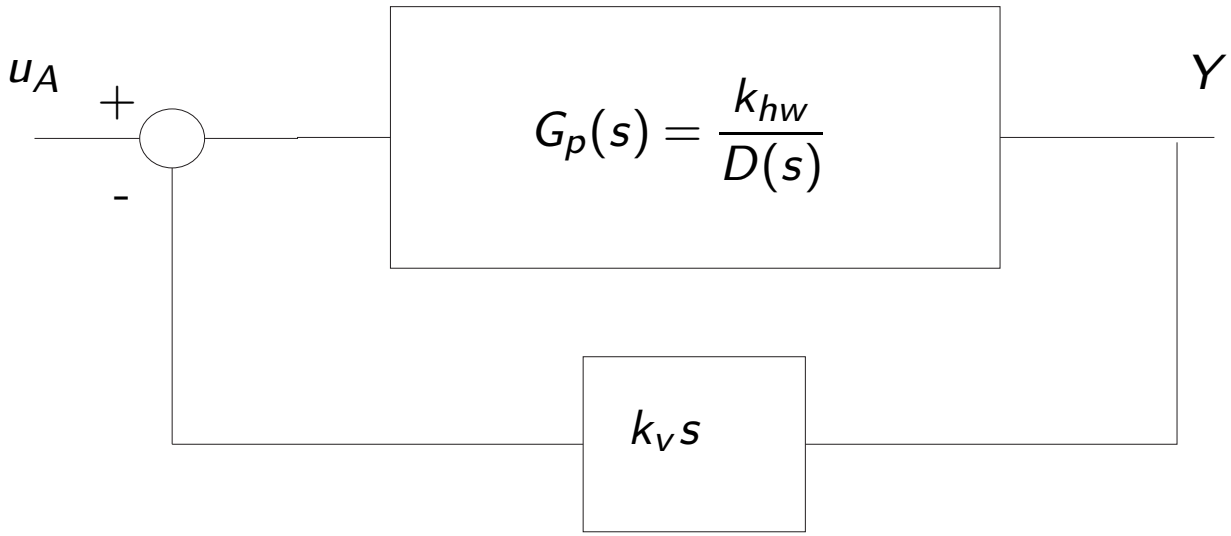


Fig 1: Realimentação de Velocidade

Os Diagramas das Figuras 1. e 2. são equivalentes.

“Viscosidade Adicional” II

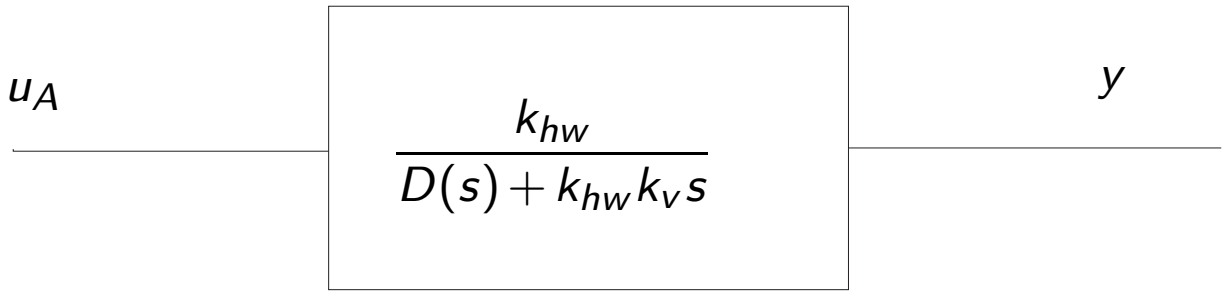
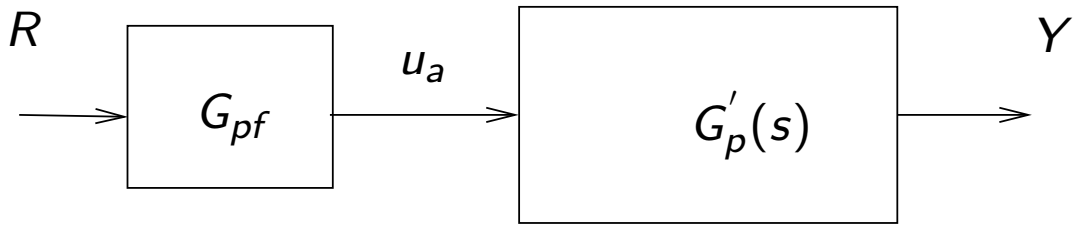


