Universidade Federal de Minas Gerais
DEPARTAMENTO DE ENGENHARIA ELETRÔNICA
ELT129 – Oficina de Modelagem e Simulação

Professor: Leonardo Mozelli – lamoz@ufmg.br

Tutorial 7 – Representação de Sistemas por Diagrama de Blocos / Simulink

1 Introdução

Neste tutorial são apresentadas a representação de sistemas lineares utilizando Diagrama de Blocos e redução destes para obter uma função de transferência equivalente. A ferramenta Simulink, integrante do pacote Matlab, permite realizar a simulação de diagrams de blocos.

2 Diagrama de Blocos

Sistemas de engenharia podem ser constituídos de um grande número de componentes. Pode ser quase impossível, por exemplo, analisar o diagrama do circuito de um rádio ou de um receptor de televisão de uma só vez. Nesses casos, é conveniente representar o sistema por meio de subsistemas interconectados, onde o diagrama de blocos é construído a partir das equações que descrevem um determinado sistema.

Um diagrama de blocos de um sistema é uma representação das funções desempenhadas por cada componente e do fluxo de sinais. Esse diagrama indica a inter-relação que existe entre os vários componentes. Em um diagrama de blocos todas a variáveis do sistema são ligadas às outras variáveis em termos de relações de entrada $X(s)$ e saída $Y(s)$. Esta relação em sistemas lineares é chamada de função de transferência $H(s)$.

A representação de sistemas por diagrama de blocos facilita visualizar todas as partes fundamentais que compõe o sistema sob análise (ver Figura 1). As variáveis que constituem as entradas e saídas de cada um dos blocos podem ser definidas de forma única, independente do tipo de grandeza física que se deseja controlar.

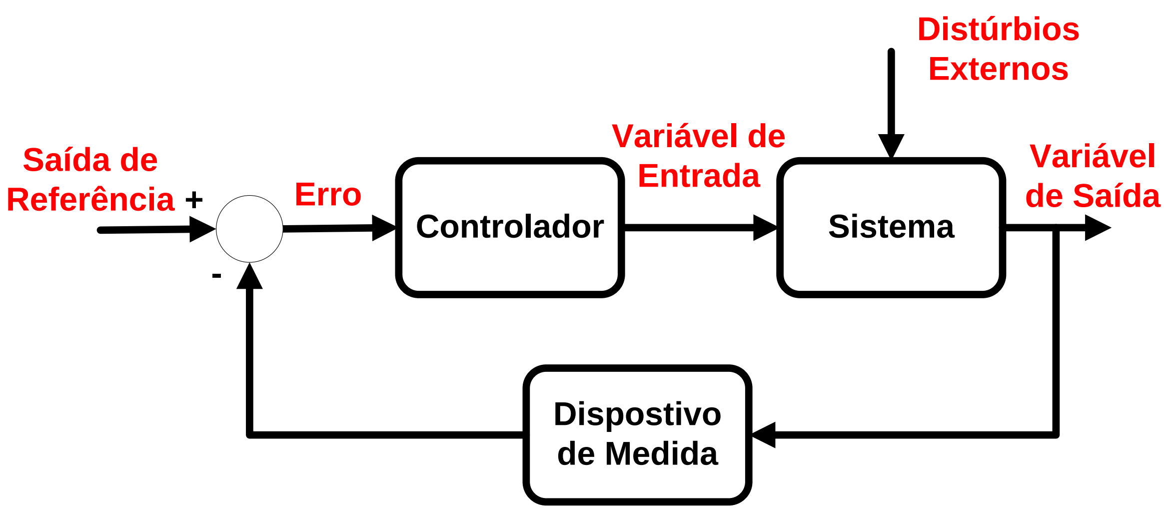


Figura 1: Diagrama de blocos.

Os sistemas lineares representados por meio de diagrama de blocos utilizam três elementos básicos: bloco, somador e nó, como ilustrado, respectivamente, nas Figuras 2, 3 e 4.

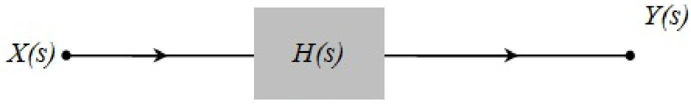


Figura 2: Representação de diagrama de bloco.

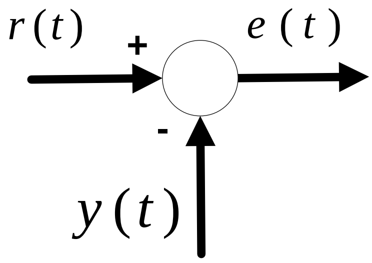


Figura 3: Representação do somador.

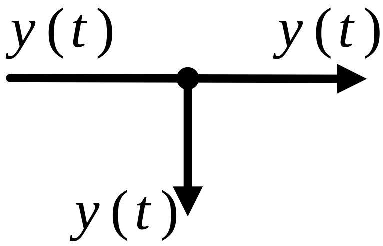


Figura 4: Representação do nó.

2.1 Álgebra de Blocos

A regra principal na álgebra de blocos é não alterar a relação entre as variáveis de entrada e saída dos blocos que se quer simplificar.

As Figuras 5-7 mostram diferentes diagramas de blocos de sistemas. Nas figuras é possível observar também que funções de transferência de subsistemas interconectados podem ser simplificadas em uma função de transferência única equivalente.

Configurações Básicas

- Conexão de blocos em série:

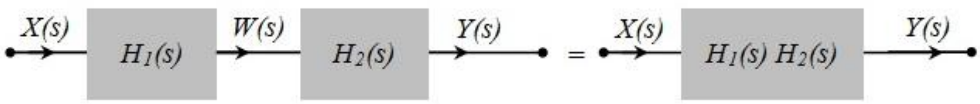


Figura 5: Representação em diagrama de blocos em série.

