



CONTROLE NEURAL DE DOIS ELOS DE UM ROBÔ DE TRÊS GRAUS DE LIBERDADE

José Antônio Riul

Paulo Henrique de Miranda Montenegro

riul@ct.ufpb.br

paulo@ct.ufpb.br

Universidade Federal da Paraíba

Universidade Federal da Paraíba, Centro de Tecnologia, Departamento de Engenharia Mecânica, 58900-000, Paraíba, Brasil

Gustavo de Sá Ferreira

gustsaf@gmail.com

Universidade Federal da Paraíba

Universidade Federal da Paraíba, Centro de Tecnologia, Programa de Pós Graduação em Engenharia Mecânica, 58900-000, Paraíba, Brasil

Resumo. O objetivo do presente trabalho é projetar um controlador neural para dois elos de um robô manipulador de três graus de liberdade (3 GDL). O robô manipulador é composto por duas juntas rotacionais e uma prismática. Um moto-redutor com inversor de frequência e duas válvulas eletropneumáticas são utilizados para o acionamento do robô. Dois potenciômetros e uma régua potenciométrica são usados para captar as respostas dos elos do robô. Neste trabalho modelos dos dois elos do robô obtidos através do algoritmo dos mínimos quadrados recursivo (MQR) são utilizados em simulações, na obtenção das respostas dos dois elos controlados por uma rede neural artificial. No modelamento dos elos é considerado o acoplamento dinâmico entre eles. Finalizando serão apresentados resultados obtidos através simulações.

Palavras-chave: Robótica, controle, rede neural

1. INTRODUCAO

O presente trabalho tem como objetivo projetar um controlador neural para dois elos de um robô manipulador de três graus de liberdade (3 GDL). O robô manipulador é composto por duas juntas rotacionais e uma prismática. Um moto-redutor com inversor de frequência e duas válvulas eletropneumáticas são utilizados para o acionamento do robô. Dois potenciômetros e uma régua potenciométrica são usados para captar as respostas dos elos do robô. Neste trabalho modelos dos dois elos do robô obtidos considerando o acoplamento dinâmico entre eles, e usando o algoritmo dos mínimos quadrados recursivo (MQR) são utilizados em simulações, na obtenção das respostas dos dois elos controlados por uma rede neural artificial denominada de controlador neural. O controlador neural é projetado em duas etapas, na primeira denominada etapa de treinamento, os pesos da rede são ajustados para obtenção de uma resposta adequada para os elos; e na segunda denominada de etapa de controle, os pesos obtidos no treinamento são usados no início da ação de controle e são ajustados durante a ação de controle. Finalizando serão apresentados resultados de simulações, obtidos para os dois elos do robô, nas etapas de treinamento e de controle.

2. DESCRIÇÃO DO SISTEMA

O Robô Manipulador de 3 GDL mostrado na Fig. 1 é composto de duas juntas rotacionais e uma prismática. A junta 1 rotacional é acionada por um moto-redutor alimentado por um inversor de frequência e movimenta o elo 1 do robô. Este elo é uma coluna cilíndrica e tem um deslocamento angular máximo de 160° medido através de um potenciômetro. O movimento da junta 1 é transmitido para o elo 1 através de duas polias e uma correia dentada. A junta 2 rotacional é acionada por um sistema eletropneumático composto por uma válvula eletropneumática e um cilindro pneumático e movimenta o elo 2 do robô. Este elo é um perfil U e tem um deslocamento angular máximo de 45° medido através de um potenciômetro. O movimento da junta 2 é transmitido para o elo 2 através do deslocamento do pistão do cilindro pneumático. A junta 3 prismática é acionada por uma válvula eletropneumática e movimenta o elo 3 do robô que é a haste de um pistão de 500 mm de curso, de um cilindro pneumático fixado no interior do perfil U (elo 2). O deslocamento linear da haste do pistão é medido através de uma régua potenciométrica.

Um computador PC é utilizado para enviar comando de acionamento ao moto-redutor através do inversor de frequência e para as duas válvulas eletropneumáticas e para receber os sinais dos sensores potenciométricos. A comunicação do robô com o computador é realizada através de duas placas de entrada e saída de dados NI USB-6009, utilizando um programa computacional nas plataformas LabView e Matlab. Considerando as características de tensão e capacidade máxima de corrente das placas de entrada e saída de dados foi inserido um amplificador de potência para servir como fonte de suprimento aos elementos de acionamento do robô manipulador.



