

# Controladores LQR aplicados ao Robô Móvel Skid-Steer com Rodas para Rastreamento de Trajetórias

Lorena Santos Vilas Boas \* Andre G. S. Conceição \*\*

LaR - Laboratório de Robótica, Departamento de Engenharia Elétrica e de Computação, Universidade Federal da Bahia, BA, Brasil.

\* E-mail: [lorennavilasb@gmail.com](mailto:lorennavilasb@gmail.com)

\*\* E-mail: [andre.gustavo@ufba.br](mailto:andre.gustavo@ufba.br)

---

## Abstract:

The navigation of a mobile robot in an environment for trajectory tracking is an important application for this class. In this context, this work aims to study the implementation of a controller for trajectory tracking using the Linear Quadratic Regulator (LQR) as a state feedback control method applied to mobile skid-steering robots with wheels. Two LQR controllers are designed from two distinct state-space representations, both considering a virtual reference robot for linearization of the problem. The AGV Husky robot will be used as a test platform, resulting in a study for mobile robot navigation at the GAZEBO simulation tool.

**Resumo:** O controle da movimentação do robô móvel em um ambiente seguindo uma trajetória delimitada é uma aplicação importante para essa classe. Nesse contexto, este trabalho tem como objetivo a implementação do controlador para rastreamento de trajetória utilizando o Regulador Linear Quadrático (LQR) aplicado à classe de robôs móveis do tipo skid-steer com rodas. Dois controladores LQR são projetados, a partir de dois modelos distintos no espaço de estados, ambos considerando um robô virtual de referência para linearização do problema. O robô AGV Husky será utilizado como plataforma de teste, resultando em um estudo para navegação de robôs móveis no ambiente de simulação GAZEBO.

*Keywords:* skid-steer robot; mobile robot; state feedback control; linear quadratic regulator; trajectory tracking.

*Palavras-chaves:* robô skid-steer; robô móvel; controle por realimentação de estados; regulador linear quadrático; rastreamento de trajetória.

---

## 1. INTRODUÇÃO

A robótica móvel tem sido amplamente utilizada em diferentes áreas, para além do campo industrial, tais como agricultura, mineração, inspeção, veículo auto-operado e logística (Santos, 2017). Nesse contexto, este trabalho visa apresentar uma solução para o controle da movimentação de robôs móveis considerando o problema de rastreamento de trajetória.

Conforme analisado em Boas and Conceicao (2020), o robô móvel do tipo skid-steer possui uma característica inerente a sua operação, que é o deslizamento das suas rodas em execuções de curvas, o que dificulta a odometria e o controle de movimento se comparado a outras configurações, como o robô do tipo diferencial ideal de duas rodas. Por esta razão, essa classe de robôs será utilizada como objeto de estudo para o problema de rastreamento de trajetória considerado neste artigo.

Dentre as referências consultadas neste estudo, a técnica de rastreamento de trajetória a partir do erro de posição entre o robô controlado e o robô de referência virtual foi citada primeiramente por Klančar et al. (2005), o qual propõe,

para o controle por realimentação, que a determinação dos ganhos do controlador seja feita através da comparação entre os polinômios real e desejado característico. Já em Klančar and Skrjanc (2007), o autor propõe, a partir da mesma técnica de rastreamento e do mesmo modelo no espaço de estados, a implementação de um Controle preditivo baseado em modelo (MPC).

Kuhne (2005), por sua vez, propôs um algoritmo linear de MPC para a solução do problema de rastreamento de trajetória. O autor afirma que, como a referência é conhecida para qualquer instante de amostragem futuro, é possível obter uma descrição linear e variante no tempo do modelo do sistema, através de linearizações sucessivas ao longo da trajetória de referência. Assim, o problema de otimização de um MPC pode ser transformado em um problema de programação quadrática.

Neste trabalho, optou-se por implementar o controle no espaço de estados utilizando o Regulador Linear Quadrático (LQR) para o rastreamento de trajetória, a partir da técnica de seguimento do robô de referência virtual. Contudo, ambos os modelos no espaço de estados propostos por Klančar et al. (2005) e Kuhne (2005) serão utiliza-

dos, resultando em dois modelos cinemáticos distintos, e, conseqüentemente, dois controladores LQR diferentes.