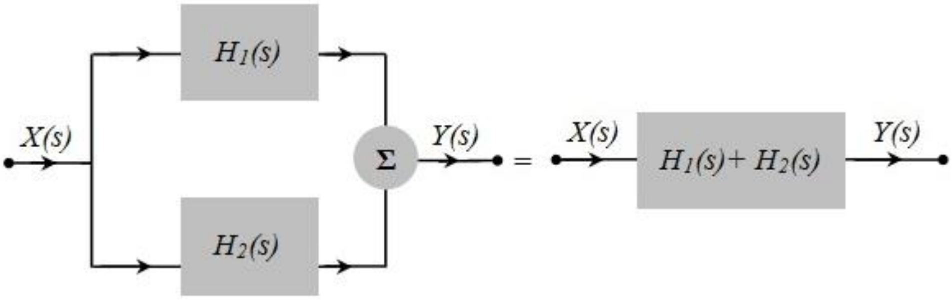


Figura 6: Representação em diagrama de blocos em paralelo.

- Conexão de blocos em paralelo:

- Conexão de blocos em malha fechada:

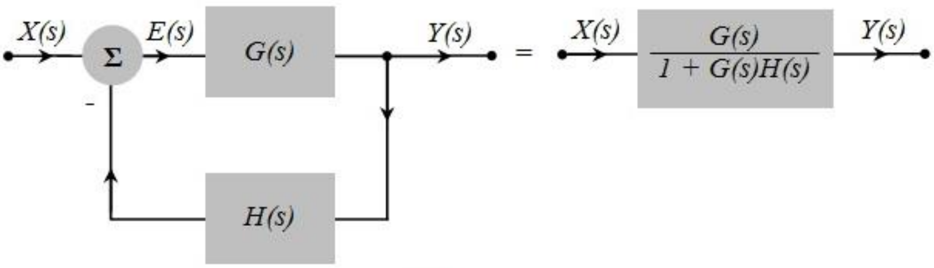


Figura 7: Representação em diagrama de blocos em malha fechada.

Particularmente, quando a saída de um sistema é realimentada à entrada (Figura 7), a função de transferência equivalente pode ser calculada da seguinte forma:

$$\left.\begin{array}{l} E(s) = X(s) - H(s)Y(s) \\ Y(s) = G(s)E(s) = G(s)[X(s) - H(s)Y(s)] \end{array}\right\} \quad \Rightarrow \quad Y(s)[1 + G(s)H(s)] = G(s)X(s)$$

Desta forma, um sistema realimentado, como o da Figura Figura 7, pode ser substituído por um bloco único com a seguinte função de transferência:

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

2.1.1 Movimento do Bloco em Relação a um Somador

Na Figura 8 são apresentadas duas manipulações de blocos com relação a um bloco somador: a inserção de um bloco dentro da malha e a retirada de um bloco de dentro da malha.

2.1.2 Movimento do Bloco em Relação a um Ponto de Junção:

Na Figura 9 são apresentadas duas manipulações de blocos com relação a um ponto de junção: mudar o bloco para depois do ponto de junção ou mudar o bloco para antes do ponto de junção.

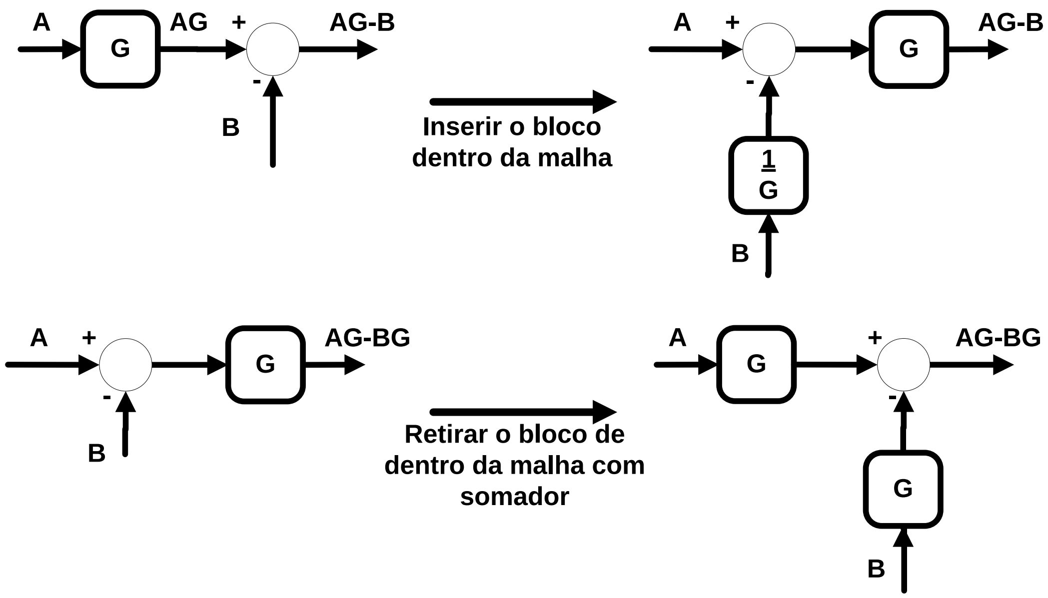


Figura 8: Movimento do Bloco em Relação a um Somador.

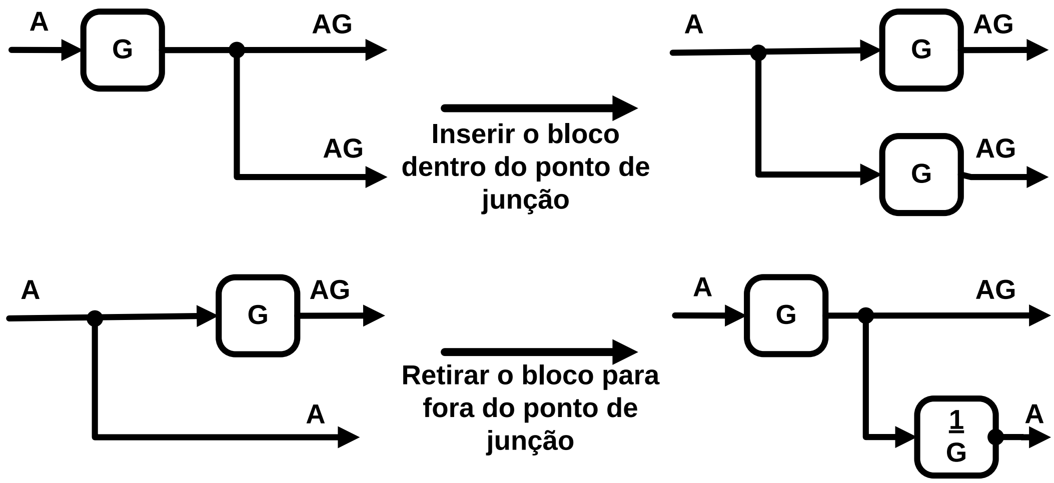


Figura 9: Movimento do Bloco em Relação a um Ponto de Junção.

