

## CONTROLADOR LINEAR QUADRÁTICO (caso contínuo)

### 1 – INTRODUÇÃO

O Controle Linear (Ótimo) Quadrático foi desenvolvido a partir dos avanços da Teoria de Controle Moderno, com base nos trabalhos de Rudolf E. Kalman, por volta do ano de 1960, quando as representações de modelos no espaço de estados e a álgebra linear foram introduzidas na teoria de sistemas.

O Controlador Linear Quadrático, comumente referenciado na literatura como LQR (do inglês, *Linear Quadratic Regulator*), é fundamentado sobre o conceito de expressar as especificações de desempenho da malha de controle em termos de um índice matemático que quantifica o “custo” empregado na solução do problema em questão. Por isto, este índice também é designado como Função Custo e, no projeto de controladores via LQR, o problema pode ser resumido em obter a lei de controle que atenda as especificações de desempenho de malha com custo mínimo, ou seja, a lei de controle que minimiza a Função Custo.

Nesta aula, os principais tópicos serão os projetos de sistemas de controle via LQR para o caso de regulação e rastreamento de referência, ambos por realimentação total de estados de sistemas SISO contínuos, lineares e invariantes no tempo, para familiarizar os alunos da Eng. Elétrica e áreas co-relatas, com técnicas de controle Linear Quadrático.

### 2 – LQR COM REALIMENTAÇÃO TOTAL DE ESTADOS

Dificilmente todas as variáveis de estado de um sistema real são variáveis medidas. Normalmente, as variáveis de estado são inferidas a partir de relações com a variável medida de saída do sistema. No entanto, há como estimar as variáveis de estado utilizando os observadores de estado estudados em aulas anteriores, mas a princípio, e para simplificar a apresentação do controle LQR, consideraremos que todas as variáveis de estado do modelo de projeto estão disponíveis, de tal forma que

$$\dot{x} = Ax + Bu \quad (1)$$

$$y = Cx \quad (2)$$

onde  $x \in \mathbf{R}^n$ ,  $u \in \mathbf{R}^m$ ,  $y \in \mathbf{R}^p$ , e  $p = n$ , logo  $C = I_{n \times n}$ .

O problema de controle LQR é obter a lei de controle por realimentação de estados,

$$u = -Kx + y_r, \quad (3)$$

que minimiza a Função Custo

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt. \quad (4)$$

A lei de controle em (3), apresenta  $y_r$ , que é a variável de referência e que no problema de regulação é igual a zero. Por ser mais simples, vamos considerar referência nula durante o desenvolvimento teórico e apresentar o caso de rastreamento na forma de exemplos com simulação.

Na equação (4), a Função Custo pode ser entendida pela explicação do funcionamento de  $Q$  e  $R$  no projeto do controlador, onde no caso SISO,  $Q$  é uma matriz  $n \times n$  semidefinida positiva que pondera as variáveis de estado e  $R$  um escalar que pondera o sinal de controle. As magnitudes relativas de  $Q$  e  $R$  vão governar o compromisso entre desempenho e custo reduzido, conforme:

- $R$  muito grande = ação de controle conservativa (*cheap control*);
- $R$  pequeno = ação de controle com alto consumo energético;
- $Q$  grande = os estados tendem a zero rapidamente;
- $Q$  pequeno = os estados tendem a zero lentamente.

Para calcular o ganho de realimentação  $K_{m \times n}$  da lei de controle em (3), primeiro é necessário resolver a Equação Algébrica de Riccati de controlador,

$$0 = A^T P + P A + Q - P B R^{-1} B^T P, \quad (5)$$

onde  $P$  é uma matriz semidefinida positiva. A solução é encontrar a matriz  $P$  que faça o lado direito da equação (5) igual a zero. Dependendo da complexidade e ordem do sistema, encontrar uma  $P$  que satisfaça a equação (5) é um procedimento difícil sem o auxílio de softwares específicos. Mais adiante, veremos como calculá-la utilizando o MATLAB.