Portanto, o presente artigo está estruturado da seguinte forma: na seção 2 será englobada uma explicação sobre o controlador LQR e a incorporação da ação integral ao modelo aumentado no espaço de estados. Em seguida, na seção 3, são detalhados os modelos no espaço de estados propostos para o problema de rastreamento de trajetória. Por fim, na seção 4, serão descritos os resultados alcançados com a aplicação dos controladores ao robô AGV Husky, seguidos pelas considerações finais, incluindo trabalhos futuros, e as referências.

## 2. O CONTROLADOR LQR

Considere o modelo discreto no espaço de estados definido em (1), em que  $A_d$ ,  $B_d$  e  $C_d$  constituem, respectivamente, as matrizes de estado, de entrada e de saída e  $x[k]$ ,  $u[k]$  e  $y[k]$  os vetores de estado, de entrada e de saída.

$$\begin{cases} x[k+1] = A_d x[k] + B_d u[k] \\ y[k] = C_d x[k] \end{cases} \quad (1)$$

Partindo do princípio de que todos os estados do sistema são mensuráveis, Uma lei de controle por realimentação de estados pode ser implementada usando o estado atual do sistema para determinar o sinal de controle, isto é,

$$u[k] = f(x[k]) = -\kappa x[k] \quad (2)$$

Dada a equação do sistema (1), o problema do regulador quadrático ótimo consiste em determinar a matriz  $\kappa$  do vetor de controle ótimo em (2) de forma a minimizar a função de custo definida em (3). Uma propriedade importante do LQR é que se houver um ganho que estabilize o sistema, então haverá uma solução válida para o LQR. Outra vantagem é que a solução resultante do LQR é estática, o que resulta em um baixo custo computacional, já que a matriz  $\kappa$  é calculada apenas uma vez no processo de design do controlador (Santos, 2017).

$$J(\mathbf{u}) = \sum_{k=0}^{\infty} (x[k]^T Q x[k] + u[k]^T R u[k]) \quad (3)$$

sendo  $\mathbf{Q} = \mathbf{Q}^T$  uma matriz semidefinida positiva e  $\mathbf{R} = \mathbf{R}^T$  definida positiva. Essas matrizes determinam a importância relativa do erro e o consumo da energia dos sinais de controle (Ogata, 2010). A solução  $\kappa$  para o problema de minimização da função de custo é dada por (4), sendo  $S$  a solução para a equação de Riccati no tempo discreto (5).

$$\kappa = (B_d^T S B_d + R)^{-1} B_d^T S A_d \quad (4)$$

$$A_d^T S A_d - S - A_d^T S B_d (B_d^T S B_d + R)^{-1} B_d^T S A_d + Q = 0 \quad (5)$$

Considerando que o controlador LQR consiste em um regulador proporcional simples, ele não garante que em regime permanente o erro será nulo na presença de perturbações. Uma alternativa a este problema é incorporar a ação integral ao modelo no espaço de estados do sistema,

constituindo um modelo aumentado, como utilizado em Wang (2009). Esse modelo pode ser obtido a partir do incremento da variável de estado ( $\Delta x[k] = x[k] - x[k-1]$ ) e do sinal de controle ( $\Delta u[k] = u[k] - u[k-1]$ ). Realizando algumas manipulações algébricas em (1), uma nova variável de estado é definida como sendo o vetor  $\xi[k] = [\Delta x[k]^T y[k]^T]^T$  e o modelo aumentado no espaço de estados resultante é dado por (6) (Wang, 2009).

$$\begin{cases} \xi[k+1] = A_a \xi[k] + B_a \Delta u[k] \\ y[k] = C_a \xi[k] \end{cases} \quad (6)$$

onde as matrizes  $A_a$ ,  $B_a$  e  $C_a$  caracterizam o modelo no espaço de estados aumentado, o qual possui a ação integral é incorporada. Elas são definidas, com relação às matrizes do modelo (1), como:

$$A_a = \begin{bmatrix} A_d & 0 \\ C_d A_d & I \end{bmatrix}, B_a = \begin{bmatrix} B_d \\ C_d B_d \end{bmatrix}, C_a = [0 \ I]. \quad (7)$$

## 3. MODELO NO ESPAÇO DE ESTADOS PARA ROBÔ DE ACIONAMENTO DIFERENCIAL

Os robôs da classe *skid-steer* com rodas, por serem veículos com acionamento diferencial, possuem restrição não-holonômica, já que as rodas acionadas apenas giram para frente ou para trás e não deslizam lateralmente (Figueiredo and Jota, 2004). Considerando  $v$  e  $w$ , respectivamente a velocidade tangencial e angular do robô, como as variáveis de estado internas, as equações de movimentação do robô estão descritas em (8).

