

CONTROLADOR ROBUSTO \mathcal{H}_∞ BASEADO EM REDES NEURAIS APLICADO EM ROBÔS MÓVEIS COM RODAS

ROBERTO S. INOUE*, TATIANA F.P.A.T. PAZELLI*, ADRIANO A.G. SIQUEIRA†, MARCO H. TERRA*

**Departamento de Engenharia Elétrica - Universidade de São Paulo em São Carlos C.P.359, São Carlos, SP, 13566-590, Brasil*

†*Departamento de Engenharia Mecânica - Universidade de São Paulo em São Carlos C.P.359, São Carlos, SP, 13566-590, Brasil*

Emails: rsinoue@sel.eesc.usp.br, tati@sel.eesc.usp.br, siqueira@sc.usp.br, terra@sel.eesc.usp.br

Abstract— A mixed model based/neural adaptive robust controller with \mathcal{H}_∞ performance is proposed for wheeled mobile robots. An adaptive control law based on neural network model is used only to adapt the uncertain dynamics of the wheeled mobile robot. This control law is developed for actuating as a complement of the nominal model. Experimental results obtained from an actual wheeled mobile robot are presented to show the robustness of this approach.

Keywords— Adaptive Control, Robust Control, Neural Network, Wheeled Mobile Robots.

Resumo— Um controlador robusto baseado em modelo e em redes neurais com critério de desempenho \mathcal{H}_∞ é proposto neste artigo para robôs móveis com rodas. Uma lei de controle adaptativa baseada em rede neurais é utilizada para adaptar somente as incertezas dinâmicas do robô móvel com rodas. Essa lei de controle atua como um complemento do modelo nominal. Resultados experimentais são apresentados para mostrar a robustez desta abordagem.

Palavras-chave— Controle Adaptativo, Controle Robusto, Redes Neurais, Robôs Móveis com Rodas.

1 Introdução

Controladores dinâmicos de robôs móveis com rodas (RMRs) baseados em modelo não têm recebido tanta atenção na literatura quanto controladores baseados apenas na cinemática. Um estudo comparativo entre duas estratégias de controle \mathcal{H}_∞ não linear foi realizado em (dos Reis et al., n.d.; Inoue et al., 2007) considerando o modelo matemático do RMR completamente conhecido. Essas estratégias baseadas em critérios \mathcal{H}_∞ foram originalmente desenvolvidas para robôs manipuladores (Chen et al., 1994). Para o caso em que o modelo matemático não é conhecido, (Chang, 2005) desenvolve controladores \mathcal{H}_∞ não lineares baseados em redes neurais e em lógica fuzzy para estimar o modelo completo do sistema. Nessa referência somente resultados simulados foram apresentados. Também, os robôs considerados são robôs manipuladores.

Quando o modelo nominal é conhecido e incertezas paramétricas estão presentes no modelo, uma abordagem interessante desenvolvida em (Ge et al., 1998) usa rede neural para estimar somente a parte incerta do sistema robótico. Vale ressaltar que a abordagem utilizada em (Ge et al., 1998) não é robusta.

Em adição a (Ge et al., 1998; dos Reis et al., n.d.; Chang, 2005; Inoue et al., 2007), este trabalho desenvolve um controle robusto adaptativo baseado em redes neurais para RMR. O modelo nominal é conhecido e a rede neural é utilizada apenas para estimar a parte incerta do sistema.

Este artigo complementa os resultados obtidos em (Inoue et al., n.d.) onde ao invés de uma rede neural foi utilizada lógica fuzzy nessa estratégia de controle.

Este artigo está organizado da seguinte forma: na Seção 2 os modelos do RMR são apresentados; na Seção 3 a rede neural e o controlador \mathcal{H}_∞ proposto são apresentados; na Seção 4 os resultados experimentais são apresentados e na Seção 5 as conclusões.

2 Modelos do Robô Móvel com Rodas

Nesta seção, os modelos cinemático e dinâmico de um RMR são apresentados e em seguida será caracterizado o termo incerto do sistema dinâmico a ser estimado pelas redes neurais. A geometria do robô é mostrada na Fig. 1, sendo (X, Y) o sistema de coordenadas inercial; (X_o, Y_o) o sistema de coordenadas local; a o comprimento do robô; d a distância entre P_o e P_c ; b a distância entre a roda atuada e o eixo de simetria; r o raio das rodas atuadas; θ_d e θ_e o deslocamento angular da roda direita e esquerda, respectivamente.