Figura 1. Robô manipulador de 3 GDL

3. CONTROLADOR NEURAL

Redes neurais (Haykin, S., 1994) podem ser usadas como controladores adaptativos (Åstrom et al., 1995) (Nascimento Júnior et al., 2000) de sistemas e de processos. Existem vários modelos de sistemas de controle usando redes neurais, como o controlador neural inverso direto (Miller III et al. (1990), o controlador neural adaptativo direto (Cavalcanti, 1994), o controlador híbrido (Miller III et al. (1990). No controlador neural adaptativo direto, mostrado na Fig. 2, o controlador é treinado para uma determinada referência, e não necessita-se de informações paramétricas do sistema. O alvo da rede depende do erro entre a referência e a saída do sistema e do jacobiano do sistema, que é determinado de forma direta ou indireta.

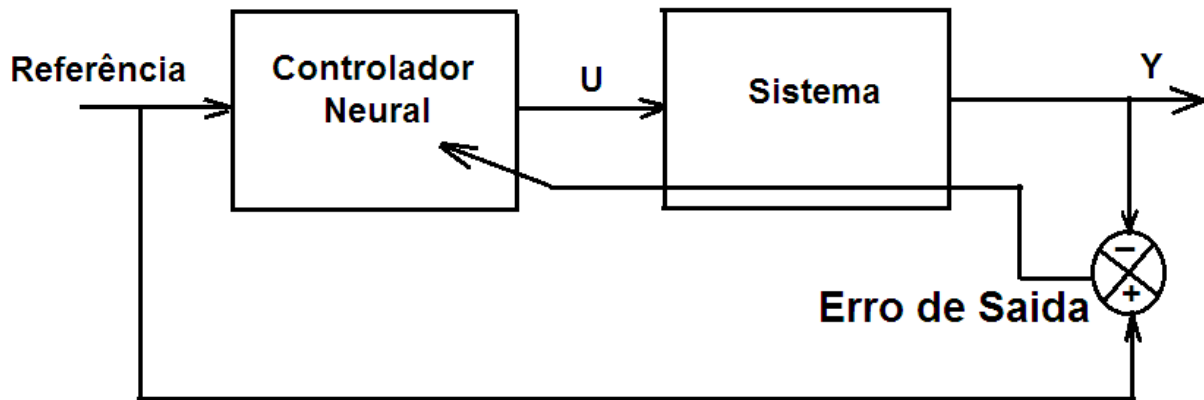


Figura 2. Diagrama de blocos do controlador neural e de um sistema

A rede neural usada para o controlador adaptativo direto é composta por uma camada de entrada (E) com neurônios lineares (L), por uma camada intermediária (I) com neurônios não lineares (NL), que usa a função de ativação tangente hiperbólica e por uma camada de saída (S) com neurônios lineares (L), conforme mostrado na Fig. 3. A entrada da rede é o vetor composto pelas referências, pelas variáveis de controle e pelos erros de saída, num total de 6 (seis) neurônios. A camada intermediária contém 4 (quatro) neurônios e a de saída 2 (dois), que são as variáveis de controle dos elos a serem controlados.