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (8)$$

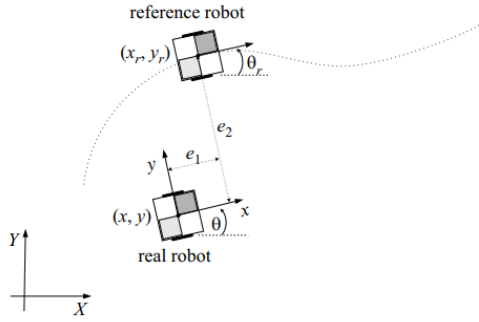
Nesse contexto, visando o problema do rastreamento de trajetória, nas seções 3.1 e 3.2 foram propostas duas alternativas para o modelo no espaço de estados a ser utilizado para a implementação do controlador LQR, ambos considerando a existência de um robô referência virtual, como na Figura 1. Nessa proposta, um controle *feedforward*, também chamado de pré-alimentação, é obtido considerando que não há distúrbios e/ou erro inicial, de modo que as velocidades tangencial ( $v_r$ ) e angular ( $w_r$ ) do robô referência podem ser calculadas pelas equações (9) e (10), as quais consistem também na entrada do controle do robô referência, conforme modelo definido em (8) (Klancar et al., 2005).

$$v_r(t) = \pm \sqrt{\dot{x}_r^2(t) + \dot{y}_r^2(t)} \quad (9)$$

$$w_r(t) = \frac{\dot{x}_r(t)\ddot{y}_r(t) - \dot{y}_r(t)\ddot{x}_r(t)}{\dot{x}_r^2(t) + \dot{y}_r^2(t)} \quad (10)$$

A equação (10) foi obtida a partir da variação no tempo da equação (11), em que  $k=0$  define a trajetória direcionada para frente e  $k=1$  para trás.

$$\theta_r(t) = \arctan 2(\dot{y}_r(t), \dot{x}_r(t)) + k\pi \quad (11)$$



$$\begin{aligned} u_1 &= v_r \cos(e_3) - v_c \\ u_2 &= w_r - w_c \end{aligned} \quad (14)$$

Figura 1. Robô real seguindo robô de referência por erro de transformação. (Klancar et al., 2005)

Portanto, a partir das relações de (9) a (11), pode-se calcular as entradas de (8), para uma trajetória de referência  $q_r(t) = [x_r(t), y_r(t), \theta_r(t)]$ . Uma condição necessária para utilizar esta estratégia é que a trajetória deve ser duas vezes diferenciável e a velocidade tangencial não pode ser igual a zero ( $v_r(t) \neq 0$ ), pois, se isto acontecer, o robô irá rotacionar em um ponto fixo com a velocidade angular  $w_r(t)$  e  $\theta_r(t)$  não poderá ser calculado por (11) (Klancar et al., 2005).

### 3.1 Modelo baseado no erro de rastreamento

Em suas contribuições, Klancar et al. (2005) considera que, quando um robô é controlado para seguir uma trajetória de referência, naturalmente existem erros entre a posição e orientação do robô real e o robô virtual, o qual segue a posição e orientação ideais, como consta na Figura 1. Sendo assim, o modelo no espaço de estados é construído tendo como estado os erros que surgem entre ambos os robôs como consta em (12):

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y_r - y \\ \theta_r - \theta \end{bmatrix}. \quad (12)$$

Em que  $e_x = (x_r - x)$ ,  $e_y = (y_r - y)$  e  $e_\theta = (\theta_r - \theta)$  caracterizam os erros calculados na base de coordenadas do mundo. Derivando a função erro (12), considerando a cinemática do robô expressa em (8), e linearizando o sistema obtido em torno da trajetória de referência ( $e_1 = e_2 = e_3 = 0$  e  $v_c = w_c = 0$ ), Klancar et al. (2005) propôs o seguinte modelo linear para o problema de rastreamento de trajetória:

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = \begin{bmatrix} 0 & w_r & 0 \\ -w_r & 0 & v_r \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_c \\ w_c \end{bmatrix} \quad (13)$$

onde  $v_c$  e  $w_c$  correspondem ao sinal de controle resultante do sistema em malha fechada (*feedback*), ou seja, é a saída do controlador LQR neste trabalho, e  $v_r \cos(e_3)$  e  $w_r$  são as entradas *feedforward*, obtidas através de uma transformação não linear das entradas de referência, compondo o vetor do sinal de entrada do sistema ( $u_1$  e  $u_2$ ), definidos por (14):

Analisando a matriz de controlabilidade  $[B_c \ A_c B_c \ A_c^2 B_c]$  a partir de (13), pode-se observar que o posto é completo apenas se  $v_r$  ou  $w_r$  forem diferentes de zero. Sendo assim, uma condição suficiente para a controlabilidade do sistema é que as entradas *feedforward* sejam sempre constantes, o que é válido apenas para trajetórias lineares ou circulares Klancar et al. (2005).

### 3.2 Modelo Linearizado

A fim de obter um modelo linearizado para a solução do problema de rastreamento de trajetória, a estratégia adotada por Kuhne (2005) foi considerar um modelo de erro entre o robô e o robô de referência virtual. Considerando que o robô segue o modelo cinemático representado em (8), expandindo o lado direito do modelo em séries de Taylor em torno do ponto  $(\mathbf{x}_r, \mathbf{u}_r)$  e descartando os termos de ordem mais alta, tem-se que:

$$\dot{\tilde{x}} = f_{x,r} \tilde{x} + f_{u,r} \tilde{u} \quad (15)$$

sendo  $\tilde{x} = x - x_r$  o erro em respeito ao robô de referência e  $\tilde{u} = u - u_r$  o termo associado à perturbação da entrada no controle do sistema. Discretizando a expressão (15) pelo método de Euler, tem-se o modelo linear variante no tempo descrito em (16), sendo  $\tilde{x}[k] = x[k] - x_r[k]$ ,  $\tilde{u}[k] = u[k] - u_r[k]$  e  $T$  o período de amostragem (Kuhne, 2005).

$$\tilde{x}[k+1] = A[k] \tilde{x}[k] + B[k] \tilde{u}[k], \quad (16)$$

$$A[k] = \begin{bmatrix} 1 & 0 & -v_r[k] \sin(\theta_r[k]) T \\ 0 & 1 & v_r[k] \cos(\theta_r[k]) T \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

$$B[k] = \begin{bmatrix} \cos(\theta_r[k]) T & 0 \\ \sin(\theta_r[k]) T & 0 \\ 0 & T \end{bmatrix} \quad (18)$$

O modelo linearizado acima, deduzido por Kuhne (2005), pode ser utilizado para o problema de rastreamento de trajetória de um robô móvel não holonômico. O autor afirma que "essa linearização torna-se controlável à medida que o robô permanece em movimento". Isto é, para que o posto da matriz de controlabilidade deste modelo seja completo, a velocidade linear de referência  $v_r$  deve ser não nula. No entanto, a velocidade angular de referência  $w_r$  não precisa ser diferente de zero, o que permite que o robô siga trajetórias retas (Kuhne, 2005).

O modelo obtido foi utilizado por Kuhne (2005) para a implementação de um controlador MPC. Por outro lado, neste trabalho esse modelo será aplicado ao controlador LQR e essa diferença de aplicação é um ponto crucial a ser considerado.

Como afirma Santos (2017), o LQR é um controlador ótimo que "resolve o problema de otimização em um horizonte de tempo infinito, buscando o ganho que minimiza a função de custo". Dessa forma, apesar da lei de controle

resultante do LQR ser estática, espera-se que a realimentação atue no sistema para compensar os erros de modelagem ou perturbações externas.

