

## Controle de Posição de um Robô Manipulador SCARA utilizando Redes Neurais Artificiais com Treinamento On-Line\*

Sandro Battistella                      Edson R. De Pieri

Marcelo R. Stemmer

Laboratório de Controle e Microinformática - LCMI

Departamento de Automação e Sistemas - DAS

Universidade Federal de Santa Catarina - UFSC

Florianópolis - Brasil

E-mails: sandro@lcmi.ufsc.br, edson@lcmi.ufsc.br, marcelo@lcmi.ufsc.br

### Abstract

*The objective of this work is the evaluation of the employment of artificial neural networks in position control of robot manipulators. Employing a on-line training method with MLP networks, it is possible to obtain more compact structures, avoiding the need for a training set to train the neural network. Acting in a parallel structure together with a classical PD controller, the neural network is capable to compensate the non-linearities, disturbances and non-modelled dynamics existing on the robot structure. The neural controller performance is compared with the classical PD control in the joint space. Numerical simulations of the whole structure of a SCARA robot are shown.*

### 1 Introdução

No contexto da teoria de controle, as RNA's são vistas como um formalismo para modelagem de processos, empregadas como estruturas para representação do conhecimento da planta e para o mapeamento de suas características que são armazenadas na RNA [1]. A propriedade de mapeamento funcional não-linear fornece às RNA's uma forte capacidade de representação interna de problemas, tornando-as atrativas para o seu emprego em controle de processos [2], [1], [3], [4], [5].

As técnicas clássicas de controle de robôs manipuladores ou aquelas baseadas em modelo (controle de posição e/ou força) envolvem o conhecimento das equações que descrevem o comportamento dinâmico do manipulador. Nesses casos, existe uma certa dificuldade inerente ao processo de levantamento desse modelo dinâmico devido às incertezas a respeito dos parâmetros do manipulador, estando esse comportamento também sujeito às influências externas como perturbações e variações paramétricas [6], [7], [8] e [9].

A dificuldade na modelagem da equação dinâmica do manipulador vem motivando o emprego de RNA's

no controle de robôs. Uma rede neural pode ter uma performance melhor quando não precisa aprender muita informação a respeito do processo [10]. Em estruturas em que a rede neural assume um papel auxiliar ao controlador do manipulador (compensação de não-linearidades, minimizar esforço de controle, corrigir um modelo paramétrico fixo, estimar valores para controladores adaptativos, etc.), a complexidade na sua utilização diminui, dispensando, muitas vezes, o treinamento *off-line*. Além disso, a dificuldade em se levantar um conjunto de exemplos adequado não se constitui de tarefa fácil, uma vez que a dinâmica do manipulador é descrita por equações não-lineares, acopladas e multivariáveis, exigindo redes com um número elevado de neurônios [11], [12] e [4].

Nesse artigo serão vistos os problemas envolvidos no controle de posição no espaço livre (sem restrição ao movimento do manipulador). Simulações numéricas para o caso do controle de um manipulador SCARA são realizadas, com o objetivo de comparar a performance do controlador clássico de juntas PD e do o controlador neural proposto.

### 2 Redes Multilayer Perceptrons - MLP

Uma rede neural artificial consiste de um conjunto de unidades de processamento, chamadas de neurônios, interligadas através de conexões, ou pesos. Uma RNA busca aproximar o funcionamento real de uma rede de neurônios. O neurônio artificial apresenta  $n$  terminais de entrada, que representam os dendritos, responsáveis por receber os sinais provenientes dos outros neurônios, e um terminal de saída, que representa o axônio, responsável por transmitir os impulsos eletroquímicos aos demais neurônios. A cada um dos terminais de entrada está associado um peso, que representa o valor da conexão entre as sinapses dos neurônios.

Uma topologia muito empregada consiste das redes *Multilayer Perceptrons*, ou simplesmente, redes MLP. As redes MLP tem a capacidade de realizar mapeamentos não-lineares complexos, sendo, portanto, muito úteis em aplicações como reconhecimento de padrões e controle de processos. Elas serão adotadas nesse trabalho devido

\*Este trabalho foi realizado com apoio do CNPq

a simplicidade de implementação e treinamento.

### 3 Dinâmica do Manipulador SCARA

A forma construtiva de um robô manipulador do tipo SCARA, do inglês *Selective Compliant Articulated Robot for Assembly* pode ser vista na fig. 1.