2.1.3 Exemplo

Considere dois tanques interligados, representados pela Figura 10, cuja relação linear entre as vazões $Q_1(t)$ e $Q_2(t)$ e as respectivas alturas das colunas de líquido existentes em cada um dos tanques é dada por:

$$Q_1(t) = \frac{H_1(t) - H_2(t)}{R_1}$$
$$Q_2(t) = \frac{H_2(t)}{R_2}$$

sendo R_1 e R_2 constantes que representam a resistência aos fluxos e são dependentes da posição de ajuste das válvulas C_1 e C_2.

O modelo que representa a variação de volume nos tanques é descrito por equações diferen-

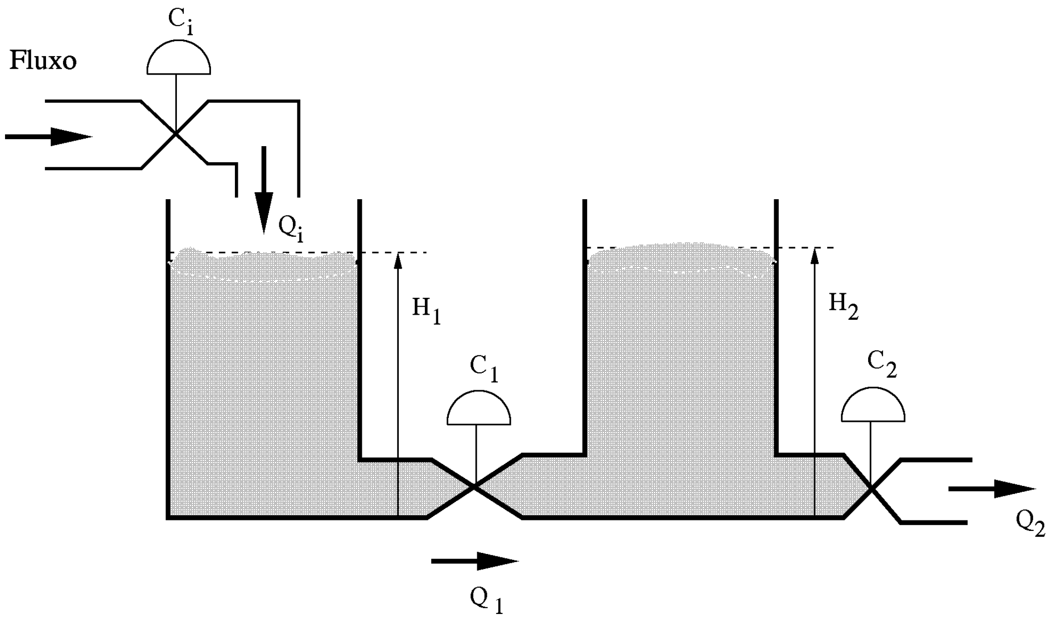


Figura 10: Dois tanques interligados.

ciais de primeira ordem invariantes no tempo:

$$\frac{dV_1(t)}{dt} = \frac{dA_1H_1(t)}{dt} = Q_i - Q_1 \quad \Rightarrow \quad \frac{dA_1H_1(t)}{dt} = Q_i - \frac{H_1(t) - H_2(t)}{R_1}$$
$$\frac{dV_2(t)}{dt} = \frac{dA_2H_2(t)}{dt} = Q_1 - Q_2 \quad \Rightarrow \quad \frac{dA_2H_2(t)}{dt} = \frac{H_1(t) - H_2(t)}{R_1} - \frac{H_2(t)}{R_2}$$

com A_1 e A_2 as áreas transversais das superfícies de cada um dos tanques, consideradas uniformes. A Figura 11 esquematiza a conexão entre os tanques e como a saída de interesse $Q_2(t)$ é afetada pela entrada $Q_i(t)$.

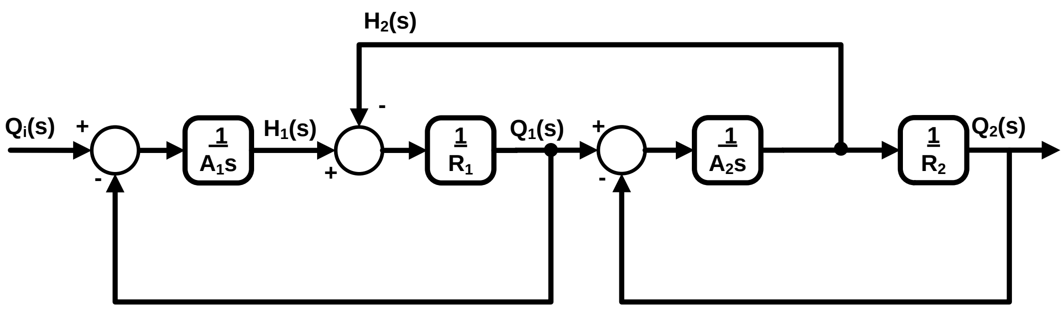
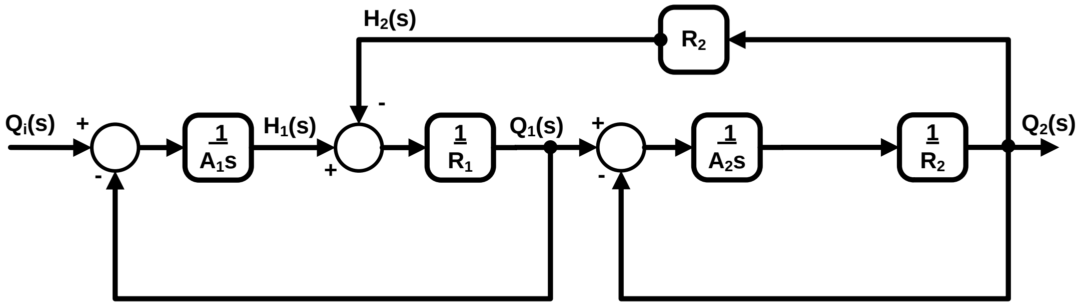