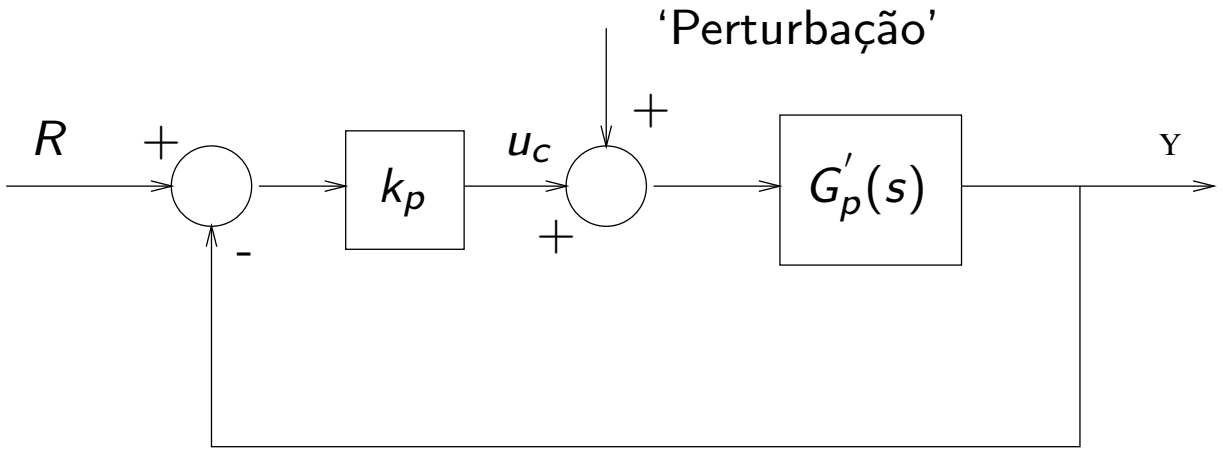
Fig 2: Sistema com Viscosidade Adicional

- Observe que a realimentação de velocidade acrescenta à equação característica um termo que corresponde a uma força ou torque de atrito viscoso.

Malha Aberta

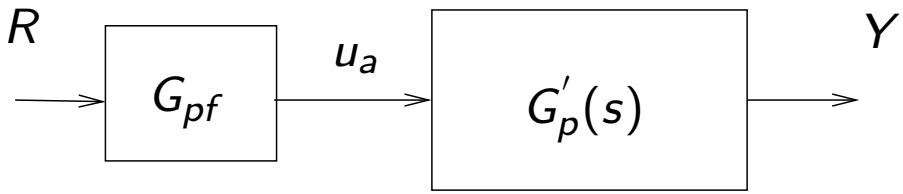


Malha Fechada: Controle Proporcional

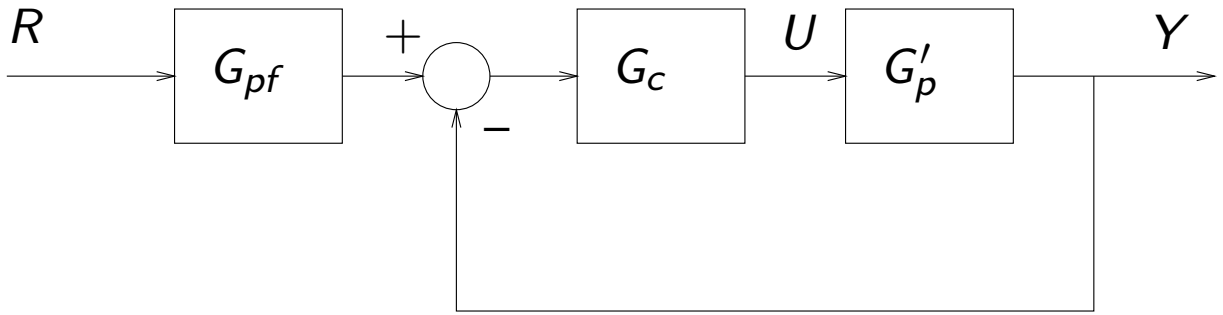


Uso do Pré Filtro I

Malha aberta:



Malha fechada:



- $G_{pf}(s)$ - função de transferência do pré-filtro

Uso do Pré Filtro II

- Retilíneo e torcional:

$$G_{pf}(s) = k_f \quad \text{(ganho)}$$

• Usado em malha fechada para obter erro de regime nulo (em malha aberta, $G_{pf}(s) = 1$)

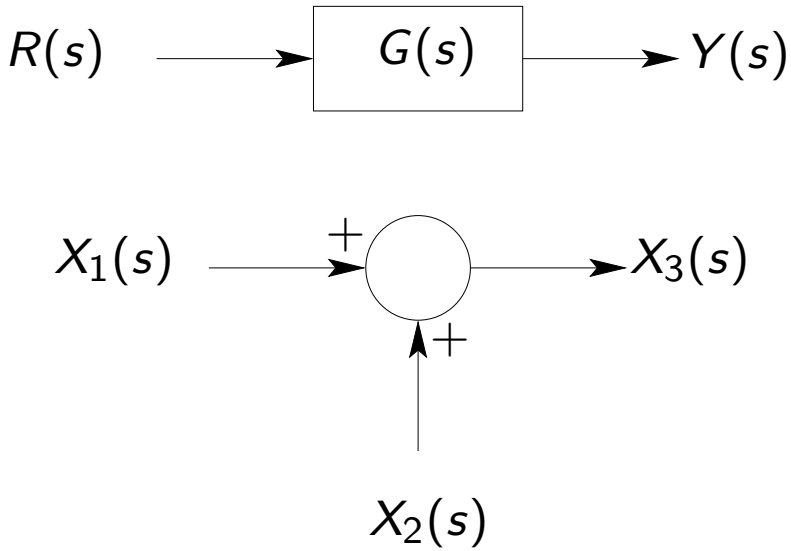
- Emulador e pêndulo:

$$G_{c1}(s) = \frac{s}{1 + 0.01s}$$

• Usado em malha aberta para cancelar pólo da planta na origem (em malha fechada, $G_{pf}(s) = 1$)

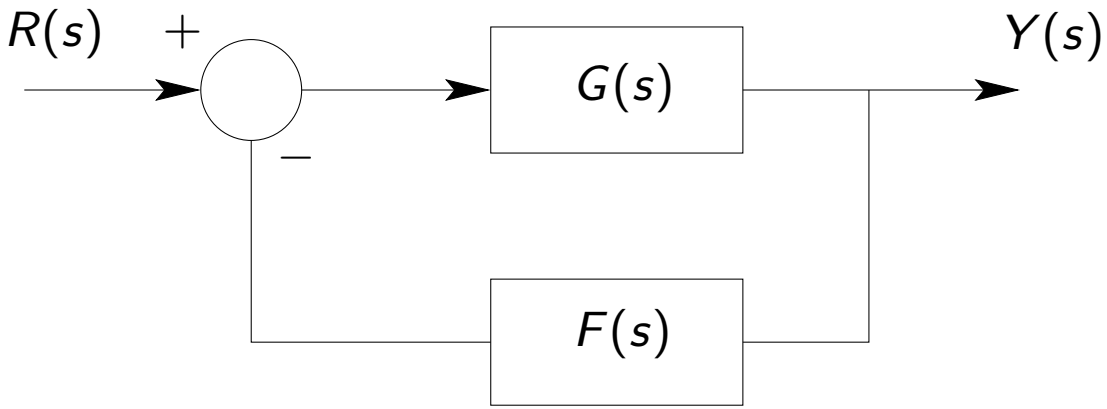
Diagrama de blocos

Regras básicas para simplificar diagramas de blocos:



- $Y(s) = G(s)R(s)$
- $X_3(s) = X_1(s) + X_2(s)$

Exemplos



- No matlab, a função de transferência que relaciona a saída $Y(s)$ e a entrada $R(s)$, isto é, $H(s) = Y(s)/R(s)$ pode ser encontrada por meio do comando: `H = feedback(G,F)`. Observação: é assumida realimentação negativa.