2.1 Modelo Cinemático

O RMR ilustrado na Fig. 1 apresenta três restrições cinemáticas (Coelho and Nunes, 2003). Assumindo que a velocidade de P_o deve estar na direção do eixo de simetria (eixo X_o), a primeira restrição com respeito a P_c é dada por

$$\dot{y}_c \cos \alpha - \dot{x}_c \sin \alpha - d\dot{\alpha} = 0,$$

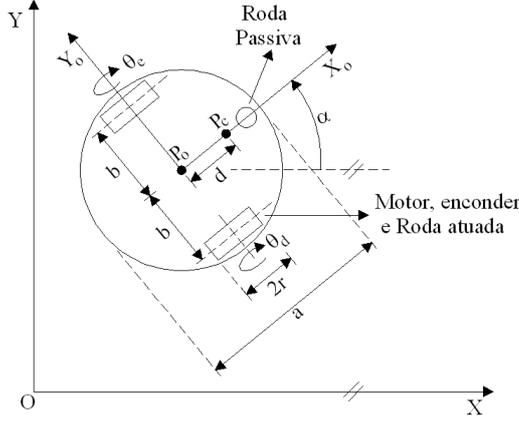


Figura 1: Geometria do RMR.

sendo (x_c, y_c) as coordenadas do centro de massa P_c no sistema de coordenadas inercial, e α é o ângulo entre o eixo de simetria do robô e o eixo X.

As outras duas restrições estão relacionadas com a rotação das rodas, ou seja, as rodas atuadas não podem girar em falso,

$$\begin{aligned}\dot{x}_c \cos \alpha + \dot{y}_c \sin \alpha + b\dot{\alpha} - r\dot{\theta}_d &= 0, \\ \dot{x}_c \cos \alpha + \dot{y}_c \sin \alpha - b\dot{\alpha} - r\dot{\theta}_e &= 0.\end{aligned}$$

Definindo $q = [q_1 \ q_2]$, $q_1 = [x_c \ y_c \ \alpha]^T$ e $q_2 = [\theta_d \ \theta_e]^T$, então as três restrições podem ser escritas na forma

$$A(q)\dot{q} = \begin{bmatrix} -\sin \alpha & \cos \alpha & -d & 0 & 0 \\ -\cos \alpha & -\sin \alpha & -b & r & 0 \\ -\cos \alpha & -\sin \alpha & b & 0 & r \end{bmatrix} \dot{q} = 0.$$

A matriz $A(q)$ tem posto pleno e pode ser expressa como $[A_1(q)_{(3 \times 3)} \ A_2(q)_{(3 \times 2)}]$, tal que $A_1(q)$ é não singular e $S(q) = [-A_1^{-1}(q)A_2 \ I_{(2 \times 2)}]^T$ é tal que $A(q)S(q) = 0$. A equação cinemática é dada por

$$\dot{q}(t) = S(q)\dot{q}_2(t). \quad (1)$$

ou

$$\begin{aligned}\dot{x}_c &= c(b \cos \alpha - d \sin \alpha)\dot{\theta}_d + c(b \cos \alpha + d \sin \alpha)\dot{\theta}_e, \\ \dot{y}_c &= c(b \sin \alpha + d \cos \alpha)\dot{\theta}_d + c(b \sin \alpha - d \cos \alpha)\dot{\theta}_e, \\ \dot{\alpha} &= c(\dot{\theta}_d - \dot{\theta}_e).\end{aligned} \quad (2)$$

sendo $c = r/(2b)$.

2.2 Controlador Baseado na Cinemática

Nesta subseção, é apresentado o controlador baseado na cinemática proposto por (Kanayama et al., 1990), que gera as velocidades desejadas para o problema de acompanhamento de trajetória de referência de RMR.

Considere o erro $q_e = [x_e \ y_e \ \alpha_e]^T$, entre a referência de postura $P_r = [x_r, y_r, \alpha_r]$, e a postura real do RMR $P_c = [x_c, y_c, \alpha]$, dada por

$$\begin{aligned}x_e &= \cos \alpha (x_r - x_c) + \sin \alpha (y_r - y_c), \\ y_e &= -\sin \alpha (x_r - x_c) + \cos \alpha (y_r - y_c), \\ \alpha_e &= \alpha_r - \alpha,\end{aligned} \quad (3)$$

sendo $(x_r, y_r) = q_r$ a trajetória de referência escolhida e $\alpha_r = \tan^{-1}(\dot{y}_r/\dot{x}_r)$. As velocidades desejadas linear (V^d) e angular (W^d) do RMR são dadas por

$$\begin{aligned}V^d &= V_r \cos \alpha_e + k_x x_e, \\ W^d &= W_r + V_r(k_y y_e + k_\alpha \sin \alpha_e),\end{aligned} \quad (4)$$

sendo k_x, k_y, k_α constantes,

$$V_r = \sqrt{(\dot{x}_r)^2 + (\dot{y}_r)^2} \text{ e } W_r = \dot{\alpha}_r. \quad (5)$$