No caso deste trabalho, por tratar-se de um problema de rastreamento de trajetória, naturalmente os parâmetros  $v_r[k]$  e  $\theta_r[k]$ , necessários para o cálculo dos elementos das matrizes do modelo, sofrerão alterações ao longo do tempo. Contudo, se considerarmos um valor inicial ( $k=1$ ) para ambos os parâmetros e forçando uma aproximação de que este valor será constante, obtemos um modelo possível de ser aplicado ao LQR. No entanto, consequentemente espera-se que este modelo tenha um desempenho de baixa precisão, devido ao erro de modelagem referente à aproximação estabelecida. A fim de reduzir esse erro, optou-se por incorporar a ação integral nessa técnica, gerando, portanto, o LQR a partir do modelo aumentado no espaço de estados, conforme consta a equação (6).

#### 4. ESTUDO DE CASO: APLICAÇÃO AO ROBÔ AGV HUSKY

Neste trabalho, o robô móvel utilizado como objeto de estudo é o AGV Husky da Clearpath. Conforme registro na Figura 2, é um veículo terrestre não-tripulado com 4 rodas, incluso na classe *skid-steer*. Sua base é equipada com *encoders* de quadratura de alta resolução com 78.000 pulsos/m, sendo sua velocidade máxima de 1 m/s. Seu controle pode ser realizado através de C++, *Python* e ROS.



Figura 2. Robô AGV Husky.

A arquitetura de controle para o problema de rastreamento de trajetória aplicado ao robô AGV Husky segue o esquema representado na Figura 3. No planejamento de trajetória, a referência foi gerada em formato lemniscata, a qual foi modelada matematicamente e parametrizada em função do tempo. O controle é implementado em cascata, sendo a malha interna realizada pelo controlador do próprio ROS, o qual tem como *set point* as velocidades tangencial e angular fornecidas pela malha externa e a saída consiste nas velocidades das rodas do lado esquerdo e direito do robô. Já o controlador da malha externa é implementado a partir dos estudos deste artigo, que resultaram em um LQR projetado para os dois modelos descritos nas seções 3.1 e 3.2.

A lei de controle resultante do LQR tem como objetivo minimizar o desvio do estado com relação ao ponto de equilíbrio, a partir da ponderação da matriz  $\mathbf{Q}$ , bem como minimizar a energia utilizada no controle, considerando o ajuste da matriz  $\mathbf{R}$ . O cálculo da matriz de ganho  $\kappa$  foi realizado utilizando o *software* Matlab.

Os controladores LQR projetados neste trabalho foram desenvolvidos através da linguagem de programação Python e implementados como um pacote do ROS, sendo o envio e recebimento de dados com o robô realizados por meio dos tópicos do ROS. Para respeitar os limites operacionais do robô, as velocidades resultantes do sinal de controle foram limitadas, sendo  $|v| < 0,8 \text{ m/s}$  e  $|w| < 1 \text{ rad/s}$ . Os resultados apresentados neste trabalho foram obtidos através da aplicação dos controladores propostos ao AGV Husky utilizando o simulador Gazebo.

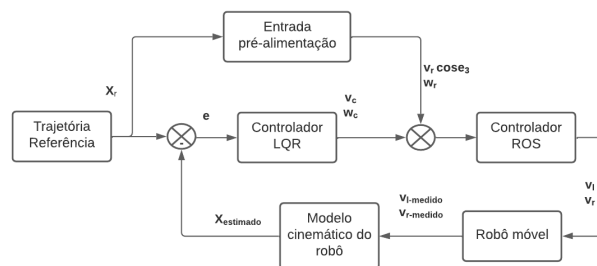


Figura 3. Arquitetura do controle para rastreamento de trajetória.

##### 4.1 Resultados obtidos com simulador Gazebo

Inicialmente foram calculados 13 ganhos diferentes para o controlador LQR a partir do modelo baseado no erro de rastreamento. Os diferentes ganhos foram obtidos através das variações das matrizes de ponderação, justamente para permitir uma análise comparativa mais completa da performance do controlador.