Erro em regime

Considere o sistema de controle em malha aberta do slide anterior. Aplicando um degrau unitário na entrada, isto é, $R(s) = 1/s$, tem-se que saída $Y(s)$ é dada por

$$Y(s) = C(s)G(s)/s$$

Definindo o sinal de erro como a diferença entra a saída e a entrada, tem-se

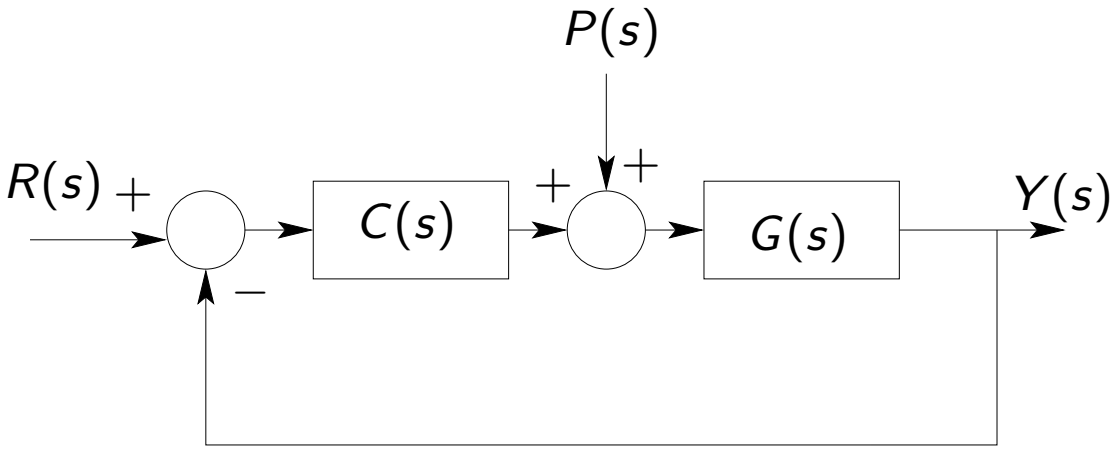
$$E(s) = Y(s) - R(s) = \frac{1}{s}(C(s)G(s) - 1)$$

Assumindo que a dinâmica de $C(s)G(s)$ é estável, ou seja, para qualquer entrada limitada, a saída é limitada, o sinal de erro pode ser avaliado (após a passagem dos transitórios) por meio do Teorema do Valor Final:

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} C(s)G(s) - 1$$

Neste caso nota-se que o erro (em regime) será nulo se $C(0)G(0) = 1$, ou seja, se o ganho de frequência zero (ou ganho DC) for unitário.
No matlab o ganho DC de uma função de transferência pode ser computado por meio do comando: dcgain(·)

Projetos de controle



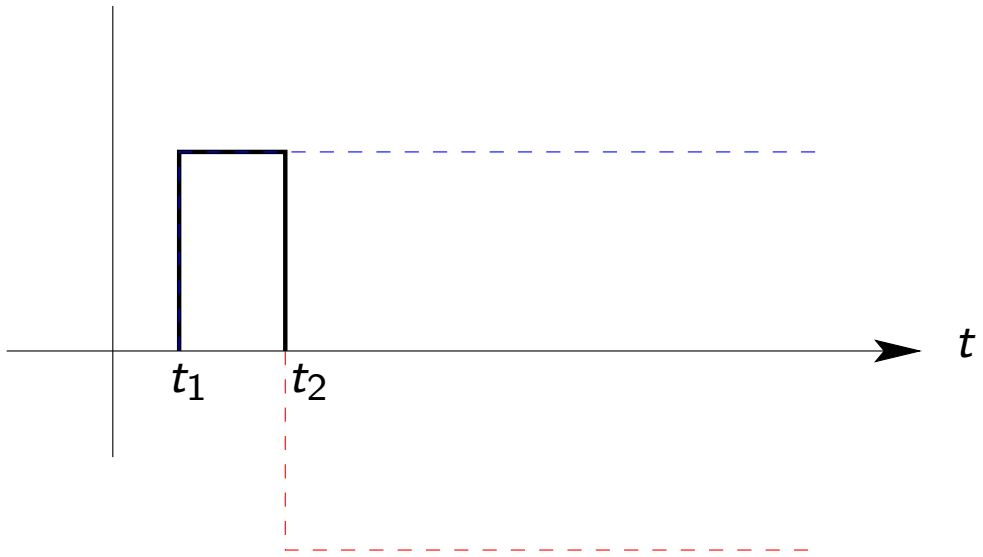
- Projeto Servo: O controle $C(s)$ deve ser projetado de modo que a saída $Y(s)$ rastreie (do inglês *track*) a entrada $R(s)$. Faz-se $P(s)$ nulo.
- Projeto de Regulação: O controle $C(s)$ deve ser projetado de modo que a saida $Y(s)$ rejeite (seja insensível) à entrada de perturbação $P(s)$. Faz-se $R(s)$ nulo.