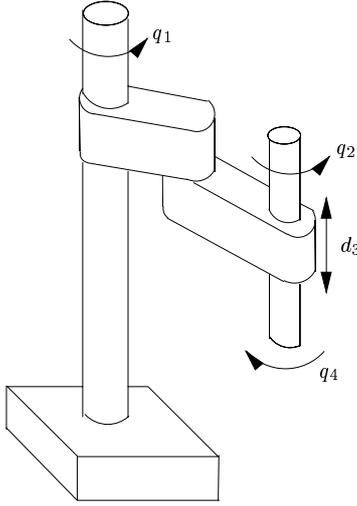


Figura 1: Robô SCARA

O modelo do manipulador, bem como seus parâmetros, correspondem ao robô manipulador SCARA, que se encontra disponível no Laboratório de Robótica da Universidade Federal de Santa Catarina.

Os deslocamentos  $\theta_i$  correspondem aos deslocamentos angulares nas juntas de rotação do manipulador, enquanto que a variável de junta  $d_3$  representa o deslocamento linear na junta vertical de translação. Os valores nominais de massa dos elos  $m_i$ , momentos de inércia  $I_i$ , comprimentos dos elos  $l_i$  e centros de massa  $l_{ci}$  são geralmente obtidos em catálogos ou através de estimações. O modelo adotado não considera o atrito e a flexibilidade existente nas juntas.

A equação dinâmica para um manipulador rígido obtida pelo método de Lagrange, e desprezando-se o atrito nas suas juntas, é definida pela equação 1 [9], [8] e [7]:

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (1)$$

onde  $q$  é o vetor ( $n \times 1$ ) de coordenadas generalizadas das juntas,  $H(q)$  é a matriz de inércia ( $n \times n$ ) positiva-definida e simétrica,  $C(q, \dot{q})$  é a matriz ( $n \times n$ ) de forças e torques centrífugos e de Coriolis,  $G(q)$  é o vetor ( $n \times 1$ ) de forças gravitacionais, e  $\tau$  é o vetor ( $n \times 1$ ) de torque de controle sobre as juntas. Para um manipulador robótico do tipo SCARA, as matrizes para a equação 1 são apresentadas a seguir.

A matriz de inércia  $H(q)$  é dada por:

$$H(q) = \begin{bmatrix} h_{11} & h_{12} & 0 & I_4 \\ h_{21} & h_{22} & 0 & I_4 \\ 0 & 0 & m_3 + m_4 & 0 \\ I_4 & I_4 & 0 & I_4 \end{bmatrix} \quad (2)$$

onde os termos  $h_{ij}$  valem

$$\begin{aligned} h_{11} &= I_1 + I_A + m_1 l_{c1}^2 + (m_2 + m_A) l_1^2 \\ &\quad + \left(\frac{m_2}{k} + m_A\right) (2l_1 l_2 \cos\theta_2) + \left(\frac{m_2}{k^2} + m_A\right) l_2^2 \\ h_{12} &= I_A + \left(\frac{m_2}{k} + m_A\right) l_1 l_2 \cos\theta_2 + \left(\frac{m_2}{k^2} + m_A\right) l_2^2 \\ h_{21} &= I_A + \left(\frac{m_2}{k} + m_A\right) l_1 l_2 \cos\theta_2 + \left(\frac{m_2}{k^2} + m_A\right) l_2^2 \\ h_{22} &= I_A + \left(\frac{m_2}{k^2} + m_A\right) l_2^2 \end{aligned} \quad (3)$$

com  $I_A = I_2 + I_3 + I_4$ ,  $k = \frac{l_2}{l_{c2}}$  e  $m_A = m_3 + m_4$ .

A matriz de torques de Coriolis e torques centrífugos  $C(q, \dot{q})$  é dada pela seguinte relação:

$$C(q, \dot{q})\dot{q} = \begin{bmatrix} c_{11} & c_{12} & 0 & 0 \\ c_{21} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4)$$

onde os coeficientes  $c_{ij}$  da matriz são dados por

$$\begin{aligned} c_{11} &= -\left(\frac{m_2}{k} + m_A l_1 l_2\right) \sin\theta_2 \dot{\theta}_2 \\ c_{12} &= -\left(\frac{m_2}{k} + m_A\right) l_1 l_2 \sin\theta_2 (\dot{\theta}_1 + \dot{\theta}_2) \\ c_{21} &= \left(\frac{m_2}{k} + m_A\right) l_1 l_2 \sin\theta_2 \dot{\theta}_1. \end{aligned} \quad (5)$$

O vetor de torques gravitacionais, por sua vez, é definido como sendo

$$G(q) = [0 \ 0 \ (m_3 + m_4)g \ 0]^T \quad (6)$$

onde  $g$  é o valor da aceleração da gravidade ( $g = 9.81 \text{ m/s}^2$ ).