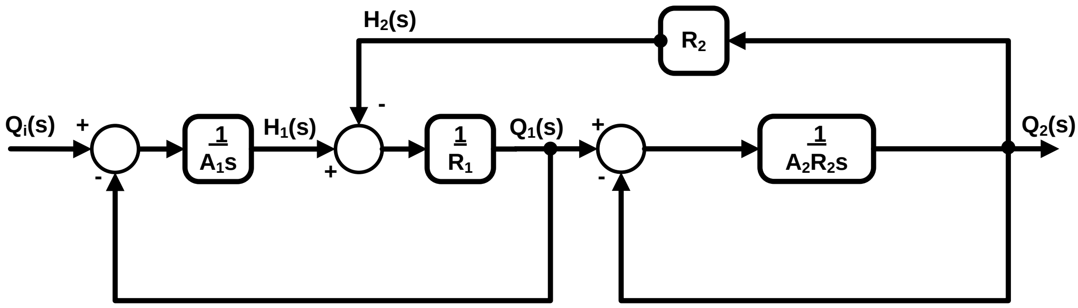
Figura 11: Diagrama de blocos.

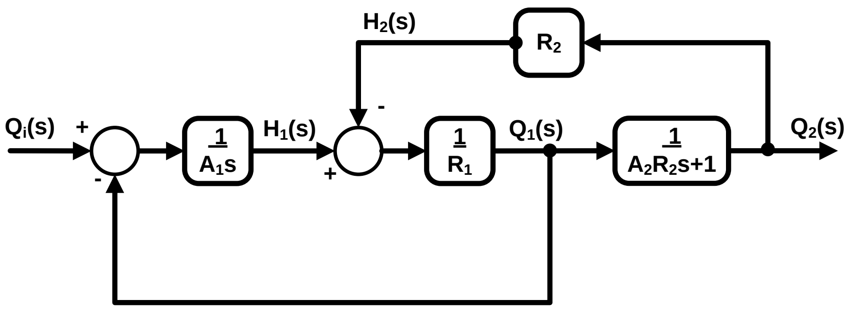
Usando álgebra de blocos, o intuito é obter a função de transferência equivalente entre Q_i e Q_2.

Passo 1: Movimentação do bloco $\frac{1}{R_2}$

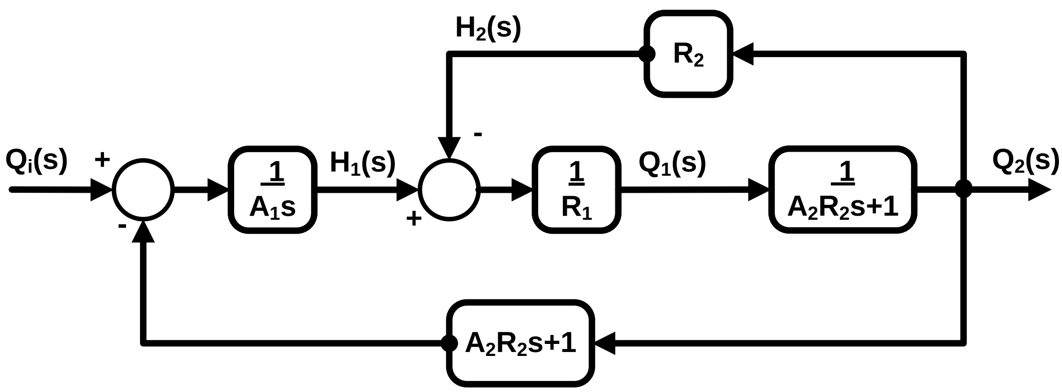


Passo 2: União dos blocos $\frac{1}{R_2}$ e $\frac{1}{A_2 s}$ e fechamento desta malha

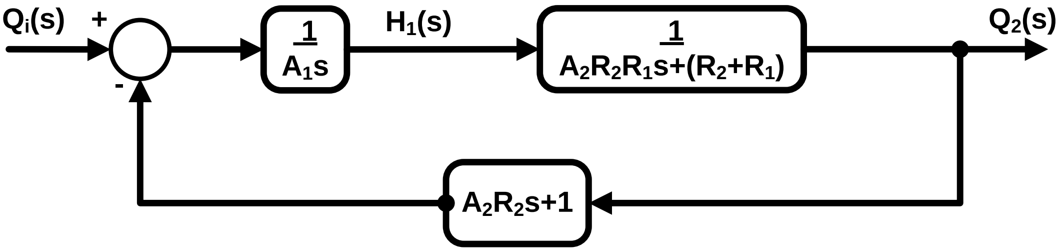




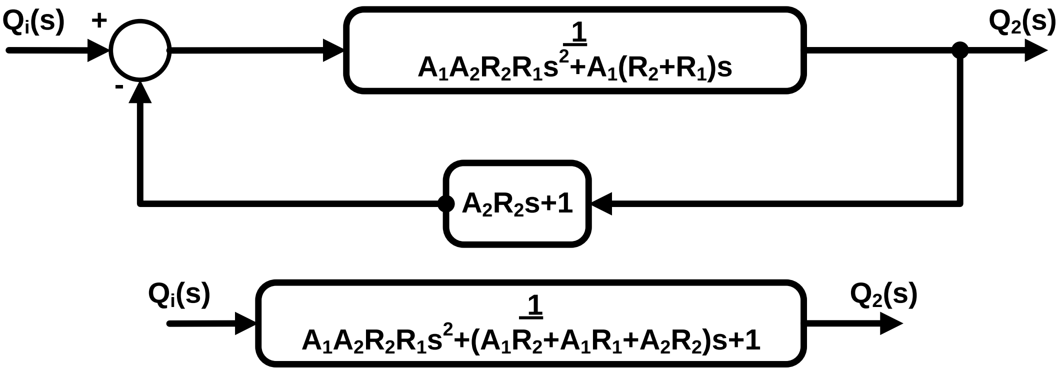
Passo 3: Passagem do bloco $\frac{Q_2}{Q_1}$ para dentro da junção