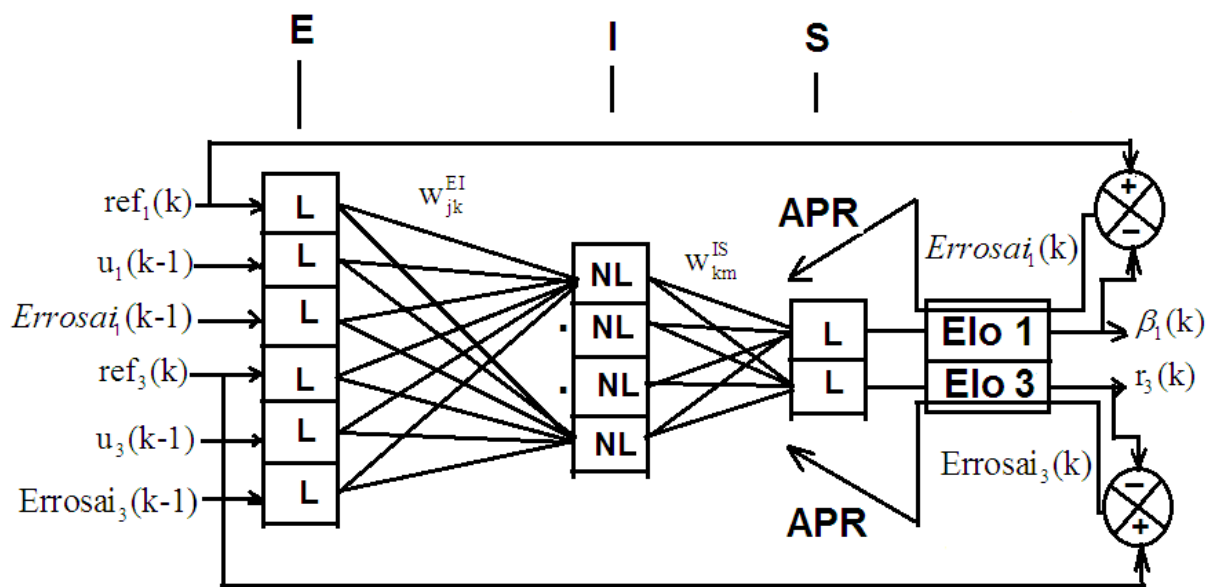


Figura 3. Controlador Neural e Elos 1 e 3 – Fase de Treinamento

As equações (1) à (6) são utilizadas para o cálculo da saída de cada neurônio, de cada camada da rede.

$$y_j^E = x_j^E \quad (1)$$

$$x_k^I = \sum_j w_{jk}^{EI} y_j^E = \sum_j (w_{jk}^{EI} x_j^E) + T_k^I \quad (2)$$

Substituindo (1) em (2), tem-se:

$$x_k^I = \sum_j w_{jk}^{EI} y_j^E = \sum_j (w_{jk}^{EI} x_j^E) + T_k^I \quad (3)$$

$$y_k^I = f(x_k^I) \quad (4)$$

$$x_m^S = \sum_k w_{km}^{IS} y_k^I \quad (5)$$

$$y_m^S = x_m^S = \sum_k w_{km}^{IS} y_k^I \quad (6)$$

onde:

j – j-ésimo neurônio da camada de entrada E;

k – k-ésimo neurônio da camada intermediária I;

m – m-ésimo neurônio da camada de saída S;

x_j^E - j-ésima entrada da camada E;

y_j^E - j-ésima saída da camada E;

x_k^I - k-ésima entrada da camada I;

y_k^I - k-ésima saída da camada I;

x_m^S - m-ésima entrada da camada S;

y_m^S - m-ésima saída da camada S;

w_{jk}^{EI} - elemento da matriz de pesos da camada I;

w_{km}^{IS} - elemento da matriz de pesos da camada de saída;

$f(.)$ – função de ativação;

T_k^I - bias do k-ésimo neurônio da camada intermediária.

O objetivo do modelo de controle é otimizar o índice de desempenho dado pela Eq. (7), representado na forma discreta, que é função das referências e das saída dos elos.

$$I_{U,w}(k) = \frac{1}{2} (Re f(k) - Y(k))^2 \quad (7)$$

A otimização é realizada como segue: primeiro, o alvo do controlador neural é determinado usando-se o índice de desempenho I_U e o algoritmo da propagação retroativa do erro (APR) (Kovács, 1996) conforme equações (8), (9) e (10).

$$I_U(k) = \frac{1}{2} (Re f(k) - Y(k))^2 \quad (8)$$

$$U(k) = U(k-1) - \eta \nabla I_U(k-1) \quad (9)$$

E,

$$U(k) = U(k-1) + \eta Errosai(k-1) \frac{\partial Y(k-1)}{\partial U(k-1)} \quad (10)$$

onde:

$ref_1(k)$ e $ref_3(k)$ - são as referências dos elos 1 e 3 do robô;

$u_1(k)$ e $u_3(k)$ - são as variáveis de controle dos elos 1 e 3 do robô;

$\beta_1(k)$ e $r_3(k)$ - são saídas dos elos 1 e 3 do robô;

$Errosai_1(k)$ e $Errosai_3(k)$ - são os erros de saída dos elos 1 e 3 do robô;