Os valores dos parâmetros e coeficientes do manipulador SCARA utilizados nesse trabalho encontram-se listados na tabela 1.

### 4 Controle de Posição

Quando o robô desempenha tarefas que não geram interações entre a dinâmica do meio e do próprio manipulador - manipulador no espaço de tarefa livre - o objetivo do sistema de controle consiste basicamente em comandar a posição, a velocidade, e eventualmente, a aceleração nas juntas do manipulador.

#### 4.1 Controle PD/PID

Dois dos controladores mais simples inicialmente utilizados para o controle de posição, constituem-se dos controladores clássicos PD e PID (equações 7 e 8):

Tabela 1: Parâmetros do manipulador

Parâmetro	Valor
Massa do elo 1 ( $m_1$ )	11.4 Kg
Massa do elo 2 ( $m_2$ )	19.5 Kg
Massa do elo 3 ( $m_3$ )	2 Kg
Massa do elo 4 ( $m_4$ )	1.5 Kg
Inércia do elo 1 ( $I_1$ )	0.23 Kg m <sup>2</sup>
Inércia do elo 2 ( $I_2$ )	0.16 Kg m <sup>2</sup>
Inércia do elo 3 ( $I_3$ )	0.1 Kg m <sup>2</sup>
Inércia do elo 4 ( $I_4$ )	0.1 Kg m <sup>2</sup>
Comprimento do elo 1 ( $l_1$ )	0.25 m
Comprimento do elo 2 ( $l_2$ )	0.25 m
Centro de massa do elo 1 ( $l_{c1}$ )	0.118 m
Centro de massa do elo 2 ( $l_{c2}$ )	0.116 m
Origem do sistema de coordenadas da terceira junta $d_0$	0.678 m

$$\tau = K_p(q_d - q) - K_d\dot{q} \quad (7)$$

$$\tau = -K_d\dot{q} + K_p(q_d - q) + K_i \int_0^t (q_d - q) \partial\tau \quad (8)$$

onde  $K_{(.)}$  são as matrizes de ganho dos controladores, de ordem ( $n \times n$ ), positivas e geralmente diagonais, e  $q_d$  é o valor de referência para o deslocamento nas juntas  $q$ .

## 4.2 Controle Neural

A lei de controle neural adotada nesse trabalho, consiste basicamente em um controlador clássico do tipo PD e uma rede neural sendo empregada para compensar as não-linearidades e as perturbações sobre o sistema. Diferente de muitas abordagens propostas de controle neural de posição de robôs manipuladores [13], [14], [10], [15] [16], [17], a estrutura de controle não necessita de fase de treinamento prévio, sendo o treinamento realizado de forma *on-line*, ou seja, atuando sobre o processo.

O objetivo da estrutura de controle proposta consiste em gerar o sinal  $\tau$  através da equação

$$\tau = u_c + \phi_n \quad (9)$$

onde  $u_c$  é o sinal de controle proveniente de um PD digital, empregado para estabilizar o processo durante o regime transitório. A figura 2 apresenta a estrutura de controle proposta.

A rede neural utiliza o sinal de controle  $u_c$  para o seu treinamento. Interpretando esse sinal como o erro de treinamento, a rede neural busca ajustar seus pesos de forma a minimizar esse erro, fornecendo um valor de torque  $\phi_n$  nas suas saídas (uma para cada junta). Dessa forma, obtém-se, indiretamente, a minimização do erro de posição e velocidade, pois o controlador PD consiste da soma do produto dos ganhos pelos erros de posição e velocidade.

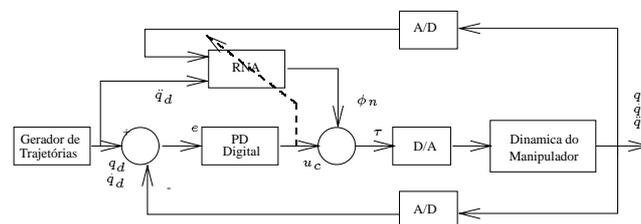


Figura 2: Proposta de Controle Neural para Juntas

## 5 Objetivo das Simulações

As trajetórias de referência empregadas constituem-se de curvas do tipo *spline* e são apresentadas na figura 3.