Na Subseção 2.4, o controlador baseado na dinâmica é proposto considerando a velocidade angular desejada das rodas, \dot{q}_2^d . Então, é necessário definir a relação de velocidades que segue

$$\dot{q}_2^d = \begin{bmatrix} \dot{\theta}_d^d \\ \dot{\theta}_e^d \end{bmatrix} = \begin{bmatrix} 1/r & b/r \\ 1/r & -b/r \end{bmatrix} \begin{bmatrix} V^d \\ W^d \end{bmatrix}, \quad (6)$$

sendo $\dot{\theta}_d^d$ e $\dot{\theta}_e^d$ as velocidade angulares desejadas da roda direita e esquerda, respectivamente.

2.3 Modelo Dinâmico

A equação dinâmica de RMR usando teoria de Lagrange é descrita por (Coelho and Nunes, 2003) como

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = E\tau - A^T(q)\lambda, \quad (7)$$

sendo $\lambda = [\lambda_1 \ \lambda_2 \ \lambda_3]^T$ o vetor de restrição de forças; $E = [0_{(2 \times 3)} \ I_{(2 \times 2)}]^T$ a matriz de entrada; $\tau = [\tau_d \ \tau_e]^T$ o vetor de entrada (torque nas rodas);

$$C(q, \dot{q}) = \begin{bmatrix} 0 & 0 & m d \dot{\alpha} \cos \alpha & 0 & 0 \\ 0 & 0 & m d \dot{\alpha} \sin \alpha & 0 & 0 \\ & & 0_{(3 \times 5)} & & \end{bmatrix}$$

a matriz de forças de Coriolis e centrípeta; e

$$M(q) = \begin{bmatrix} m & 0 & m d \sin \alpha & 0 & 0 \\ 0 & m & -m d \cos \alpha & 0 & 0 \\ m d \sin \alpha & -m d \cos \alpha & I & 0 & 0 \\ 0 & 0 & 0 & I_r & 0 \\ 0 & 0 & 0 & 0 & I_r \end{bmatrix}$$

a matriz de inércia. Os parâmetros m e I são dados por

$$\begin{aligned}m &= m_p + 2m_r \text{ e} \\ I &= I_c + 2m_r(d^2 + b^2) + 2I_m + m_p d^2,\end{aligned}$$

sendo m_r a massa de cada roda atuada mais o rotor do motor; m_p a massa da plataforma do robô; I_c a inércia do robô em relação ao eixo vertical em P_c ; I_r o momento de inércia de cada roda com o rotor em relação ao eixo da roda; I_m o momento de inércia em relação ao eixo definido no plano da roda (perpendicular ao eixo da roda).

2.4 Formulação do Problema

O modelo dinâmico mostrado na Subseção 2.3 é descrito nesta subseção na forma de espaço de estado. Diferenciando (1) com respeito ao tempo, substituindo o resultado em (7), e então multiplicando o resultado por S^T à esquerda, obtém-se

$$M_2\ddot{q}_2 + C_2(\dot{q}_2)\dot{q}_2 = S^T E\tau = \tau. \quad (8)$$

Incertezas paramétricas podem ser introduzidas em (8) dividindo a matriz de parâmetros M_2 e $C_2(\dot{q}_2)$ em uma parte nominal e uma parte perturbada

$$\begin{aligned} M_2 &= M_0 + \Delta M_0 \\ C_2(\dot{q}_2) &= C_0(\dot{q}_2) + \Delta C_0(\dot{q}_2) \end{aligned}$$

sendo M_0 uma matriz constante simétrica, não singular, dada por $S^T(q)M(q)S(q)$ e $C_0(\dot{q}) = C_0(\dot{\alpha}) = C_0(\dot{q}_2) = S^T(q)C(q, \dot{q})S(q) + S^T(q)M(q)\dot{S}(q)$. Adicionando um distúrbio de torque $w = [w_d \ w_e]^T$ e substituindo (2) em (8), segue que

$$\ddot{q}_2 = A_2(\dot{q}_2)\dot{q}_2 + B\tau + Bw, \quad (9)$$

sendo $A_2(\dot{q}_2) = -M_2^{-1}C_2(\dot{q}_2)$ e $B = M_2^{-1}$. Define-se o estado como

$$\tilde{x} = \begin{bmatrix} \dot{q}_2 \\ q_2 \end{bmatrix} = [(\dot{\theta}_d - \dot{\theta}_d^d) \ (\dot{\theta}_e - \dot{\theta}_e^d) \ (\theta_d - \theta_d^d) \ (\theta_e - \theta_e^d)]^T.$$