Outras dicas do matlab

- Criar uma função de transferência: Exemplo

$$G(s) = \frac{s+2}{s^2+3s+4} \Rightarrow \texttt{G = tf([1 2],[1 3 4])}$$

- Criar um pulso no simulink: fazer como a diferença entre dois degraus (bloco `step`)



Outras dicas do matlab

- Produtos de função de transferência:

```
G=tf([1],[1 2]);
H=tf([2],[1 1 2]);
X=G*H;
```

- No simulink, para criar um bloco para armazenar uma função de transferência, dar preferência para o bloco `LTI system`. O bloco permite carregar uma função de transferência definida na linha de comando do matlab.
- Para salvar os dados da saída de uma simulação, dar preferência para o bloco `out`. Para plotar os dados resultantes, basta digitar `plot(tout,yout)` na linha de comando do matlab.

Diagrama de blocos do simulink

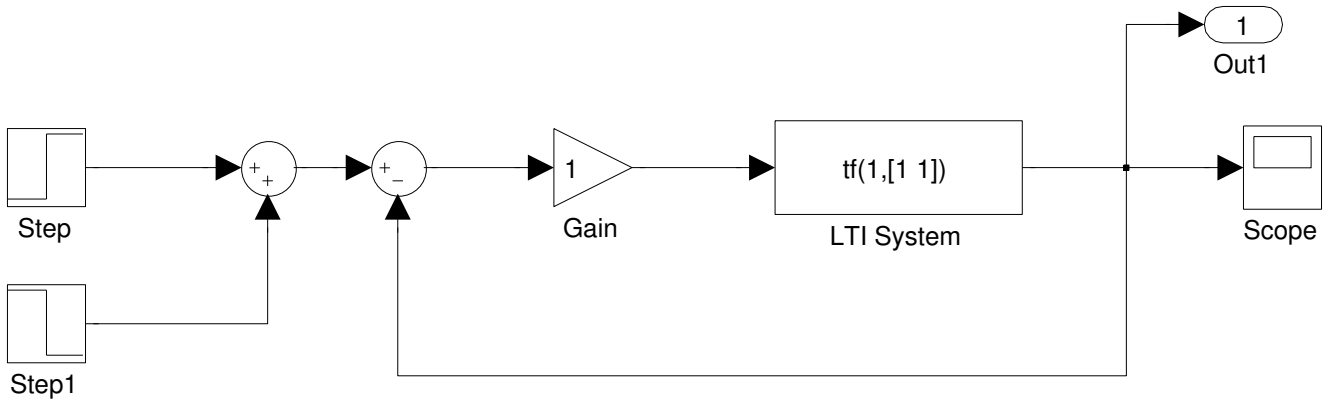
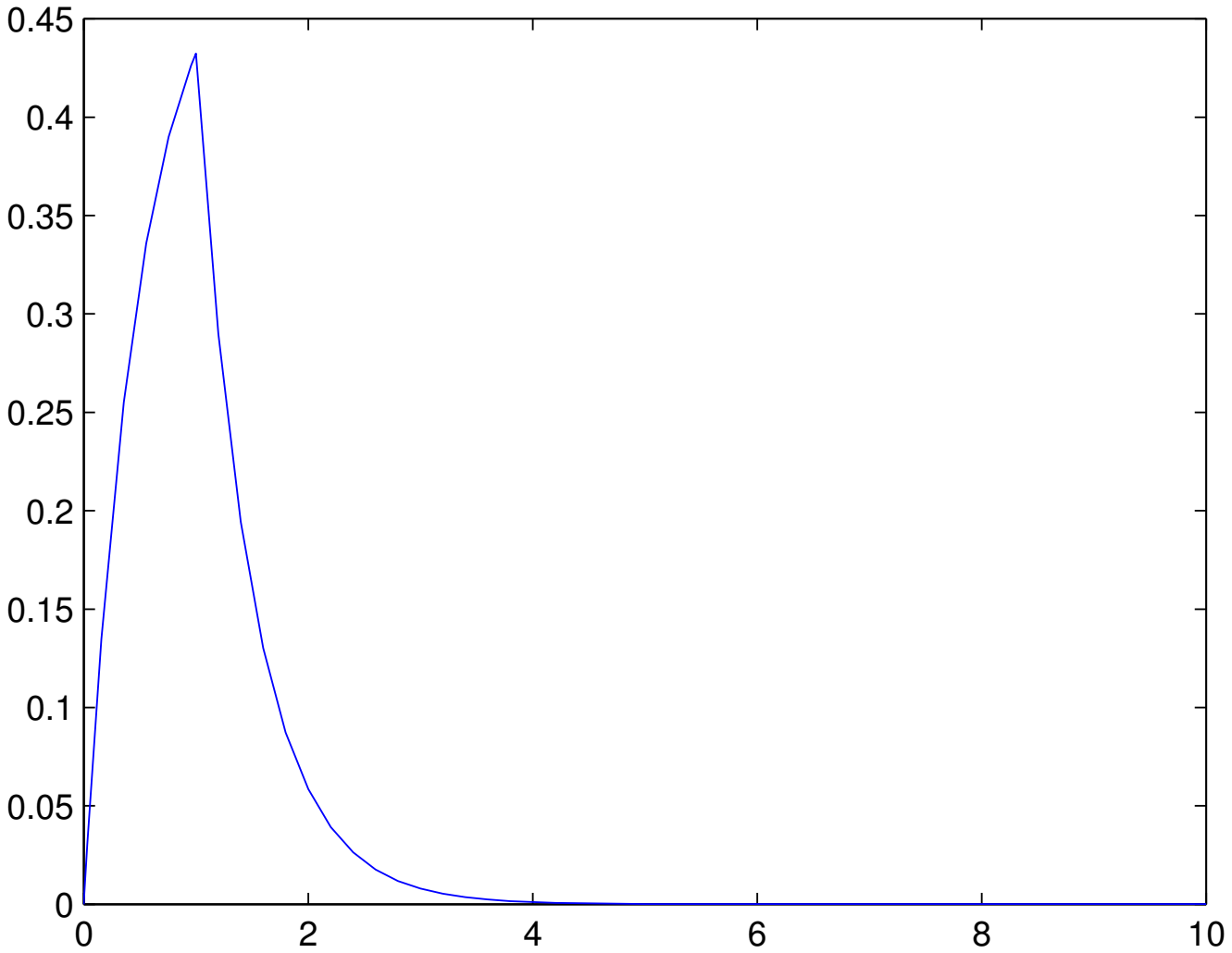


Diagrama de blocos do simulink

```
plot(tout,yout)
```



Rotina auxiliar: `plotRawData.m` I

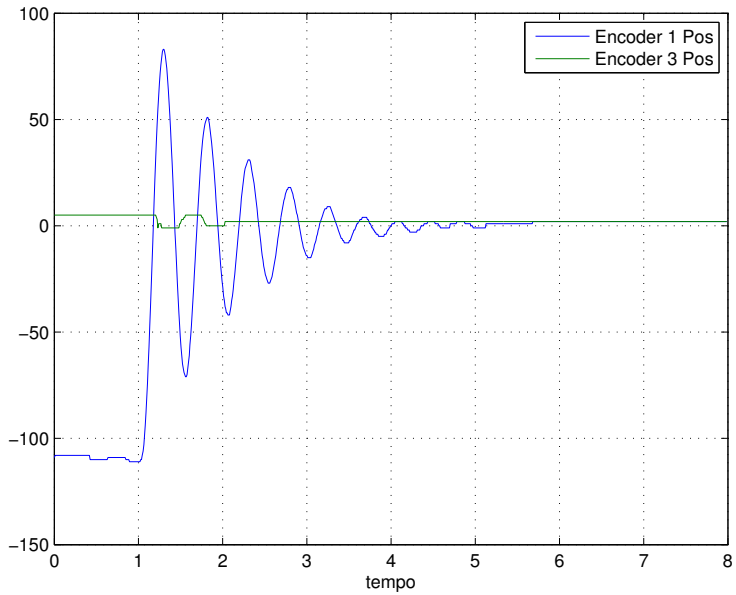
- A rotina `plotRawData.m` é muito útil para plotar dados gerados pelos equipamentos ECP no Matlab. No software que gerencia os equipamentos, é possível exportar os dados por meio do menu `Data/Export Raw Data`. A saída é um arquivo no formato `txt`, e será o parâmetro de entrada da rotina `plotRawData.m`.