O melhor desempenho do seguimento ocorreu com o  $LQR_1$ , sendo as matrizes  $\mathbf{Q}_1$  e  $\mathbf{R}_1$  ajustadas conforme (19),  $v_r = 0,3 \text{ m/s}$  e  $w_r = \pm 0,2 \text{ rad/s}$ , e  $\kappa_1$  resultante descrito em (20) (ver Figuras 4 a 7). Essa configuração apresentou um controle menos agressivo, com variações mais suaves no sinal de controle e mantendo a velocidade tangencial positiva ao longo de toda a trajetória, o que viabiliza a aplicação ao robô real, por respeitar os limites físicos de resposta do veículo. Dessa forma, o controlador LQR atua principalmente na compensação do sinal de controle, de forma a garantir a resposta do sistema ao aparecimento de erros de diferentes naturezas, tais como por aspectos dinâmicos ou outras características não previstas no modelo.

$$\mathbf{Q}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0,5 \end{bmatrix}, \mathbf{R}_1 = 0,1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (19)$$

$$\kappa_1 = \begin{bmatrix} 2,37 & -0,797 & -0,11 \\ -0,1333 & 2,177 & 2,07 \end{bmatrix} \quad (20)$$

Foi observado também a influencia das variações das matrizes  $\mathbf{Q}$  e  $\mathbf{R}$  no desempenho do controle. Com a matriz  $\mathbf{Q}$  fixa, foi possível analisar que, quanto maior o valor dos elementos da diagonal principal de  $\mathbf{R}$ , menos agressivo é o controlador e, consequentemente, um erro maior é admitido ao sistema. Por outro lado, ao fixar a matriz  $\mathbf{R}$ , o elemento da diagonal principal de  $\mathbf{Q}$  com o maior valor corresponde à variável que provocará maior sensibilidade no sinal de controle.

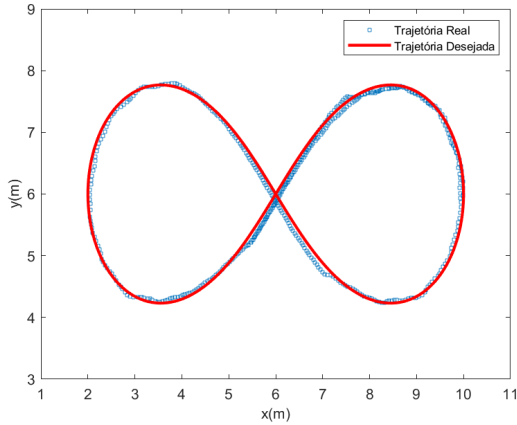


Figura 4. Traj. Lemniscata ( $LQR_1$ ).

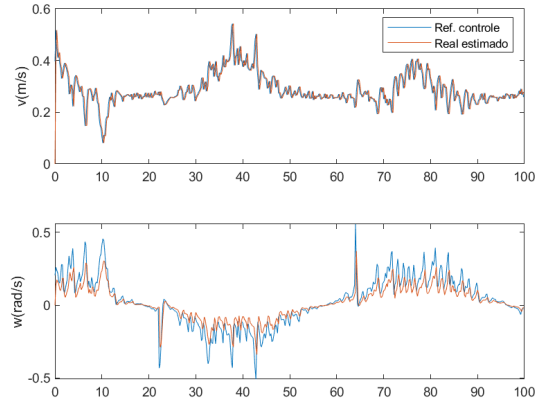


Figura 7. Sinal de controle ( $LQR_1$ ).

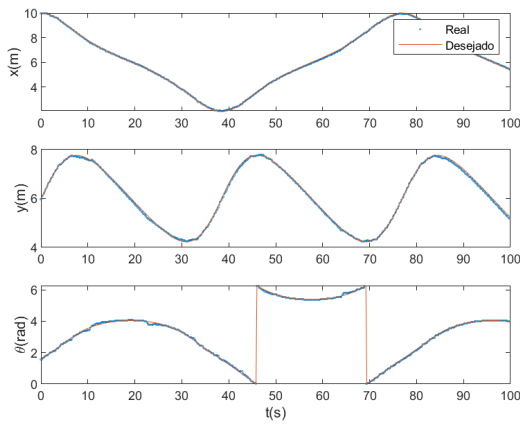


Figura 5. Pose do robô ( $LQR_1$ ).

$$\mathbf{Q}_2 = 0,01 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0,1 \end{bmatrix}, \mathbf{R}_2 = 0,01 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (21)$$

$$\kappa_2 = \begin{bmatrix} 4,777 & 0 & 0,045 & 0,537 & 0,536 & -0,003 \\ 0,077 & 0 & 2,245 & -0,036 & 0,048 & 0,278 \end{bmatrix} \quad (22)$$