Adicionando e subtraindo \ddot{q}_2^d e $A_2(\dot{q}_2)\dot{q}_2^d$ em (9) a forma de espaço de estado para acompanhamento de trajetória de RMR é dada por

$$\dot{\tilde{x}} = \begin{bmatrix} A_2(\dot{q}_2) & 0 \\ I & 0 \end{bmatrix} \tilde{x} + \begin{bmatrix} I \\ 0 \end{bmatrix} [-\ddot{q}_2^d + A_2\dot{q}_2^d + B\tau] + \begin{bmatrix} B \\ 0 \end{bmatrix} w$$

Considerando a transformação de estado

$$T_0\tilde{x} = \begin{bmatrix} T_{11} & T_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{q}_2 \\ \tilde{q}_2 \end{bmatrix}$$

sendo $T_{11}, T_{12} \in \mathfrak{R}^{n \times n}$ matrizes constantes a serem determinadas, a equação dinâmica do erro de estado de acompanhamento de trajetória torna-se

$$\dot{\tilde{x}} = A_T(\dot{q}_2)\tilde{x} + B_T T_{11}(-F_0(x_e) - \Delta F_0(x_e) + \tau) + B_T d, \quad (10)$$

com

$$A_T(\dot{q}_2) = T_0^{-1} \begin{bmatrix} -M_0^{-1}C_0(\dot{q}_2) & 0 \\ T_{11}^{-1} & -T_{11}^{-1}T_{12} \end{bmatrix} T_0,$$

$$B_T = T_0^{-1} \begin{bmatrix} M_0^{-1} \\ 0 \end{bmatrix}, \quad d = M_0 T_{11} M_0^{-1} w,$$

$$x_e = [q_2^T \ \dot{q}_2^T \ (q_2^d)^T \ (\dot{q}_2^d)^T \ (\ddot{q}_2^d)^T],$$

$$F_0(x_e) = M_0(\ddot{q}_2^d - T_{11}^{-1}T_{12}\dot{q}_2^d) + C_0(\dot{q}_2)(\dot{q}_2^d - T_{11}^{-1}T_{12}\tilde{q}_2), \quad e$$

$$\Delta F_0(x_e) = \Delta M_0(\ddot{q}_2^d - T_{11}^{-1}T_{12}\dot{q}_2^d) + \Delta C_0(\dot{q}_2)(\dot{q}_2^d - T_{11}^{-1}T_{12}\tilde{q}_2).$$

O termo incerto $\Delta F_0(x_e)$ é considerado totalmente desconhecido e é estimado por um conjunto de redes neurais.

3 Controle \mathcal{H}_∞ Baseado em Modelo e em Redes Neurais

3.1 Rede Neural

Nesta subseção, uma rede neural $\Delta\hat{F}_0(x_e, \Theta)$, sendo Θ um vetor contendo os parâmetros de ajuste da rede, é usada para aproximar somente o termo incerto $\Delta F_0(x_e)$ em (10). São definidas n redes neurais $\Delta\hat{F}_{0k}(x_e, \Theta_k)$, $k = 1, \dots, n$ para cada atuador do RMR (atuadores da roda direita e esquerda), compostas por neurônios não lineares em todas as camadas escondidas e neurônios lineares nas camadas de entrada e saída, com parâmetros ajustáveis Θ_k na camada de saída (Chang, 2000; Chang and Chen, 1997). As saídas das redes neurais são da forma

$$\begin{aligned} \Delta\hat{F}_{0k}(x_e, \Theta_k) &= \sum_{i=1}^{p_k} H \left(\sum_{j=1}^{5n} w_{ij}^k x_{ej} + b_i^k \right) \Theta_{ki} \\ &= \xi_k^T \Theta_k \end{aligned} \quad (11)$$