$Errosai(k) = Ref(k) - Y(k)$;

$$\text{Re } f(k) = \begin{bmatrix} \text{Re } f_1(k) \\ \text{Re } f_3(k) \end{bmatrix}, \quad Y(k) = \begin{bmatrix} \beta_1(k) \\ r_3(k) \end{bmatrix}, \quad \text{Errosai}(k) = \begin{bmatrix} \text{Errosai}_1(k) \\ \text{Errosai}_3(k) \end{bmatrix}, \quad U(k) = \begin{bmatrix} u_1(k) \\ u_3(k) \end{bmatrix}$$

$\frac{\partial Y(k)}{\partial U(k)}$ - jacobiano dos elos do robô;

$U(k-1)$ – saídas do controlador neural;

$U(k)$ – alvos do controlador neural;

η - fator de correção

Com a determinação do alvo do controlador, Eq. (10), usa-se o índice de desempenho dado pela Eq. (11), que é função do alvo obtido para o controlador e da saída do controlador, e através do algoritmo de propagação retroativa do erro, os parâmetros da rede são ajustados conforme as equações (11), (12) e (13).

$$I_w(k) = \frac{1}{2}(U(k) - U(k-1))^2 \quad (11)$$

$$w_{km}^{IS}(k) = w_{km}^{IS}(k-1) - \mu(\nabla I_w(w_{km}^{IS}))(k-1) \quad (12)$$

$$w_{jk}^{EI}(k) = w_{jk}^{EI}(k-1) - \mu(\nabla I_w(w_{jk}^{EI}))(k-1) \quad (13)$$

Das Equações (12) e (13), obtém-se a Eq. (14) e a Eq. (15), respectivamente.

$$w_{km}^{IS}(k+1) = w_{km}^{IS}(k) + \mu(E_m^S y_k^I)(k) \quad (14)$$

$$w_{jk}^{EI}(k+1) = w_{jk}^{EI}(k) + \mu\left(E_m^S w_{km}^{IS} \frac{\beta_k^I}{2} \left[1 - (y_k^I)^2\right] x_j^E\right)(k) \quad (15)$$

onde:

μ - fator de treinamento do controlador;

$k = t = nT_s$ – tempo discreto.

O jacobiano dos elos do robô foi calculado indiretamente, usando-se modelos lineares dos elos obtidos através do método de identificação dos mínimos quadrados recursivo (MQR) (Aguirre, 2000), (Coelho et al, 2004), dados pela Eq. (14) e Eq. (15). Foi dada uma sequência de entrada impulsiva $u_i(k)$ em cada elo e determinada a saída $y_i(k)$ de cada modelo, e através dos resultados obtidos, as funções $y_i(u_i)$ foram determinadas e conseqüentemente suas derivadas que são os jacobianos dos elos 1 e 3.

$$\beta_1(k) = 1,047\beta_1(k-1) - 0,111\beta_1(k-2) + 0,078u_1(k) - 0,005u_1(k-1) \quad (14)$$

$$r_3(k) = 1,760r_3(k-1) - 0,798r_3(k-2) + 0,004u_3(k) + 0,029u_3(k-1) \quad (15)$$

O aprendizado da variável de controle $U(k)$ para o controlador neural direto é concluído quando o índice de desempenho $I_{U,W}$ atinge um valor mínimo; ou seja, quando $Y(k) \cong \text{Ref}(k)$. Concluída a fase de treinamento os pesos w_{jk}^{EI} e w_{km}^{IS} são armazenados e são utilizados na segunda etapa para simulação dos elos 1 e 3 do robô sob controle.

4. RESULTADOS OBTIDOS PARA OS ELOS DO ROBÔ

Os resultados apresentados envolvem as etapas de treinamento do controlador neural e de controle. Nas Figuras 4 e 5 são mostradas as referências e as respostas obtidas dos elos do robô, na etapa de treinamento do controlador neural. As curvas são mostradas em função do tempo discreto $t = nT_s$, sendo T_s o tempo de amostragem e n as amostras. O tempo de amostragem utilizado nas simulações foi de 100 milissegundos.

