

TUTORIAL 6

Instrutor: Alexandre R. Mesquita

Nosso objetivo é simular o sistema de controle da agulha de um disco rígido. Como podemos imaginar, a habilidade de posicionar a agulha com precisão sobre as faixas do disco é um fator determinante tanto para a velocidade de leitura/escrita quanto para a capacidade de armazenamento.

Em nossa simulação, usaremos um diagrama de simulação como o da Figura 1. Esse diagrama mostra os sinais E , o erro percentual de posicionamento da agulha dentro de uma determinada faixa, e I , a corrente fornecida ao motor que posiciona a agulha.

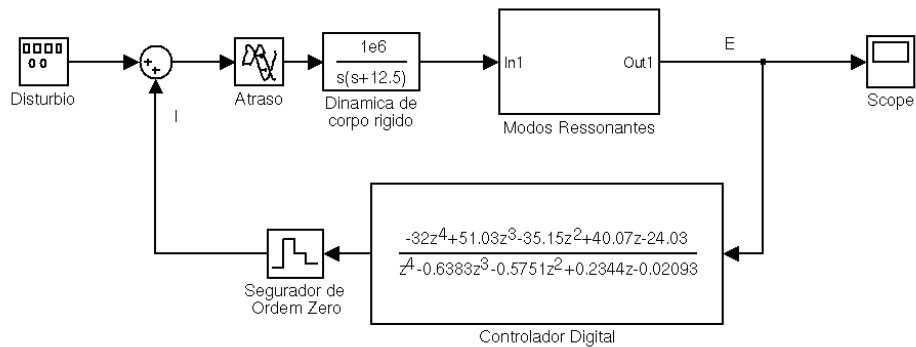


FIGURA 1. Diagrama de Simulação para o Sistema de Controle de um Disco Rígido

A função de transferência entre I e E é dada por

$$G(s) = G_R(s)G_F(s) .$$

Aqui, $G_R(s)$ representa a dinâmica da agulha quando esta é vista como um corpo rígido. Como na realidade a agulha é um objeto bastante flexível, necessitamos de um segundo termo, $G_F(s)$, para modelar sua dinâmica de maneira satisfatória. A dinâmica de corpo rígido é dada por

$$G_R(s) = e^{-1 \cdot 10^{-5}s} \frac{1 \cdot 10^6}{s(s + 12.5)}$$

Note que essa função de transferência consiste num atraso de $10\mu s$ em série com uma função de transferência racional. Para modelar o movimento de flexão da agulha, consideramos as contribuições dadas pelos quatro principais modos ressonantes que foram observados experimentalmente:

$$G_F(s) = \sum_{i=1}^4 \frac{\omega_i(a_i s + b_i)}{s^2 + 2\zeta_i \omega_i s + \omega_i^2} .$$

Essa é uma outra vantagem do uso de modelos lineares e de funções de transferência: a tarefa de obter um modelo do sistema a partir de experimentos torna-se em geral

mais fácil. Os parâmetros acima têm valores $a_1 = 1.15 \cdot 10^{-5}$, $a_2 = a_3 = 0$, $a_4 = 0.0273$, $b_1 = -5.75 \cdot 10^{-3}$, $b_2 = 2.3 \cdot 10^{-2}$, $b_3 = 8.185 \cdot 10^{-1}$, $b_4 = 1.642 \cdot 10^{-1}$, $\zeta_1 = 5 \cdot 10^{-2}$, $\zeta_2 = 5 \cdot 10^{-3}$, $\zeta_3 = 5 \cdot 10^{-2}$, $\zeta_4 = 5 \cdot 10^{-3}$, $\omega_1 = 439.82$, $\omega_2 = 1.382 \cdot 10^4$, $\omega_3 = 2.5133 \cdot 10^4$, $\omega_4 = 5.655 \cdot 10^4$.

Dessa forma, o bloco Modos Ressonantes deve possuir a estrutura mostrada na Figura 2:

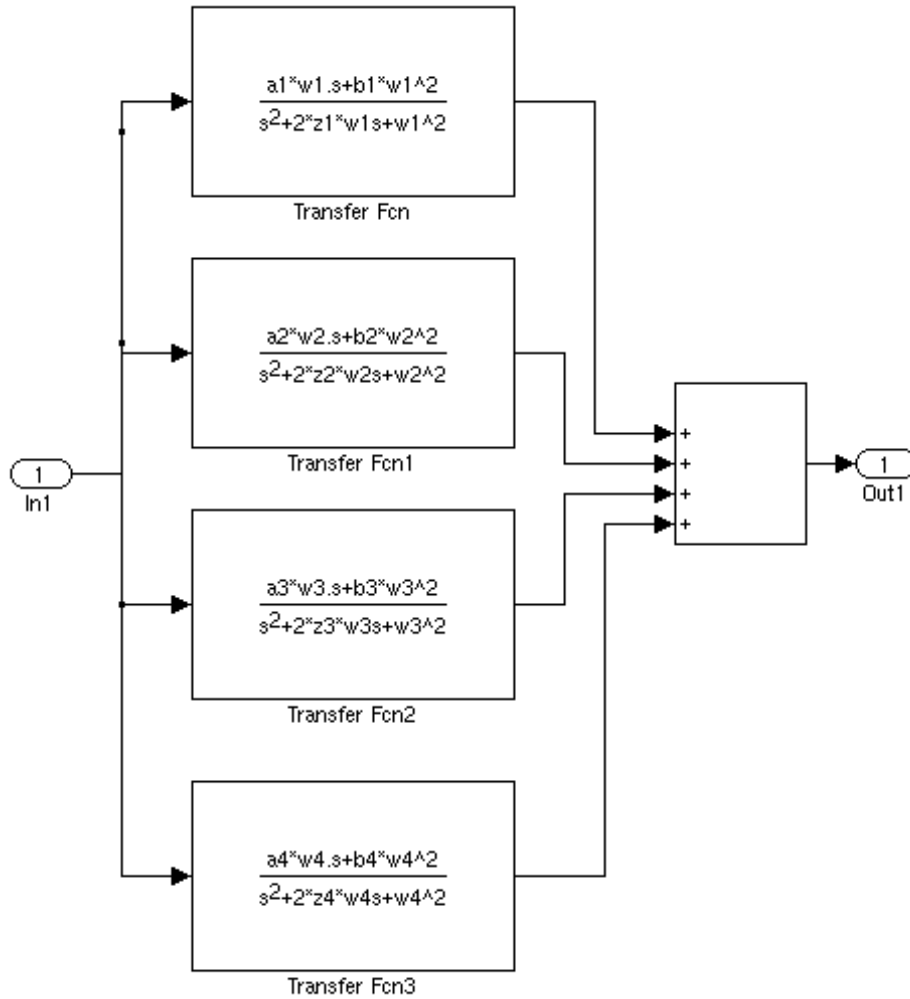


FIGURA 2. Diagrama de blocos para $G_F(s)$

O sinal de erro é monitorado por um controlador digital, que determina a corrente a ser utilizada no motor que movimenta a agulha. Como sistemas digitais operam em tempo discreto, o controlador digital é modelado como um sistema linear de tempo discreto cuja função de transferência é dada pela transformada-z:

$$C(z) = \frac{-32z^4 + 51.03z^3 - 35.15z^2 + 40.07z - 24.03}{z^4 - 0.6383z^3 - 0.5751z^2 + 0.2344z - 0.02093}$$

Esse controlador amostra o sinal de erro $E(t)$ com período de amostragem $T = 70\mu s$. O sinal discreto $E(kT)$ passa pela função de transferência $C(z)$ gerando valores $I(kT)$ para a corrente do motor. Para definir o sinal $I(t)$ a partir de $I(kT)$, usamos um interpolador de ordem zero chamado segurador de ordem zero. Esse interpolador mantém o sinal de tempo contínuo constante no valor $I(kT)$ até o próximo tempo de amostragem.

Apesar de estarmos lidando com um sistema linear, esse sistema é complexo o bastante de modo que se torna difícil encontrar uma solução analítica. Em primeiro lugar, esse sistema não possui apenas funções de transferência racionais (há um termo de atraso que corresponde a uma exponencial em s). Isso quer dizer que teríamos dificuldades em definir e manipular funções de transferência. Em segundo lugar, há subsistemas de tempo contínuo e tempo discreto interagindo. Isso quer dizer que teríamos dificuldades em usar métodos como ode45. Nesses casos, o Simulink torna-se a melhor escolha para simular o sistema.

Crie um modelo no Simulink com nome mod6a.mdl e monte o diagrama mostrado na Figura 1. Para criar funções de transferência de tempo contínuo, vá até a biblioteca Continuous. Lá você encontrará o bloco Transport Delay, que deverá ser usado para criar o bloco de atraso; o bloco Zero-Pole, que deverá ser usado para definir a parte racional da dinâmica de corpo rígido a partir dos pólos e zeros da função de transferência; e o bloco Transfer Fcn, que deverá ser usado para definir as funções de transferência dos modos ressonantes a partir dos coeficientes do numerador e do denominador.

Para criar o bloco Modos Ressonantes, o ideal é montar as funções de transferência como na Figura 2 sem as portas de entrada e saída, e então, selecionando os blocos e clicando com o botão direito, criar um subsistema. Para criar um somador com quatro entradas, basta dar um clique duplo sobre um somador e adicionar símbolos + ao campo List of Signs. Você pode tanto usar os valores numéricos dos parâmetros a_1 , b_1 , etc. diretamente na função de transferência ou definir esses parâmetros como variáveis no workspace.

Para montar a parte discreta do diagrama, vá até a biblioteca Discrete. Use o bloco Discrete Transfer Fcn para definir $C(z)$ a partir dos coeficientes de seu numerador e de seu denominador. Não se esqueça de definir o período de amostragem. O segurador de ordem zero é implementado pelo bloco Zero-Order Hold. Para esse bloco, você pode definir o período de amostragem ou colocar período -1 , que significa que o período de amostragem será herdado do bloco anterior.

Para criar o sinal de distúrbio, vá até a biblioteca Sources e utilize o bloco Signal Generator. Configure-o para gerar uma onda quadrada com amplitude 20 e frequência 0.325 Hz. O bloco Scope pode ser obtido na biblioteca Sinks e tem como função representar graficamente o sinal simulado.

No menu Simulation/Configuration Parameters, defina o tempo de simulação para 4s. Execute a simulação e observe os resultados dando um clique duplo sobre o bloco Scope.

Note que somente os últimos 0.2 s da simulação são mostrados. Isso acontece porque esse bloco está configurado para salvar no máximo 5000 pontos. Há duas soluções aqui: a primeira é aumentar o número de pontos salvos; a segunda e mais interessante é aumentar o período de amostragem do bloco Scope. Isso pode ser feito clicando-se no botão Parameters na janela do bloco Scope e mudando-se o valor do parâmetro Decimation para 200. Isso quer dizer que, de cada 200 pontos, o bloco salvará apenas 1.

Execute a simulação e observe o desempenho do sistema de controle. Note o efeito das mudanças bruscas devido ao sinal de distúrbio. Qual o tempo de recuperação do

sistema, isto é, quanto tempo é necessário para que a posição da agulha volte para a faixa de 1% de erro após a aplicação do distúrbio? Anote esse tempo no seu modelo (basta dar um clique duplo sobre qualquer região em branco para abrir uma caixa de texto).