com

$$\xi_k = \begin{bmatrix} H \left(\sum_{j=1}^{5n} w_{1j}^k x_{ej} + b_1^k \right) \\ \vdots \\ H \left(\sum_{j=1}^{5n} w_{p_k j}^k x_{ej} + b_{p_k}^k \right) \end{bmatrix}, \quad \Theta_k = \begin{bmatrix} \Theta_{k1} \\ \vdots \\ \Theta_{kp_k} \end{bmatrix}$$

sendo p_k o número de neurônios da camada escondida. Os pesos w_{ij}^k e o bias b_i^k para $1 \leq i \leq p_k$, $1 \leq j \leq 5n$ e $1 \leq k \leq n$ são assumidos constantes e especificados pelo projetista e $H(\cdot)$ é escolhida como sendo uma função tangente hiperbólica

$$H(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}.$$

A rede neural completa pode ser denotada por

$$\begin{aligned} \Delta\hat{F}_0(x_e, \Theta) &= \begin{bmatrix} \Delta\hat{F}_{01}(x_e, \Theta_1) \\ \vdots \\ \Delta\hat{F}_{0n}(x_e, \Theta_n) \end{bmatrix} = \begin{bmatrix} \xi_1 \Theta_1 \\ \vdots \\ \xi_n \Theta_n \end{bmatrix} \\ &= \begin{bmatrix} \xi_1 & 0 & \dots & 0 \\ 0 & \xi_2 & \vdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \xi_n \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \vdots \\ \Theta_n \end{bmatrix} = \Xi \Theta. \end{aligned} \quad (12)$$

Considere uma região de restrição Ω_Θ de parâmetros Θ . Para garantir que o parâmetro estimado $\hat{\Theta}(t)$ para todo t esteja dentro da região de restrição, o algoritmo de projeção pode ser usado. Considere $\Omega_0 = \{\Theta : \Theta^T \Theta \leq N\}$ e $\Omega_\Theta = \Theta : \{\Theta^T \Theta \leq N + \Delta\}$ para algum $N > 0$ e $\Delta > 0$. Um algoritmo de projeção pode ser dado

como em (Chang, 2000)

$$Proj[\Phi] = \begin{cases} \Phi, & \text{se } \Theta^T \Theta \leq N \text{ ou} \\ (\Theta^T \Theta > N \text{ e } \Theta^T \Phi \leq 0) \\ \Phi - \frac{(\Theta^T \Theta - N)\Theta^T \Phi}{\delta \Theta^T \Theta} \Theta, & \text{caso contrário} \end{cases} \quad (13)$$

para alguma função suave $\Phi(x_e)$.

É assumido que existe um valor $\Theta^* \in \Omega_0$ tal que $\Delta \hat{F}_0(x_e, \Theta^*)$ pode aproximar $\Delta F_0(x_e)$ o mais próximo possível. Um modo de alcançar o desempenho \mathcal{H}_∞ , quando a rede neural é usada, é considerar o erro de aproximação, $\Delta F(x_e) = \Delta \hat{F}_0(x_e, \Theta^*) - \Delta F_0(x_e)$, como distúrbio. Esta abordagem foi usada em (Chang and Chen, 1997), onde o termo dinâmico do robô $F(x_e)$ é aproximado por uma rede neural. Porém, desde que o erro de aproximação inclua os efeitos de incertezas dinâmicas, a propriedade do distúrbio para ser quadraticamente integrável não é simples de ser demonstrada. Uma alternativa para esta limitação é não considerar o erro de aproximação como distúrbio e adicionar um controlador por estrutura variável (CEV), proposto em (Chang, 2000), no algoritmo proposto deste artigo.

A exigência agora, gerada pelo uso de CEV, é que o erro de aproximação deve ser limitado por uma função dependente do estado, isto é, existe uma função $k(x_e) > 0$ tal que $|(\Delta F(x_e))_i| \leq k(x_e)$, para todo $1 \leq i \leq n$.