- Exemplo: Seja o arquivo `teste.txt` o arquivo exportado pelo software que gerencia os equipamentos ECP. No Matlab, o seguinte comando pode ser executado:

```
>> plotRawData('item1-6.txt')
Dados disponiveis para plotar:
Encoder 1 Pos  (coluna 1)
Encoder 3 Pos  (coluna 2)
```

- O próximo passo é escolher quais dados serão desenhados simultaneamente. Nesse exemplo, dois dados estão disponíveis: Posição do Encoder 1 (Encoder 1 Pos) e Posição do Encoder 3 (Encoder 3 Pos). Se mais de um dado for escolhido para o desenho, então as colunas correspondentes devem ser informadas entre colchetes. Exemplo: [1 2]. Com essa escolha, tem-se o seguinte gráfico

Rotina auxiliar: `plotRawData.m` II



Saída gerada pela rotina `plotRawData.m`.

- Caso cada dado precise ser plotado em uma figura diferente, basta em cada execução do comando escolher apenas uma coluna. Uma outra utilidade importante da rotina é a possibilidade do usuário apenas extrair os dados contidos no arquivo, sem desenhar nenhum gráfico. Para isso, basta colocar um parâmetro de saída na rotina:

Rotina auxiliar: `plotRawData.m` III

```
>> saida = plotRawData('teste.txt')

saida =

tempo: [903x1 double]
var: [903x2 double]
nomeVar: {'Encoder 1 Pos '  'Encoder 3 Pos '}
```

- Dentro da variável será criada uma estrutura, contendo basicamente os instantes de tempo em que os dados foram colhidos, as variáveis (cada variável ocupa uma coluna da matriz) e os nomes das variáveis (armazenados num `cell array`). Uma possível utilidade dessa opção é o tratamento de vários arquivos, eventualmente fazendo vários desenhos numa mesma figura.

- Exemplo: Suponha que existam dois arquivos exportados, `teste1.txt` e `teste2.txt`. O objetivo é extrair os dados e desenhá-los numa mesma figura, mas em gráficos separados. Primeiramente extraímos os dados:

Rotina auxiliar: plotRawData.m IV

```
>> saida1 = plotRawData('teste1.txt')

saida1 =

tempo: [903x1 double]
var: [903x2 double]
nomeVar: {'Encoder 1 Pos '  'Encoder 3 Pos '}

>> saida2 = plotRawData('teste2.txt')

saida2 =

tempo: [903x1 double]
var: [903x2 double]
nomeVar: {'Encoder 2 Pos '  'Encoder 4 Pos '}
```