Passo 4: União dos blocos $\frac{1}{R_1}$ e $\frac{1}{A_2R_2s+1}$ e fechamento da malha entre $\frac{Q_2}{H_1}$



Passo 5: União dos blocos $\frac{1}{A_1s}$ e $\frac{Q_2}{H_1}$ e fechamento da malha entre $\frac{Q_2}{Q_i}$



2.1.4 Exemplo com MatLab

Suponha as seguintes funções de transferência

$$H_1(s) = \frac{num1}{den1} \quad \text{e} \quad H_2(s) = \frac{num2}{den2}$$

Em Matlab, as possíveis associações entre as funções H_1 e H_2, conforme descritas na seção anterior, podem ser obtidas a partir dos comandos `series`, `parallel` e `feedback`, cuja sintaxe é:

```
>> Sys = series(H1,H2);
>> Sys = parallel(H1,H2);
>> Sys = feedback(H1,H2);
```

Também é possível realizar redução de blocos através dos comandos `connect` e `sumblk`.

2.2 Fórmula de Mason

O método para redução de diagrama de blocos apresentado na seção anterior utiliza de reduções sucessivas, o que pode resultar muito trabalhoso. Para evitar este trabalho, a *Fórmula de Mason* permite uma redução imediata dos diagramas de blocos da seguinte forma

$$F(s) = \frac{\sum \bar{G}_k \Delta_k}{\Delta}$$

sendo que $F(s)$ é a função de transferência do sistema representado pelo diagrama de blocos, $\bar{G}_k$ é o ganho do caminho de avanço de número k que liga a entrada à saída, e Δ é dado por

$$\Delta = 1 + \sum_i L_i + \sum_{i,j} L_i L_j + \sum_{i,j,r} L_i L_j L_r + \cdots$$

com L_i, L_j L_r, ... são os ganhos das malhas $i, j, r, \ldots$ do diagrama, com os sinais trocados, sendo que os produtos dos vários ganhos não devem incluir malhas que se tocam (passar mais de uma vez pelo mesmo caminho).

No numerador, Δ_k é o mesmo que Δ, não contendo, porém, o ganho das malhas que tenham um ou mais pontos de contato com o caminho de avanço k.

2.2.1 Exemplo

Reduza com auxílio da fórmula de Mason o diagrama de blocos da Figura 12.

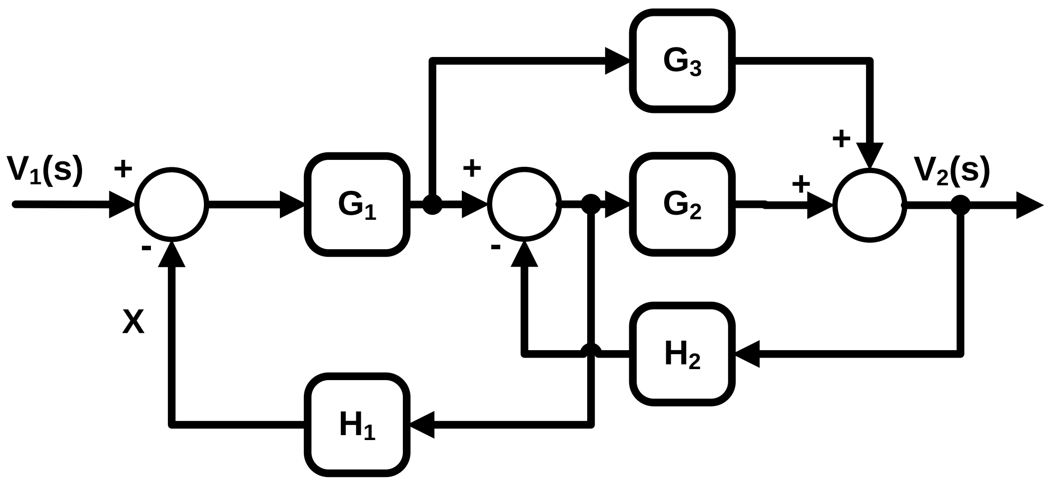


Figura 12: Diagrama de blocos.

Solução:

- Caminhos de avanço

$$\bar{G}_1 = G_1 G_2 \quad \text{e} \quad \bar{G}_2 = G_1 G_3$$

- Há três malhas cujos ganhos são $L_1 = G_1 H_1$, $L_2 = G_2 H_2$ e $L_3 = -G_1 G_3 H_1 H_2$. O Δ do denominador é, então

$$\Delta = 1 + G_1 H_1 + G_2 H_2 - G_1 G_3 H_1 H_2$$

- Finalmente,

$$F(s) = \frac{G_1 G_2 + G_1 G_3}{1 + G_1 H_1 + G_2 H_2 - G_1 G_3 H_1 H_2}$$

3 Simulink

No Tutorial 6 foram discutidas formas de encontrar soluções numéricas para equações diferenciais que descrevem o comportamento dinâmico de sistemas. No âmbito do Tutorial 5, as funções `ode` foram largamente utilizadas para chegar às soluções desejadas. Entretanto, em alguns casos, pode ser mais conveniente representar as equações dinâmicas de sistemas graficamente por meio de diagramas de blocos. Para tanto, a ferramenta *Simulink*, disponível junto com o Matlab, pode ser utilizada para representar sistemas contínuos, sistemas discretos e, também, sistemas híbridos (sistemas que combinam dinâmicas contínuas e discretas).

O Simulink pode ser inicializado digitando-se `simulink` na linha de comando do Matlab ou clicando no botão presente na aba “Home”. Após inicializado, uma tela será mostrada para escolher um modelo e iniciar a construção dos diagramas de blocos. Na grande maioria dos casos, a opção mais adequada é “Blank Model”.