3.2 Controle \mathcal{H}_∞ Não Linear

Com essas suposições, o problema de controle \mathcal{H}_∞ não linear adaptativo neural para sistemas robóticos pode ser formulado como se segue: dado um nível de atenuação γ , encontre um controlador adaptativo neural com realimentação de estado

$$\tau = F_0(x_e) + \Xi \Theta + T_{11}^{-1} u + u_s \quad (14)$$

com

$$\dot{\Theta} = Proj[-Z^{-1} \Xi^T T_{11} B^T T_0 \tilde{x}] \quad (15)$$

$$u = -R^{-1} B^T T_0 \tilde{x}, \quad (16)$$

$$u_s = -k(x_e) \text{sgn}(T_{11} B^T T_0 \tilde{x}) \quad (17)$$

sendo $B = [I \mid 0]^T$, tal que o seguinte índice de desempenho seja satisfeito

$$\int_0^T (\tilde{x}^T Q \tilde{x} + u^T R u) dt \leq \tilde{x}^T(0) P \tilde{x}(0) + \tilde{\Theta}^T(0) Z \tilde{\Theta}(0) + \gamma^2 \int_0^T (d^T d) dt, \quad (18)$$

para matrizes $Q = Q^T > 0, R = R^T > 0, P = P^T > 0$, e $Z = Z^T > 0$, sendo $\tilde{\Theta} = \Theta - \Theta^*$ denotando o erro de estimativa dos parâmetros neurais. O termo u_s em (14) é o CEV, usado para

eliminar o efeito do erro de aproximação. Para a função de Lyapunov

$$V = \frac{1}{2} \tilde{x}^T P \tilde{x} + \frac{1}{2} \tilde{\Theta}^T Z \tilde{\Theta},$$

sendo P , simétrica definida positiva, solução da equação de Riccati

$$\dot{P} + P A_T + A_T^T P + P B_T \left(R^{-1} - \frac{1}{\gamma^2} I \right) B_T^T P + Q = 0. \quad (19)$$

A equação (19) pode ser simplificada para uma equação algébrica desde que $(C_0(\dot{q}_2) - \frac{1}{2} \dot{M}_0)$ seja uma matriz anti-simétrica e a matriz P seja selecionada como

$$P = T_0^T \begin{bmatrix} M_0 & 0 \\ 0 & K \end{bmatrix} T_0.$$

A equação algébrica simplificada é da forma

$$\begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} - T_0^T B \left(R^{-1} - \frac{1}{\gamma^2} I \right) B^T T_0 + Q = 0, \quad (20)$$

sendo

$$T_0 = \begin{bmatrix} R_1^T Q_1 & R_1^T Q_2 \\ 0 & I \end{bmatrix}$$

e

$$K = \frac{1}{2} (Q_1^T Q_2 - Q_2^T Q_1) - \frac{1}{2} (Q_{21}^T + Q_{12}),$$

com condições: $K > 0$ e $R < \gamma^2 I$. A matriz R_1 é o resultado da fatoração de Cholesky

$$R_1^T R_1 = \left(R^{-1} - \frac{1}{\gamma^2} I \right)^{-1},$$

e a matriz Q simétrica definida positiva é fatorada como

$$Q = \begin{bmatrix} Q_1^T Q_1 & Q_{12} \\ Q_{12}^T & Q_2^T Q_2 \end{bmatrix}.$$

4 Resultados Experimentais

Os resultados experimentais foram obtidos usando o RMR mostrado na Fig. 2. Este robô móvel consiste de uma placa controladora e motores DC com torque 0.29 N.m. A placa controladora é composta de um microprocessador MSP430F169 rodando a 8 MHz e um módulo RF (Rádio Frequência). Os parâmetros nominais do RMR são $a = 0.17(m)$, $b = 0.065(m)$, $d = 0(m)$, $r = 0.028(m)$, $m_r = 0.075(Kg)$, $m_p = 0.597(Kg)$ e inércias $I_c = 0.0022938(kg.m^2)$, $I_w = 0.000375(kg.m^2)$ e $I_m = 3.6788 \times 10^{-7}(kg.m^2)$.

Os dados de posição são obtidos com um sistema de visão usando Transformada de Hough Circular para estimar a informação de postura. O sistema de visão consiste de uma câmera COHU H10x8M-II com resolução de 320 x 240 pixels e



Figura 2: Robô Móvel com Rodas.

uma placa de aquisição PXC200F com taxa de processamento de 30 quadros/seg, o que define o tempo de amostragem do controlador, selecionado como 33 ms. O computador do usuário é um Pentium 4 com 3.2GHz. Os comandos de controle são gerados do computador do usuário e enviados através do módulo de RF rodando em 38400 bps. O algoritmo de controle foi implementado em C++. A Fig. 3 mostra o diagrama de blocos completo da estratégia de controle do robô móvel com rodas.