De posse da  $P$ , o ganho ótimo de realimentação de estados é calculado por

$$K = R^{-1} B^T P. \quad (6)$$

O sistema em malha fechada pode ser avaliado a partir da substituição da lei de controle no modelo do processo, ficando então

$$\begin{aligned} \dot{x} &= (A - BK)x + B y_r \\ y &= Cx \end{aligned} \quad (7)$$

### 3 – EXEMPLO 1: REGULAÇÃO DE VELOCIDADE DE UM MOTOR CC

Neste exemplo de projeto e simulação de um regulador de velocidade de um motor CC com LQR, usaremos o seguinte modelo de planta:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{s+1} \quad (8)$$

A planta em (8) é controlada por uma entrada em tensão de -5~5 V e a saída é medida também em tensão através de um gerador tacométrico na mesma faixa, -5~5 V. O modelo da planta no espaço de estados pode ser representado por

$$\begin{aligned} \dot{x}_1 &= -x_1 + u, \\ y &= x_1. \end{aligned} \quad (9)$$

Para um sistema simples como o apresentado em (9), o cálculo da  $P$  é facilmente obtido. Deste modo, considerando  $Q=1$  e  $R=20$ ,

$$\begin{aligned} 0 &= (-1)P + P(-1) + (1) - P(1)(20)^{-1}(1)P, \\ P &= 0.4939, \end{aligned}$$

fornecendo o ganho de realimentação de estados  $K = 0.0247$ . Os sinais de controle e de saída do sistema são apresentados na Figura 1. Na mesma figura também foram plotados os gráficos para uma ponderação bem menor do sinal de controle, com  $R=0.2$ . Veja que o consumo de energia é maior quando  $R$  é pequeno, evidenciado pelo gráfico do sinal de controle na Figura 1, portanto, o custo é maior.

Este exemplo pode ser reproduzido no MATLAB usando o código da Tabela 1.

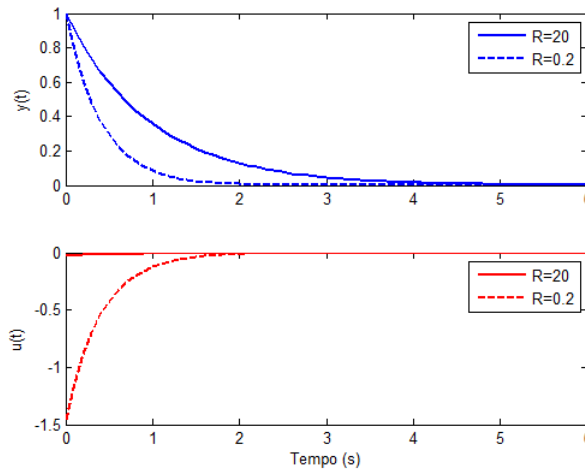


Figura 1. Influência da variação de  $R$  na malha de controle.

Tabela 1. Código em MATLAB do Exemplo 1.

```
A=-1; B=1; C=1; Q=1; R=20;
[K,P] = lqr(A,B,Q,R);
sys=ss(A-B*K,B,C,0);
x0=[1]; T=0:0.1:6; yr=zeros(1,length(T));
[y,t,x] = lsim(sys,yr,T,x0);
u=-K*x;
```

Apesar de simples, o exemplo apresentado ressalta uma importante característica sobre o controle LQR, que é a herança da complexidade do modelo na estrutura do

controlador. Este é um aspecto positivo quando não se conhece qual a melhor estrutura de controle a ser utilizada para um determinado problema, mas pode se tornar um entrave quando se usa modelos de projeto de ordem elevada, pois a estrutura do controlador também aumenta assim como a complexidade da sintonia.

#### 4 – EXEMPLO 2: CONTROLE DE POSIÇÃO DE UM PÊNDULO AMORTECIDO

Neste exemplo vamos utilizar um modelo de segunda ordem que representa a dinâmica de um pêndulo amortecido. Na Figura 2 apresenta-se o diagrama da planta, que de forma resumida, mede o deslocamento angular do braço do pêndulo a partir de um potenciômetro ligado no ponto de pivô, acarretando uma variação de tensão no terminal de saída, que pode variar de 0~5V. O braço do pêndulo é movimentado quando uma tensão de entrada (também de 0~5V) é aplicada no motor CC que compõe um sistema propulsor juntamente com uma hélice no eixo do motor. O problema de controle em questão consiste em posicionar o braço do pêndulo em um ângulo (proporcional a tensão medida) desejado. A variável de referência,  $y_r$ , é utilizada.

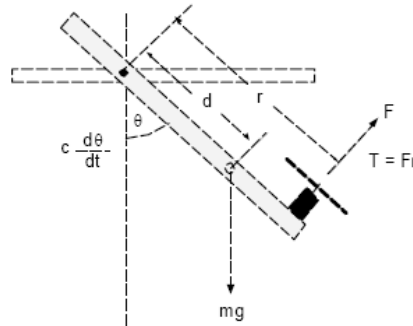


Figura 2. Diagrama do processo pêndulo amortecido.