Uma vez que o novo modelo seja carregado, clique em “Library Browser” e uma tela similar à apresentada na Figura 13 será mostrada. Dependendo dos *toolboxes* instalados haverá uma coleção bastante grande de opções de blocos que podem ser utilizados. Para este curso, apenas os blocos disponibilizados na biblioteca padrão, denominada *Simulink*, serão utilizados. Utilize um tempo para conhecer os blocos existentes em cada coleção, sobretudo em

- *Commonly Used Blocks*
- *Continuous*
- *Discrete*
- *Math Operations*
- *Sinks*
- *Sources*

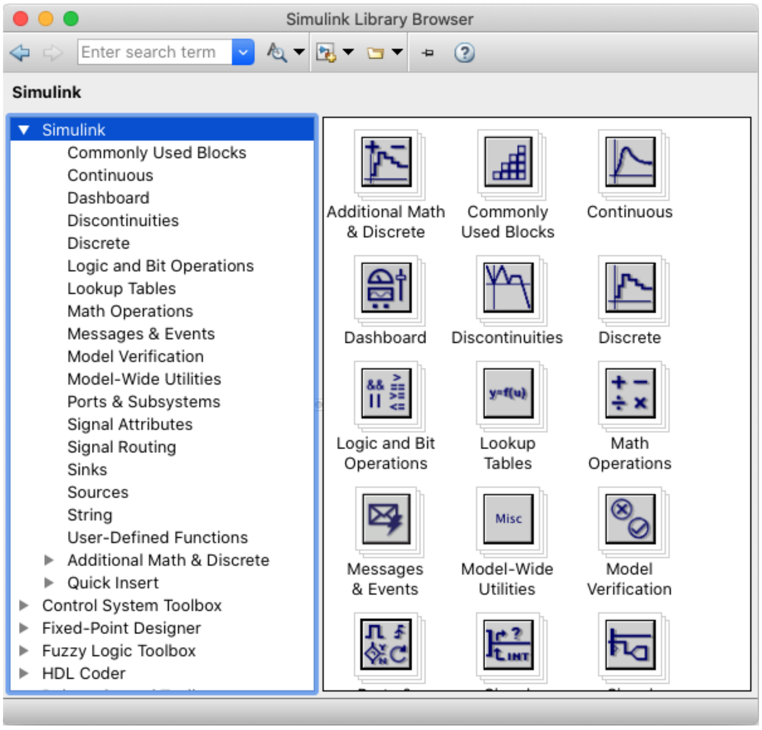


Figura 13: Biblioteca Simulink.

Utilizando os blocos presentes na coleção *Commonly Used Blocks*, implemente o diagrama de blocos da Figura 14. Qual equação é descrita por esse diagrama?

Note que os parâmetros de cada bloco na Figura 14 podem ser modificados clicando duas vezes sobre o bloco. Na Figura 15 são apresentados dois exemplos correspondentes ao bloco de ganho a e ao primeiro integrador.

É possível definir diretamente em cada bloco os valores dos ganhos a, b e c, além das condições iniciais $y(0)$, $\dot{y}(0)$ e $\ddot{y}(0)$. No entanto, isso traz pouca flexibilidade, pois, se for interessante alterar quaisquer valores, será necessário realizar tal alteração diretamente no diagrama. Uma forma mais conveniente e flexível é definir os valores dos parâmetros em termos de variáveis e definir tais variáveis diretamente no *workspace* do Matlab ou em um *script* que é executado previamente. Ao clicar no botão “Run”, o Simulink verificará se as variáveis estão definidas no *workspace* e as importará antes de iniciar a simulação.

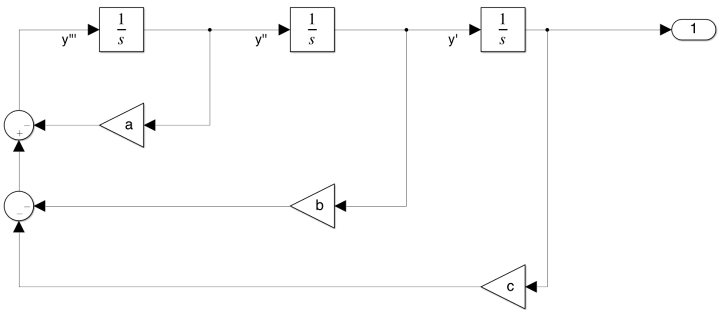
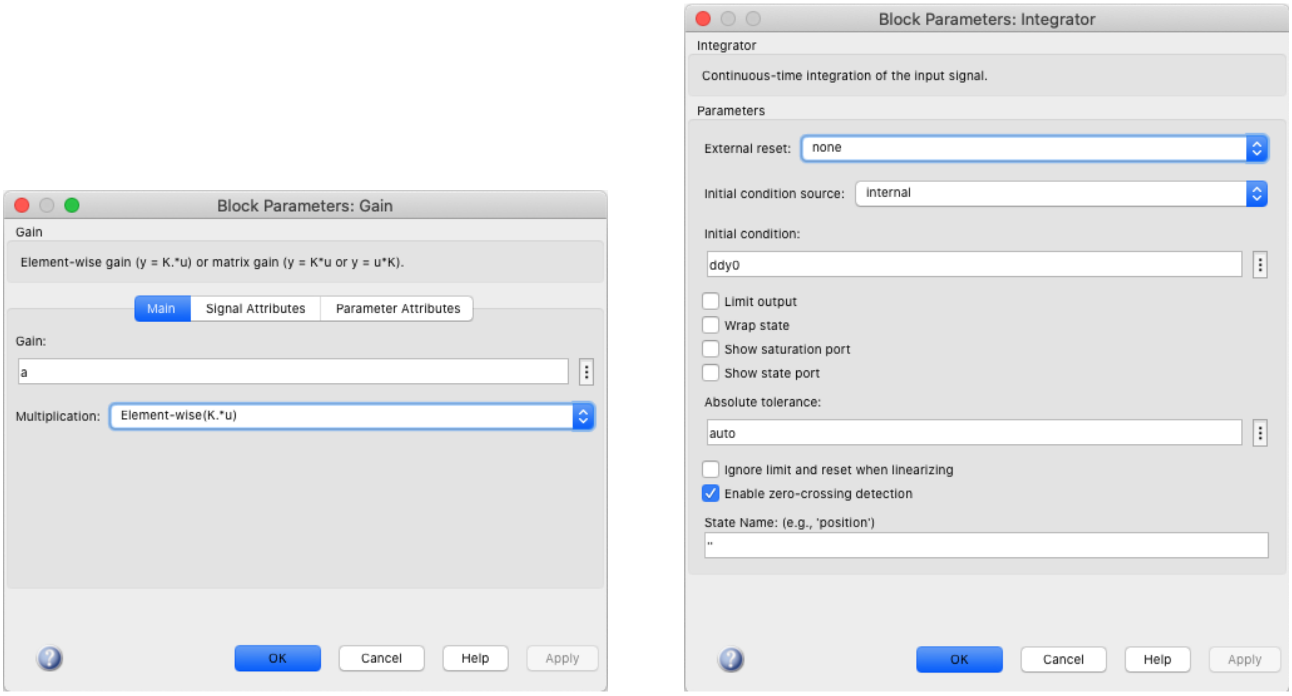