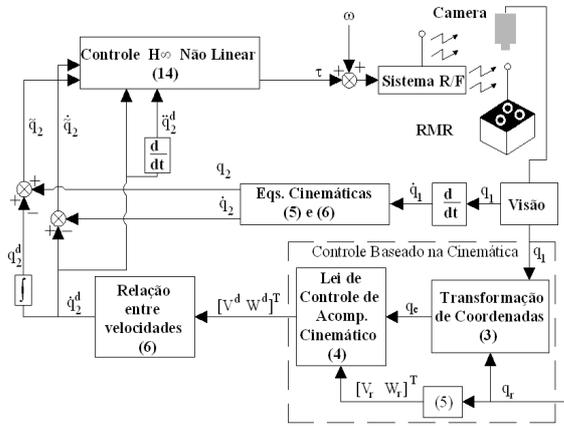


Figura 3: Sistema de controle de acompanhamento de trajetória para RMR.

A trajetória de referência é definida por $x_r = 0.0339t$ e $y_r = 0.1132\text{sen}(0.3t)$, com condições iniciais $(x_{c_o}, y_{c_o}, \alpha_{c_o}) = (0, 0, 0)$. Para o controlador baseado na cinemática, os ganhos foram definidos como $k_x = 0.75$, $k_y = 200$ e $k_\alpha = 25.5$. Para realizar uma análise de robustez dos controladores, distúrbios externos foram introduzidos nos torques das rodas da seguinte forma

$$w_d = 0.05e^{-(t-6)^4}\text{sen}(1.3\pi t),$$

$$w_e = -0.15e^{-(t-6)^4}\text{sen}(1.3\pi t).$$

O controlador \mathcal{H}_∞ é caracterizado por um nível de atenuação $\gamma = 66.632$ e $k(x_e) = \frac{1}{200}\sqrt{\tilde{x}^T \tilde{x}}$. As matrizes de ponderação selecionadas foram $Q_1 = I_2$, $Q_2 = 0.5 \times I_2$, $Q_{12} = 0$, $R = 10.8 \times I_2$ e $Z = 3000 \times I_{14}$. A lei de controle adaptativa para ajustar y_i é implementada baseada na equação (15). Definem-se $\Delta F(x_e, \Theta) = [\Delta F_1(x_e, \Theta_1) \Delta F_2(x_e, \Theta_2)]^T$ com $p_k = 7$ neurônios na camada escondida, $b_k = [-3 \ -2 \ -1 \ 0 \ 1 \ 2 \ 3]$ e a matriz de pesos para a primeira camada $\Omega_i^k = [\omega_{ij}^k] = [-1 \ -1 \ -1 \ -1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$. Os parâmetros Θ são definidos como

$$\Theta = \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix}$$

com

$$\Theta_1 = [\theta_{11} \ \theta_{12} \ \theta_{13} \ \theta_{14} \ \theta_{15} \ \theta_{16} \ \theta_{17}]^T,$$

$$\Theta_2 = [\theta_{21} \ \theta_{22} \ \theta_{23} \ \theta_{24} \ \theta_{25} \ \theta_{26} \ \theta_{27}]^T.$$

e a matriz Ξ pode ser calculada como

$$\Xi = \begin{bmatrix} \xi_1^T & 0 \\ 0 & \xi_2^T \end{bmatrix},$$

com $\xi_1^T = [\xi_{11}, \dots, \xi_{17}]$ e $\xi_2^T = [\xi_{21}, \dots, \xi_{27}]$. Os resultados experimentais, considerando distúrbios externos, são mostrados nas Figuras 4, 5, e 6.

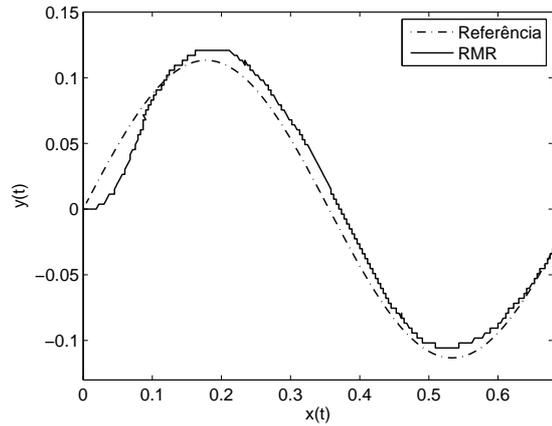


Figura 4: Acompanhamento de trajetória (sistema com distúrbios externos).