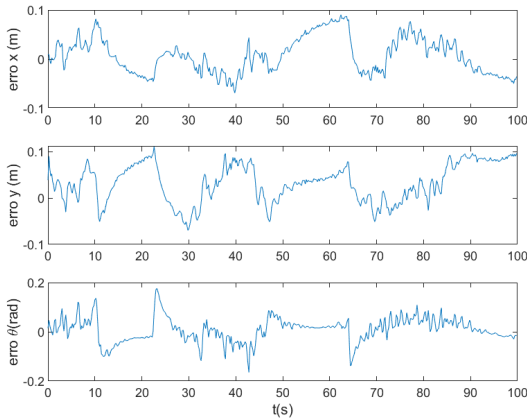


Figura 6. Sinal de erro ( $LQR_1$ ).

O controlador  $LQR_2$  baseado na versão aumentada do modelo linearizado proposto por Kuhne (2005) resultou na matriz de ganho  $\kappa_2$  descrita na expressão (22), sendo  $v_r = 0,3m/s$  e  $w_r = 0,1 rad/s$  e  $\mathbf{Q}_2$  e  $\mathbf{R}_2$  descritas em 21. No ajuste fino da matriz  $\mathbf{Q}$ , optou-se por atribuir valor menor aos elementos  $Q_{3,3}$  e  $Q_{6,6}$  por estarem associados ao erro de orientação do robô, que é uma variável muito sensível em comparação com o erro em x e em y.

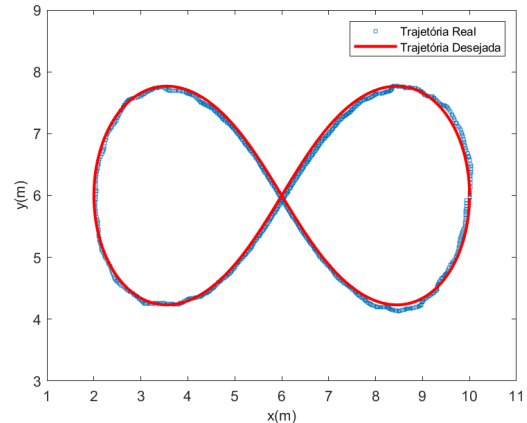


Figura 8. Traj. Lemniscata ( $LQR_2$ ).

Como consta nas Figuras 8 a 11, pode-se perceber que, ainda que o modelo tenha resultado de um conjunto de aproximações, o controlador  $LQR_2$  implementado foi capaz de conduzir o robô ao longo da trajetória de referência de forma suave, sem variações muito bruscas no sinal de controle. É importante frisar que só foi possível de fato seguir a referência, graças à ação integral incorporada, visto que, o modelo testado sem ela gera um erro crescente a cada instante de tempo, desestabilizando completamente o controle.

## 5. CONCLUSÃO

Este trabalho propôs um estudo e aplicação de controladores LQR aos robôs móveis da classe *skid-steer* com rodas visando o problema de rastreamento de trajetórias. Foram

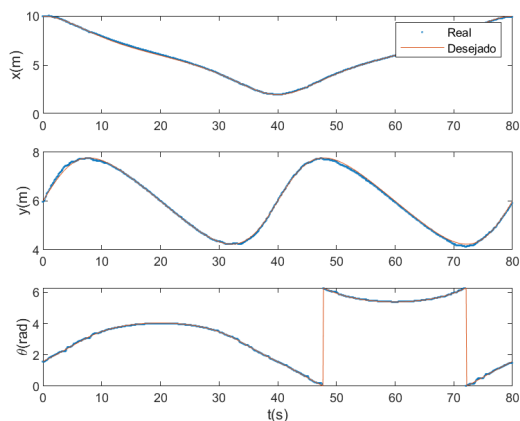


Figura 9. Pose do robô (LQR<sub>2</sub>).

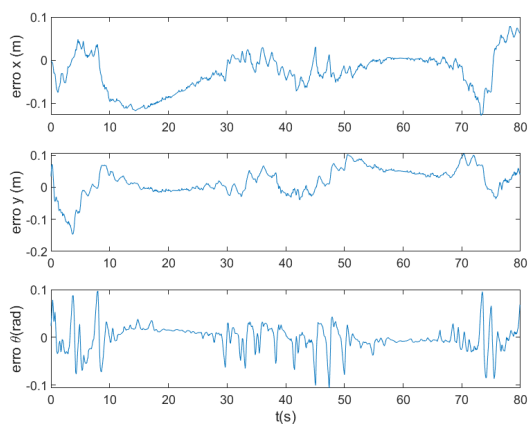


Figura 10. Sinal de erro (LQR<sub>2</sub>).