Na fase de treinamento os pesos da última época (iteração) foram gravados e usados na fase de simulação. A Figura 4 mostra a referência e a saída do elo 1 do robô manipulador e a Fig. 5 mostra a referência e a saída do elo 3 do robô manipulador. Foram usadas 1100 amostras de cada elo do robô, a rede neural de 3 camadas mostrada na Fig. 3, treinada com: fator de treinamento $\mu = 0,4$ e 1 época (iteração). Os pesos treinados pela rede foram usados na etapa de controle, em simulações realizadas considerando outras referências, e os resultados são mostradas nas Figuras 6 a 11. A Figura 6 mostra a referência e a saída do elo 1 do robô manipulador, a Fig. 7 mostra o erro entre a referência e a saída do elo 1 e a Fig. 8 mostra a variável de controle obtida para o elo 1. A Figura 9 mostra a referência e a saída do elo 3 do robô manipulador, a Fig. 10 mostra o erro entre a referência e a saída do elo 3 e a Fig. 11 mostra a variável de controle obtida para o elo 3.

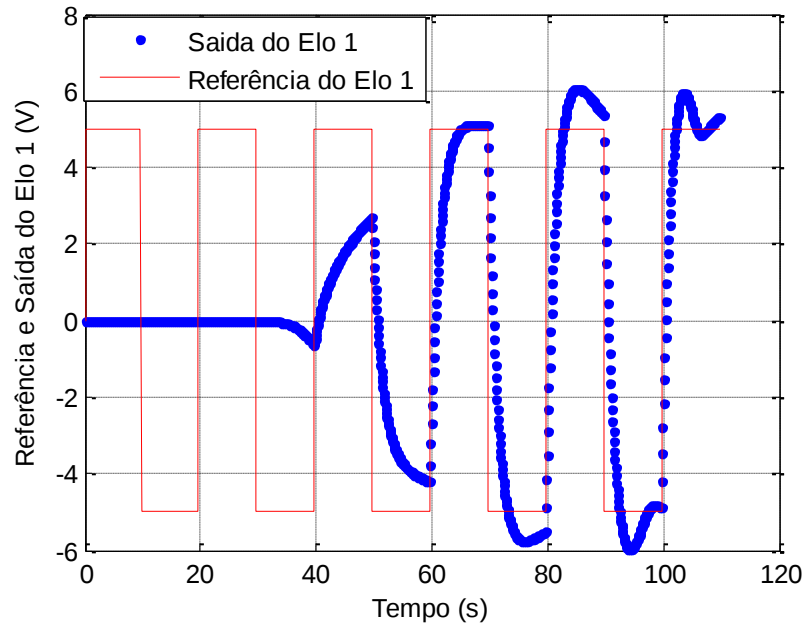


Figura 4. Referência e saída do elo 1 do robô (etapa de treinamento)

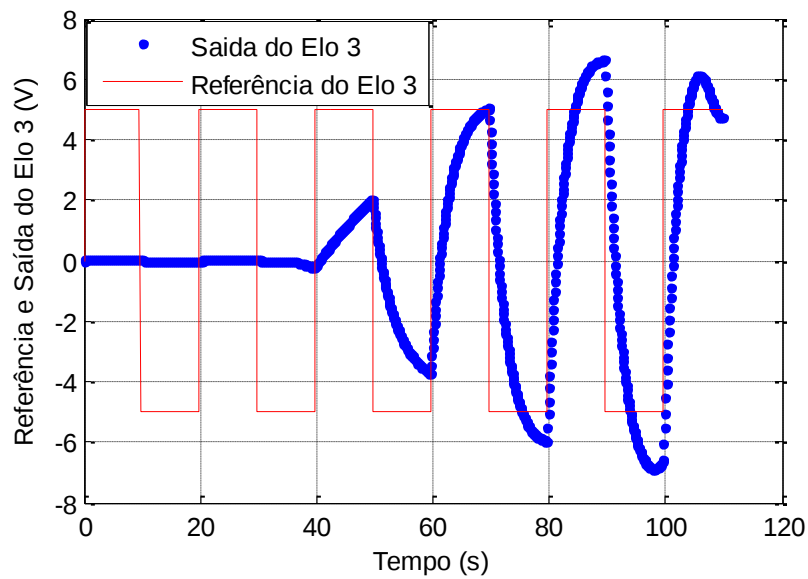


Figura 5. Referência e saída do elo 3 do robô (etapa de treinamento)