5 Conclusões

Neste artigo, um controlador robusto baseado em redes neurais e em modelo matemático é desenvolvido para um robô móvel com rodas. Uma rede

neural adaptativa aproxima as incertezas do modelo do robô, atuando como um complemento do modelo nominal. Resultados experimentais obtidos de um RMR são apresentados. Por ser uma estratégia de controle complementar à desenvolvida em (Inoue et al., n.d.), este artigo não apresenta um estudo comparativo com outras técnicas de controle. Os argumentos utilizados em (Inoue et al., n.d.) para justificar a utilização deste tipo de estratégia de controle são válidos aqui.

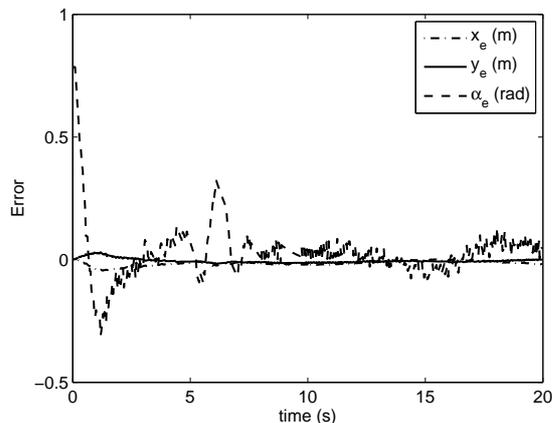


Figura 5: Erro de postura (sistema com distúrbios externos).

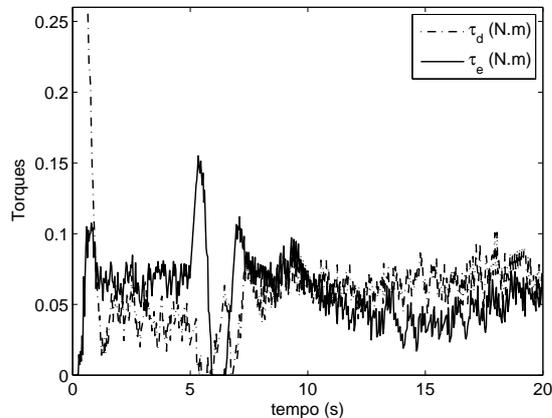


Figura 6: Torques τ_d e τ_e aplicados para a roda direita e esquerda, respectivamente (sistema com distúrbios externos).

Referências

- Chang, Y. C. (2000). Neural network-based \mathcal{H}_∞ tracking control for robotic systems, *IEE Proceedings of Control Theory Applications* **147**(3): 303–311.
- Chang, Y. C. (2005). Intelligent robust control for uncertain nonlinear time-varying systems and its application to robotic systems, *IEEE*

Transactions on Systems, Man and Cybernetics - Part B: Cybernetics **35**(6): 1108–1119.

Chang, Y. C. and Chen, B. S. (1997). A nonlinear adaptive \mathcal{H}_∞ tracking control design in robotic systems via neural networks, **5**(1): 13–29.

Chen, B. S., Lee, T. S. and Feng, J. H. (1994). A nonlinear \mathcal{H}_∞ control design in robotic systems under parameter perturbation and external disturbance, *International Journal of Control* **59**(2): 439–461.

Coelho, P. and Nunes, U. (2003). Lie algebra application to mobile robot control: A tutorial, *Robotica* **21**(5): 483 – 493.

dos Reis, G. A., Siqueira, A. A. G. and Terra, M. H. (n.d.). Nonlinear \mathcal{H}_∞ control via quasi-LPV representation and game theory for wheeled mobile robots, *Proceedings of the Mediterranean Conference on Control and Automation*, Limassol, Chipre.

Ge, S. S., Lee, T. H. and Harris, C. J. (1998). *Adaptive Neural Network Control of Robotic Manipulators*, World Scientific, Singapore.

Inoue, R. S., de F. P. A. T. Pazelli, T., Siqueira, A. A. G. and Terra, M. H. (n.d.). Mixed model based/fuzzy adaptive robust controller with \mathcal{H}_∞ criterion applied to wheeled mobile robots, *To appear in the proceedings of the 3rd IFAC Symposium on System, Structure and Control*, Iguassu Falls, Brazil.

Inoue, R. S., Siqueira, A. A. G. and Terra, M. H. (2007). Experimental results on the nonlinear \mathcal{H}_∞ control via quasi-LPV representation and game theory for wheeled mobile robots, *To appear in the proceedings of the IEEE Conference on Control Applications*, Singapore.

Kanayama, Y., Kimura, Y., Miyazaki, F. and Noguchi, T. (1990). A stable tracking control method for an autonomous mobile robot, *Proc. IEEE International Conference on Robotics and Automation*, pp. 384 – 389.