Figura 14: Diagrama de blocos de uma EDO de terceira ordem.



(a) Parâmetros do bloco de ganho.

(b) Parâmetros do bloco integrador.

Figura 15: Exemplos de parâmetros modificáveis em diferentes blocos.

Antes de simular o sistema, há mais uma configuração a ser feita e que auxiliará na apresentação dos resultados. Clique em `Modeling > Model Settings > Data Import/Export` e verifique se as opções estão iguais às apresentadas na Figura 16. Essas configurações são importantes para podermos plotar os gráficos de resultados de forma simples.

Efetuadas todas as configurações necessárias, defina $a = 2$, $b = 1$, $c = 1.5$, $y(0) = 1$, $\dot{y}(0) = 0$ e $\ddot{y}(0) = 0.3$ e execute a simulação. Pode ocorrer de aparecer uma caixa de diálogo solicitando para atualizar os parâmetros, bastando clicar no botão correspondente. Assim que a simulação finalizar, plote o gráfico de y em função do tempo utilizando o seguinte código

```
>> plot(tout,yout(:,1))
>> xlabel('Tempo [s]')
>> ylabel('y(t)')
```

O resultado obtido deve se assemelhar ao apresentado na Figura 17.

Neste caso específico, o comando `plot` apresentado acima poderia ser simplificado para

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

pois há somente uma saída. Quando houver mais de uma saída (bloco `Out`) do diagrama, é importante plotar como inicialmente feito, pois cada coluna da variável `yout` corresponde a

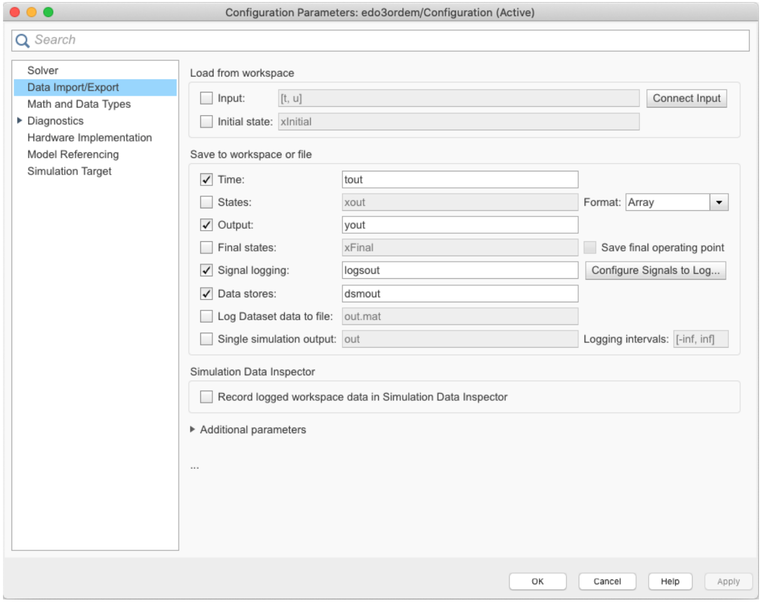


Figura 16: Configuração básica dos dados de entrada e saída do Simulink.

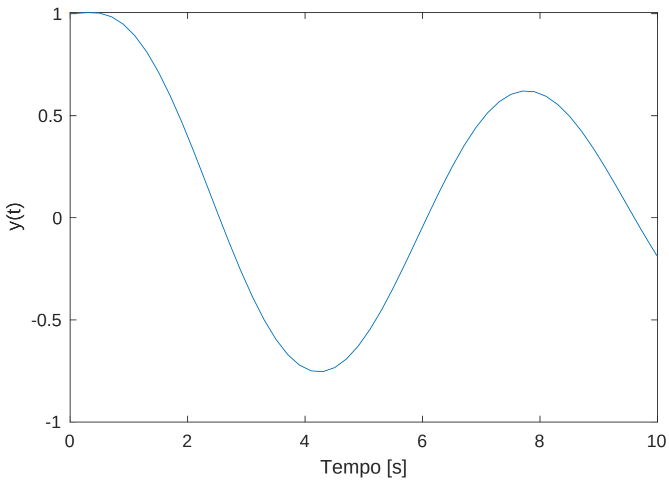


Figura 17: Resultado da simulação da EDO apresentada no diagrama de blocos da Figura 14.

uma saída do diagrama.

Além disso, em certos casos é interessante acompanhar graficamente como a simulação está evoluindo e os diversos sinais intermediários. Para tanto são úteis os blocos “Scope” e “XY Graph”. Aumente o tempo de simulação para 100 segundos e acrescente blocos “Scope” ao diagrama da Figura 14 para verificar a evolução temporal de $\dddot{y}$, $\ddot{y}$ e $\dot{y}$.