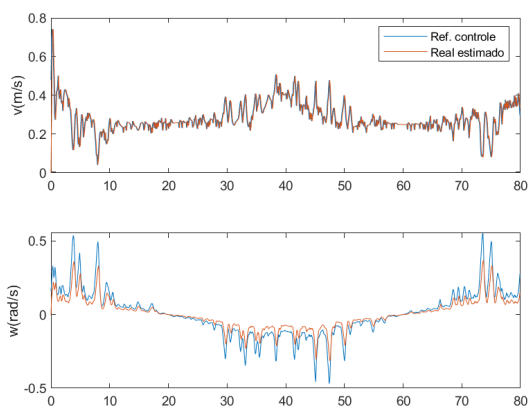


Figura 11. Sinal de controle (LQR<sub>2</sub>).

implementadas duas técnicas de controle utilizando o LQR a partir de dois modelos no espaço de estados. O primeiro modelo foi proposto por Klancar et al. (2005) e tem como princípio o erro de rastreamento de trajetória em comparação com um robô virtual de referência. Já o segundo modelo consistiu em uma versão aumentada do modelo linearizado proposto por Kuhne (2005), o qual sugeriu a implementação para um controle MPC. Portanto, ao implementar este modelo na versão aumentada, foi possível

avaliar o desempenho de um controlador que considera um horizonte infinito, a partir de um modelo idealizado para uma implementação de horizonte finito.

Com os resultados alcançados, foi possível atingir o objetivo do robô de fato conseguir seguir as trajetórias de referência e, além disso, avaliar os efeitos na ação do controlador das ponderações das matrizes  $\mathbf{Q}$  e  $\mathbf{R}$ , necessárias para a solução do LQR. Para o controlador LQR implementado a partir da versão aumentada do modelo linearizado, foi possível constatar a importância da ação integral no sistema, a qual foi fundamental para que o sinal de controle fosse suficiente para conduzir o robô para a trajetória de referência. Também pode-se constatar que, ainda que existam aproximações no modelo, foi possível atingir uma resposta satisfatória para o rastreamento da trajetória imposta ao sistema no ambiente simulado.

Em decorrência da pandemia de Covid-19 e a suspensão das atividades na Universidade Federal da Bahia, os testes experimentais em laboratório foram limitados. Portanto, em trabalhos futuros, pretende-se aplicar os controles implementados neste estudo ao robô real em laboratório, bem como projetar novas técnicas de controle visando o problema de rastreamento de trajetória, como o MPC.

## 6. AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Os autores agradecem também ao CNPq pelo apoio financeiro, termo de outorga 311029/2020-5.

## REFERÊNCIAS

- Boas, L.S.V. and Conceicao, A.G.S. (2020). Modelagem cinemática de robôs móveis da classe skid-steer. 2. doi: 10.48011/asba.v2i1.1165.
- Figueiredo, L.C. and Jota, F.G. (2004). Remote control of an omnidirectional mobile robot with time-varying delay and noise attenuation. *Sba Controle e Automação*, 15. doi:doi.org/10.1590/S0103-17592004000300002.
- Klancar, G., Matko, D., and Blazic, S. (2005). Mobile robot control on a reference path. In *Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005.*, 1343–1348. doi:10.1109/.2005.1467211.
- Klancar, G. and Skrjanc, I. (2007). Tracking-error model-based predictive control for mobile robots in real time. *Robotics and Autonomous Systems*, 55(6), 460–469. doi: https://doi.org/10.1016/j.robot.2007.01.002.
- Kuhne, F. (2005). Controle preditivo de robôs móveis não holonômicos.
- Ogata, K. (2010). *Engenharia de Controle Moderno*, 5ª edição. Editora Pearson.
- Santos, J. (2017). Applying a state feedback controller plus the filtered smith predictor to control an omnidirectional mobile robot under time-varying delay.
- Wang, L. (2009). *Model Predictive Control System Design and Implementation Using MATLAB*. Advances in Industrial Control. Springer London.