A função de transferência que descreve a dinâmica sub-amortecida do sistema é

$$G(s) = \frac{Y(s)}{U(s)} = \frac{3.48}{s^2 + 0.8s + 4},$$

e uma representação no espaço de estados pode ser dada por

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 0 & 1 \\ -4 & -0.8 \end{bmatrix} x + \begin{bmatrix} 0 \\ 3.48 \end{bmatrix} u \\ y &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x \end{aligned}$$

onde estamos considerando que todos os estados são medidos e que  $x_1$  e  $x_2$  são posição e velocidade angular (dadas em sinais de tensão), respectivamente.

Para garantir que o controlador possa modificar todos os estados, evidentemente o sistema precisa ser controlável. Isto é verdadeiro para o modelo dado.

No Exemplo 1, deu-se prioridade em explorar como  $R$  agia sobre a sintonia do controlador. Já neste exemplo, há maior flexibilidade, pois a matriz  $Q$  vai modificar a ponderação em relação a posição e a velocidade. Usando uma

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 0.01 \end{bmatrix} \text{ e } R = 1,$$

praticamente despreza-se a ponderação da velocidade, pois utiliza-se um pequeno valor (lembre-se que a ponderação de  $Q$  com baixo valor faz com que o estado tenda a zero lentamente) e prioriza-se a variável de posição. A Figura 3 reflete esta característica através das respostas transitórias oscilatórias dos estados para uma referência do tipo degrau unitário.

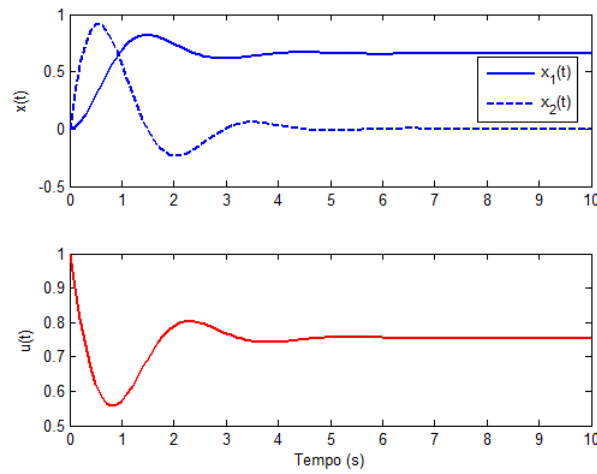


Figura 3. Velocidade com baixa ponderação e conseqüentes oscilações.

Repetindo-se o projeto do controlador, com um aumento da ponderação da velocidade para 1, e mantendo a  $R$ , verifica-se na Figura 4 que o controlador LQR passou a “segurar” mais a variável  $x_2$  e conseqüentemente eliminou as oscilações do sistema.

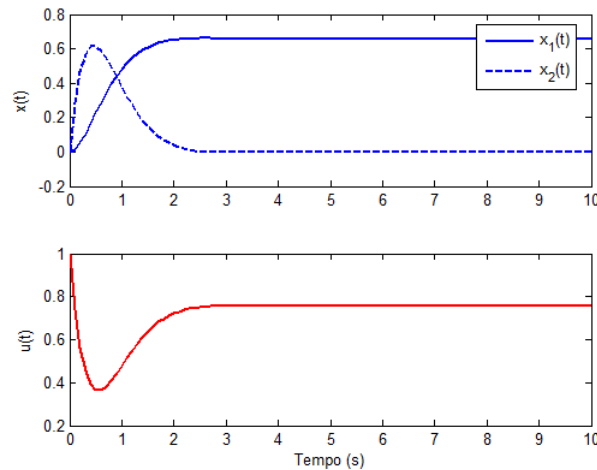


Figura 4. Sobre-sinal eliminado após aumento da ponderação da velocidade.

Apesar de o controlador projetado conseguir amortecer as oscilações do sistema, não foi capaz de eliminar o erro em regime permanente, pois o controlador não possui integrador. No entanto, seria possível fazer um ajuste fino da atual sintonia a partir do valor de  $R$ , reduzindo-o para fornecer mais energia ao sistema controlado, mas isto não eliminaria totalmente o erro de rastreamento. Além disso, sabemos que a natureza não é tão gentil com o engenheiro como um modelo linear invariante no tempo pode ser, portanto, em um processo real existiriam não-linearidades, como variação do ganho estático em cada ponto de operação do sistema, que sempre acarretariam em erro estático não nulo.

Como mencionado anteriormente, a estrutura do controlador é fornecida de forma natural pela técnica de projeto e é herdada a partir do modelo. Sendo assim, para incluir a ação integral na lei de controle do LQR, basta que o modelo de projeto seja aumentado com a adição de um integrador da mesma forma como apresentamos em aulas anteriores, ou seja, pode-se utilizar

$$\dot{x}_a = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} x_a + \begin{bmatrix} 0 \\ I \end{bmatrix} \dot{u}, \quad x_a = \begin{bmatrix} x \\ u \end{bmatrix} \quad (10)$$

para aumentar o sistema com integrador e passar esta característica ao controlador. A Figura 5 apresenta os resultados de simulação com o projeto baseado na equação (10) e a ponderação dos estados dada por

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

ou seja, com integrador livre de ponderação. O código em MATLAB da Tabela 2 pode ser utilizado para reproduzir as simulações apresentadas neste exemplo.