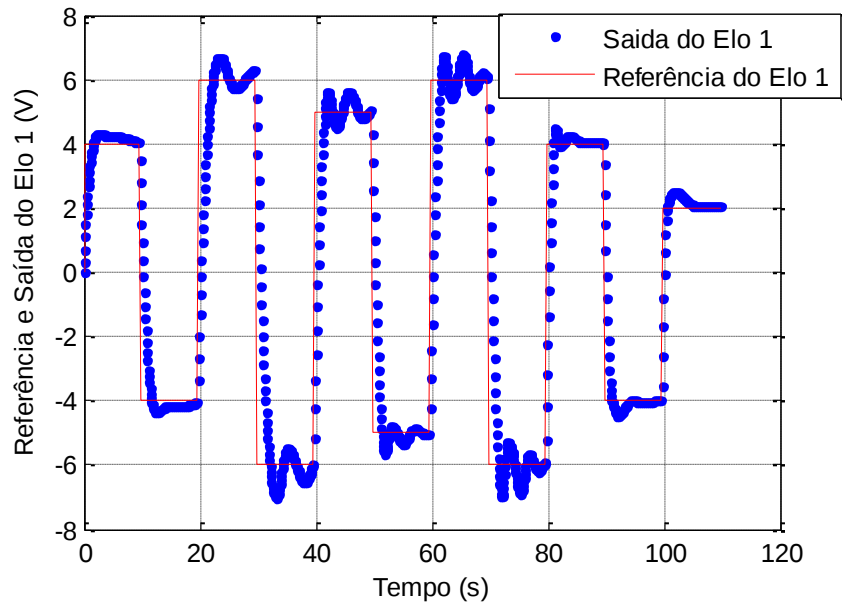


Figura 6. Referência e saída do elo 1 do robô (etapa de controle)

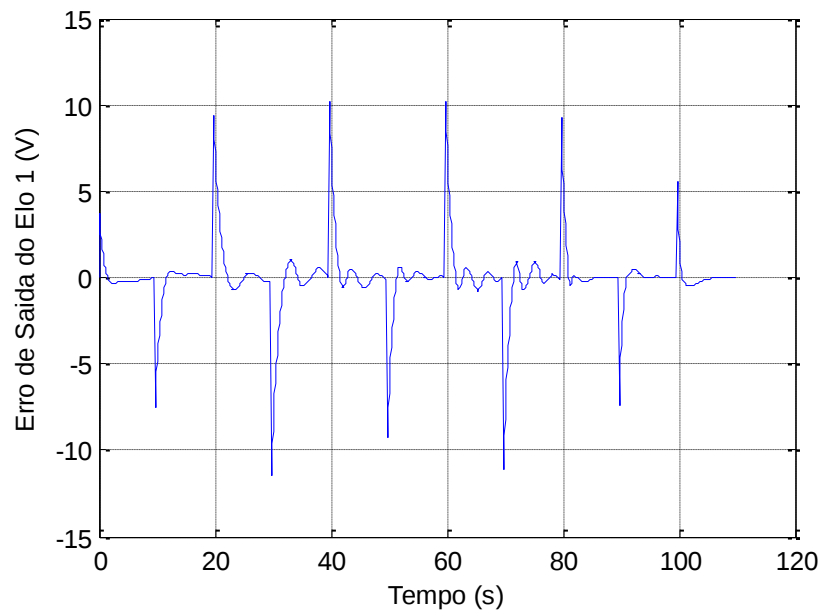


Figura 7. Erro de saída do elo 1 do robô (etapa de controle)