3.1 Invocando o Simulink por meio de *scripts*

Após ter sido construído, um diagrama Simulink pode ser invocado e executado diretamente pela linha de comando, não sendo necessário abrir o diagrama previamente. Isso pode ser feito por meio do comando `sim()`, como exemplificado a seguir:

```
a = 2;
b = 1;
c = 1.5;
y0 = 1;
dy0 = 0;
```

```
ddy0 = 0.3;

sim('diagEdo',10);

plot(tout,yout(:,1))
xlabel('Tempo [s]')
ylabel('y(t)')
```

O primeiro argumento do comando `sim()` é o nome do diagrama Simulink sem extensão e o segundo argumento, opcional, é o tempo de simulação. Quando omitido, o tempo de simulação é aquele registrado dentro do diagrama Simulink. O resultado do *script* acima é idêntico ao apresentado na Figura 17.

Invocar o Simulink diretamente pela linha de comando é bastante útil quando se deseja, por exemplo, analisar a resposta de um sistema para diferentes conjuntos de parâmetros e comparar as saídas obtidas.

4 Outros pontos importantes

Além de uma forma mais pictórica para construir um modelo de simulação de sistemas dinâmicos, há outras vantagens em utilizar o Simulink, dentre as quais destacam-se:

- **Não linearidades:** há diversos blocos que auxiliam na construção de diagramas de sistemas sujeitos a não linearidades (como saturação, histerese, zona morta, dentre outras).
- **Sinais de referência e ruídos:** é possível adicionar blocos para simular diversos sinais de referência a serem aplicados a sistemas reais além de blocos referentes a perturbações externas afetando o sistema.
- **Funções definidas pelo usuário:** nem sempre uma determinada operação sobre um sinal está disponível ou, por outro lado, pode ser muito complicado implementá-la em termos de blocos. Nesses casos, pode-se utilizar blocos que permitem ao usuário definir sua própria função, a qual é executada ao longo da simulação do sistema.
- **Subsistemas:** em diversas situações o diagrama de blocos criado pode se tornar demasiadamente grande, impossibilitando a sua manutenção ao longo do tempo, ou pode ser que seja desejável isolar partes do diagrama que possuem uma função específica. Nesses casos, podemos criar subsistemas a partir de um conjunto de blocos do diagrama. Os subsistemas, de modo mais geral, são caracterizados por blocos que possuem uma dinâmica independente do restante do sistema, sendo possível trocá-los por outros blocos que desempenham funções similares.
- **Conexão com dispositivos externos:** existem bibliotecas dedicadas à conexão do Simulink com um *hardware* externo, permitindo que parte da simulação ocorra em *software* e parte em dispositivo físico.

5 Exercícios

1. Dadas as funções de transferência $H_1(s) = \frac{1}{s+2}$ e $H_2(s) = \frac{3}{s+4}$, pede-se:
   a. Conecte os modelos em série
   b. Conecte os modelos em paralelo
   c. Conecte os modelos em malha fechada, com $H_1(s)$ no ramo direto e $H_2(s)$ na realimentação

d. Idem ao exercício 1c, mas considerando realimentação positiva

e. Repita o exercício 1c usando o comando `feedback`

f. Repita o exercício 1c usando o comando `connect`

2. Baseado no sistema realimentado da Figura 7 com $G(s) = \frac{K}{s(s+8)}$ e $H(s) = 1$, determine a função de transferência para cada um dos seguintes casos:

   a. $K = 7$

   b. $K = 16$

   c. $K = 80$

   Comente a respeito da 'resposta ao degrau unitário' em cada um dos casos. Trace as respostas no mesmo gráfico usando `plot` e `hold`.

   Adicionalmente:

   d. Encontre a resposta à rampa unitária quando $K = 80$. Note que a resposta à rampa unitária é equivalente à derivada da resposta ao degrau unitário.

3. Para o exemplo do dois tanques interligados, obtenha as seguintes funções de transferência utilizando redução de blocos:

   a. $H_2(s)/H_1(s)$

   b. $H_2s)/Q_i(s)$

   c. $H_1(s)/Q_i(s)$

4. Considere um sistema de primeira ordem descrito por

$$\dot{x} = f(t, x) = -x^3, \quad x(0) = x_0$$

   Construa um diagrama de blocos no Simulink representando esse sistema. Defina um certo horizonte de tempo T e escolha uma condição inicial não nula. Utilizando o que foi apresentado neste tutorial, simule o sistema para ao menos duas condições iniciais e apresente em um gráfico a evolução ao longo do tempo de $x(t)$.

5. Seja um sistema não linear descrito pela equação diferencial

$$\ddot{y} + 3\dot{y} + e^{-t}y = 8\text{sen}\,(5t)$$

   Pede-se:

   a) Construa um diagrama de blocos que represente o sistema.

   b) Simule o comportamento do sistema para condições iniciais nulas.

6. Considere o sistema massa-mola-amortecedor apresentado no Tutorial 2, o qual é reproduzido na Figura 18 e cujo modelo matemático em torno do ponto de equilíbrio estático é

$$m\ddot{y}(t) + c\dot{y}(t) + ky(t) = f(t)$$

   Tomando $m = 3\,\text{kg}$, $c = 1\,\text{Ns/m}$ e $k = 10\,\text{N/m}$, pede-se:

   a) Construa um diagrama de blocos que represente o sistema.

   b) Simule o sistema para $f(t) = 3u(t)$ e apresente a evolução da posição vertical da massa m ao longo do tempo.

   c) Repita o item anterior, mantendo $k = 10\,\text{N/m}$, mas escolha dois valores distintos para c, um maior e outro menor. O que é possível observar em cada caso?

   d) Repita o item anterior, matendo $c = 1\,\text{Ns/m}$, mas escolha dois valores distintos para k, um maior e outro menor. O que é possível observar em cada caso?

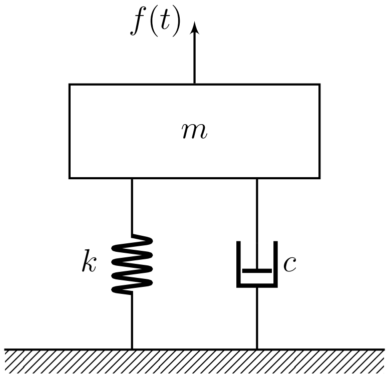


Figura 18: Sistema massa-mola-amortecedor.