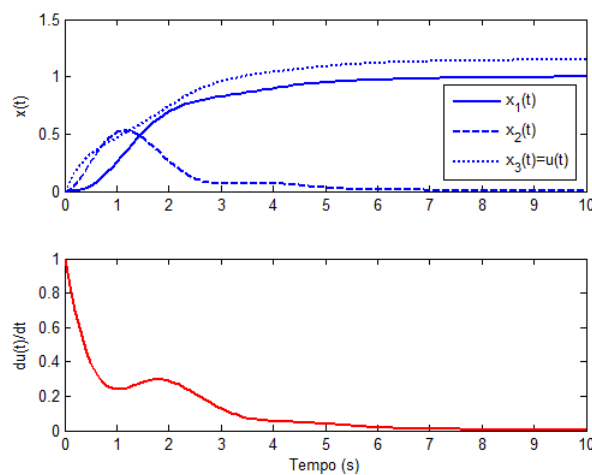


Figura 5. Projeto baseado em modelo aumentado com integrador.

Tabela 2. Código em MATLAB do Exemplo 2.

```
A=[0 1; -4 -0.8]; B=[0; 3.48]; C=eye(2);
AA=[A B; 0 0 0]; BB=[0;0;1]; CC=eye(3);
Q=diag([1 1 0]); R=1;
[K,P] = lqr(AA,BB,Q,R);
sys=ss(AA-BB*K,BB,CC,0);
x0=[0;0;0]; T=0:0.1:10; yr=ones(1,length(T));
[y,t,x] = lsim(sys,yr,T,x0);
u=-K*x'+yr;
```

**Obs: Modelo de estados aumentado com integrador: entendendo o modelo e a obtenção do diagrama de blocos:**

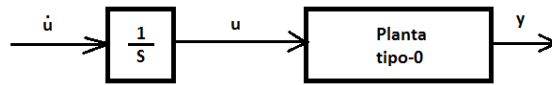


Figura 2. Planta com descrição aumentada pela inclusão de integradores nas entradas.

$$\begin{bmatrix} \dot{x} \\ \dot{u} \end{bmatrix} = \underbrace{\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}}_{A_a} \underbrace{\begin{bmatrix} x \\ u \end{bmatrix}}_{x_a} + \underbrace{\begin{bmatrix} 0 \\ I \end{bmatrix}}_{B_a} \dot{u} \quad (11)$$

$$u_a = \dot{u} \quad (12)$$

$$u_a = -Kx_a + y_r = -\begin{bmatrix} K_x \\ K_u \end{bmatrix}^T \begin{bmatrix} x \\ u \end{bmatrix} + y_r \quad (13)$$

$$u_a = \dot{u} = -K_x x - K_u u + y_r \quad (14)$$

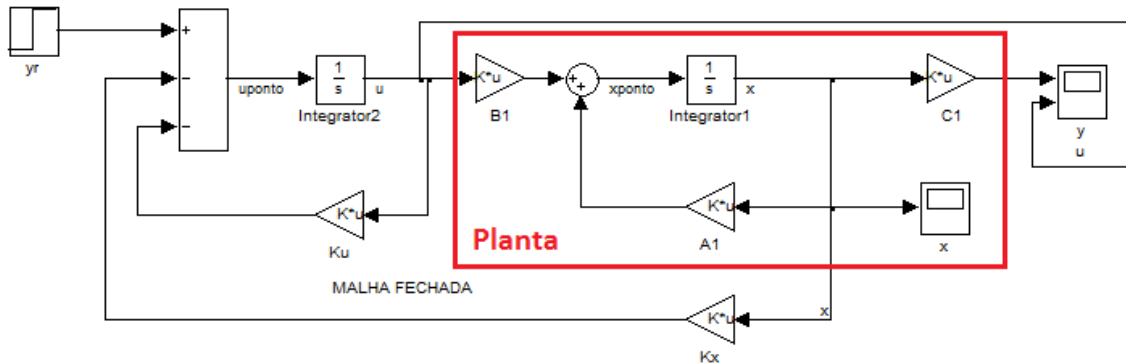


Figura 3. Diagrama de Blocos do modelo nominal controlado pelo controlador LQR com modelo aumentado.