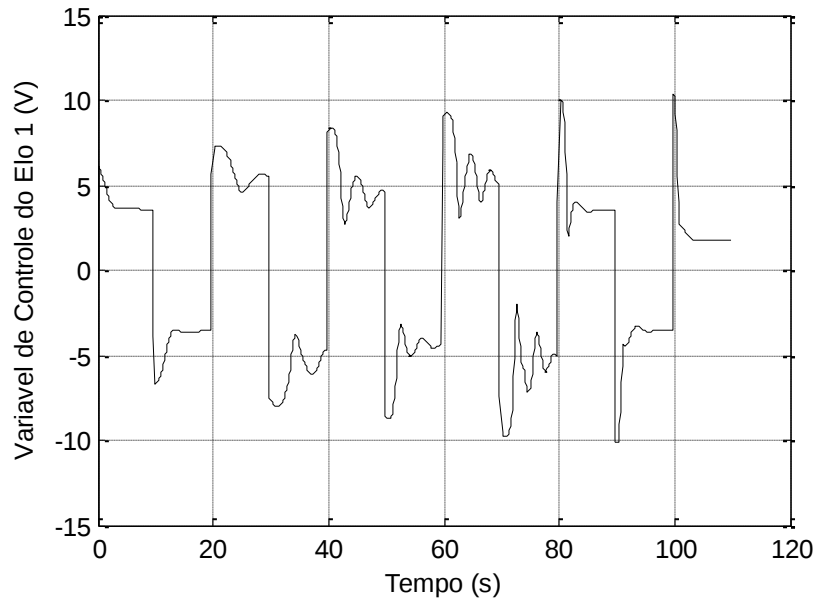


Figura 8. Variável de controle do elo 1 do robô (etapa de controle)

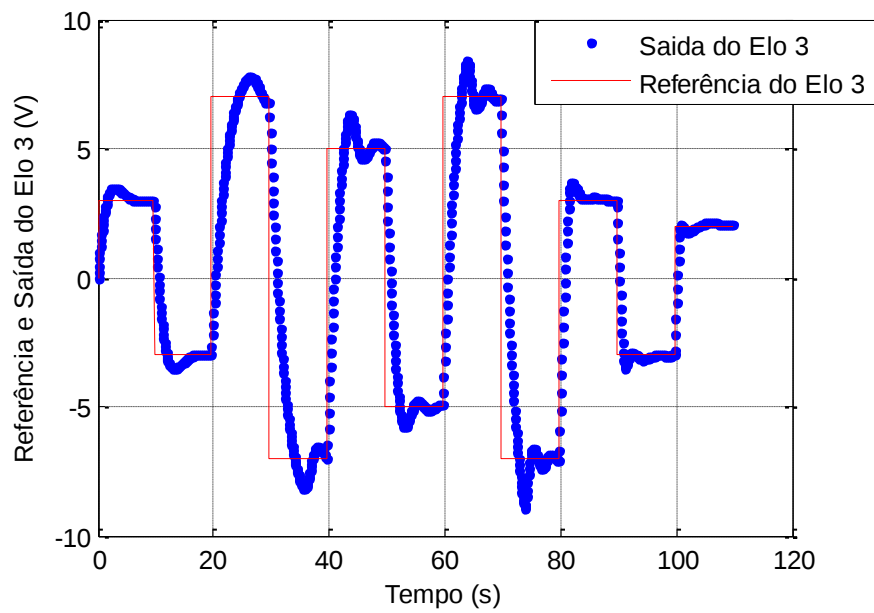


Figura 9. Referência e saída do elo 3 do robô (etapa de controle)

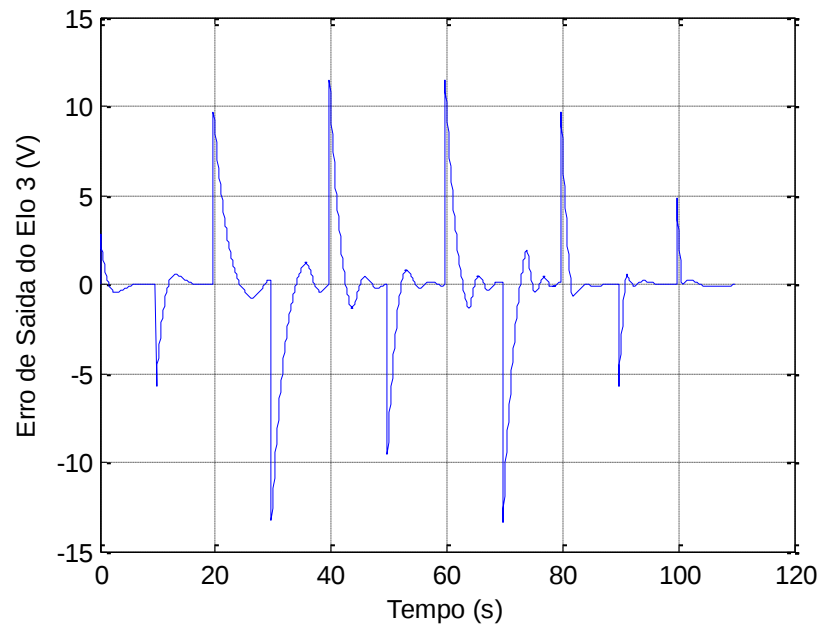


Figura 10. Erro de saída do elo 3 do robô (etapa de controle)