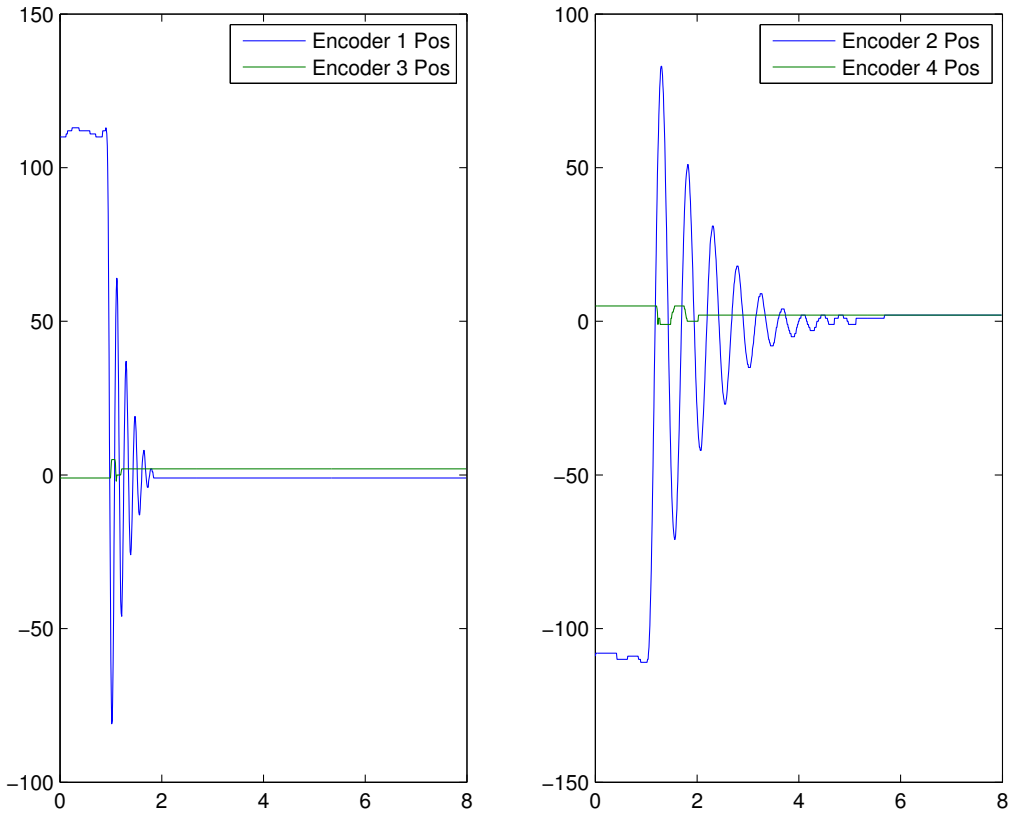
Rotina auxiliar: `plotRawData.m` V

- Na sequência usamos o comando `subplot(m,n,p)` para colocar vários gráficos numa mesma figura. Os parâmetros $m \times n$ definem um grid de m linhas e n colunas, e o parâmetro p define a posição do grid (contagem da esquerda para a direita, de cima para baixo). Para o exemplo em estudo, vamos definir um grid horizontal de duas posições:

```
>> subplot(1,2,1)
>> plot(saida1.tempo,saida1.var)
>> legend(saida1.nomeVar{1},saida1.nomeVar{2})
>> subplot(1,2,2)
>> plot(saida2.tempo,saida2.var)
>> legend(saida2.nomeVar{1},saida2.nomeVar{2}
```

- O resultado é ilustrado a seguir

Rotina auxiliar: `plotRawData.m` VI



Comando `subplot()`: dois gráficos numa mesma figura.

Gráficos simultâneos

- Seja as matrizes

$$t = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{bmatrix}, \quad y = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 1 & 0.5 \\ 4 & 2 & 1 \\ 6 & 3 & 1.5 \\ 8 & 4 & 2 \\ 10 & 5 & 2.5 \\ 12 & 6 & 3 \\ 14 & 7 & 3.5 \end{bmatrix}$$

Caso deseje-se desenhar um gráfico em que os valores da abscissa estão em t e os valores da ordenada estão na primeira coluna de y, usa-se o comando `plot(t,y(:,1))`. De modo similar, para desenhar como ordenada os valores da terceira coluna, utiliza-se `plot(t,y(:,3))`. A notação ":" indica "todos" e $y(i,j)$ indica o elemento que fica na linha i e coluna j.

Dicas Gerais

- Muita atenção na montagem do equipamento, pois montagem errada implica em resultados errados (o experimento precisará ser realizado novamente).
- Não esquecer de clicar em *Implement Algorithm* **sempre** que algum parâmetro relacionado ao controle for modificado.
- Sempre que for solicitado que o parâmetro k_{pf} (ganho do pré-filtro) seja ajustado, tome como critério de desempenho a ser melhorado (menor possível) o **erro em regime**.
- Seja o polinômio: $t_2 s^2 + t_1 s + t_0$. Em geral o coeficiente com índice zero, por exemplo t_0, é o coeficiente associado ao **termo constante** do polinômio.
- Não é necessário realizar o pré-relatório. Contudo, a sua elaboração pode acelerar a realização da experiência.