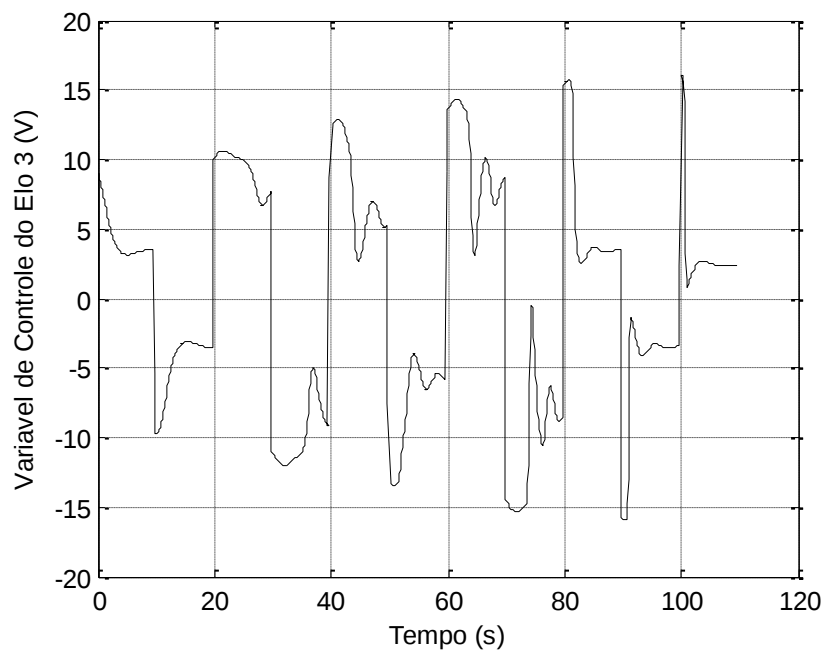


Figura 11. Variável de controle do elo 3 do robô (etapa de controle)

5. CONCLUSÃO

Este trabalho apresentou a técnica de controle neural direto, em dois elos de um robô manipulador de 3 GDL. O controle da posição dos elos do robô foi realizado utilizando-se modelos dos elos do robô. A controlador neural foi inicialmente treinado para determinadas referências e posteriormente atuou nos modelos dos elos, considerando-se outras referências; e os resultados, indicam que as saídas, em comparação com as referências, são satisfatórias, dados os níveis de sobressinal e tempo de resposta apresentados, considerando que a etapa de treinamento foi realizada com apenas uma época (iteração).

REFERÊNCIAS

- Aguirre, L. A., 2000. *Introdução à Identificação de Sistemas: Técnicas lineares e não-lineares aplicadas a sistemas reais*, 2ª edição, Belo Horizonte, MG, UFMG.
- Åström, K. J., Wittenmark, B., 1995. *Adaptive Control*, 2ª ed., New York: Ed. Addison Wesley Publishing Company, Inc.
- Cavalcanti, J. H. F., 1994. *Controladores Neurais Adaptativos*, Tese de Doutorado, Universidade Federal da Paraíba, Campina Grande, Paraíba, Brasil.
- Coelho, A.A.R., Coelho, L.S., 2004. *Identificação de Sistemas Dinâmicos Lineares*, 1ª ed., Florianópolis: Ed. Universidade Federal de Santa Catarina.
- Haykin, S., 1994. *Neural Networks – A Comprehensive Foundation*, Macmillan, New York.
- Kovács, Z. L., 1996. *Redes Neurais Artificiais: Fundamentos e Aplicações*. São Paulo: Edição Acadêmica São Paulo.
- Miller III, W. T. & Sutton, R. S. & Werbos, P. J., 1995. *Neural Networks for Control*, The MIT Press, London.
- Nascimento Júnior, C. L. & Yoneyama, T., 2000. *Inteligência Artificial em Controle e Automação*, Editora Edgard Blucher Ltda, 1ª Edição.