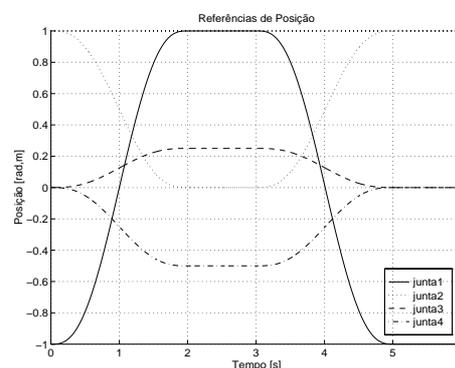


Figura 3: Referência de posição nas juntas

Os desempenhos dos controladores PD e Neural são analisados em três situações distintas: condições nominais de operação; variações e incertezas paramétricas e perturbações; e movimentação de uma carga desconhecida acoplada ao efetuador final.

O período de amostragem empregado foi de 2.5 ms. As simulações foram feitas nos softwares *MATLAB* 4.2 e *SIMULINK* 1.3c.

Os controladores implementados são digitais. Todos os controladores foram projetados levando em consideração somente as condições nominais de operação.

## 6 Resultado das Simulações

Na estrutura de controle neural, foi empregada uma rede MLP, com quatro camadas de neurônios. A camada de entrada contém 12 neurônios - quatro para aos valores de posição  $q$ , quatro para os valores de velocidade nas juntas  $\dot{q}$ , e mais quatro neurônios para os valores de aceleração desejadas  $\ddot{q}_d$ . As duas camadas escondidas possuem, respectivamente, oito e cinco neurônios. A última, apresenta quatro neurônios, um para cada junta do manipulador. Foi escolhida como função de ativação dos

neurônios a função tangente hiperbólica, com inclinação igual a um.

Para o treinamento da rede neural utilizou-se o algoritmo *backpropagation*. As matrizes de pesos e *bias* foram inicializadas com valores randômicos  $\leq 0.05$ . A taxa de aprendizado empregada é da ordem de 0.001.

### 6.1 Condições Nominais

Considerando o manipulador livre de perturbações externas e sem imprecisões na sua modelagem, observou-se o desempenho dos controladores para o seguimento da trajetória da figura 3.

As figuras 4 e 5 apresentam a performance dos controladores de juntas.

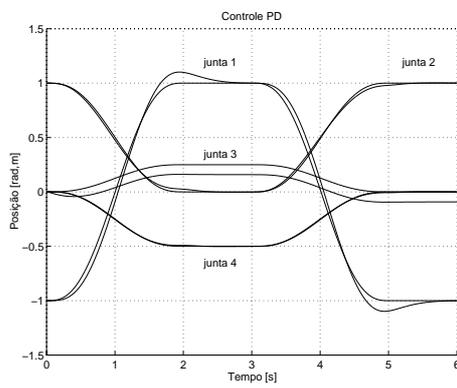


Figura 4: Controle PD: Caso nominal

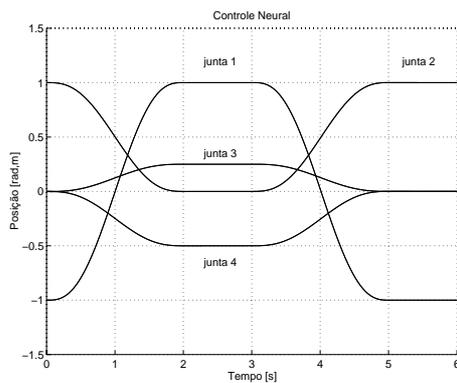


Figura 5: Controle Neural: Caso nominal

A existência de um erro em regime permanente do posicionamento da terceira junta, ocorre em função da não compensação do termo de gravidade pelo controlador PD (figura 4).

Por sua vez, o controlador neural apresenta um seguimento com um erro muito pequeno, em comparação ao controle PD (figura 5). Observa-se que a capacidade de compensação que a rede oferece ao sistema de controle, no caso, contrabalança o efeito gravitacional existente na terceira junta do manipulador.

### 6.2 Variação Paramétrica e Perturbações

Aqui considera-se a existência de imprecisões no conhecimento dos parâmetros, decorrente de possíveis erros de estimação ou variações paramétricas devido às influências do meio. As variações consideradas são: + 20% para  $m_1$  e  $m_4$ , - 20% para  $m_3$ , + 10% para  $I_1$  e  $I_4$ , - 10% para  $I_3$ , + 15% para  $l_1$  e - 15% para  $l_2$ .

De forma a verificar o comportamento dos controladores frente a existência de perturbações, foram aplicados torques contrários ao movimento do manipulador. No instante 3 segundos foi aplicada na junta 1 uma perturbação de 20% do valor máximo permitido para essa junta, e outro sinal no instante 4 segundos na junta 3, também com valor de 20% do máximo torque permitido.

As figuras 6 e 7 apresentam a performance dos controladores para os casos de perturbação e variação de parâmetros.

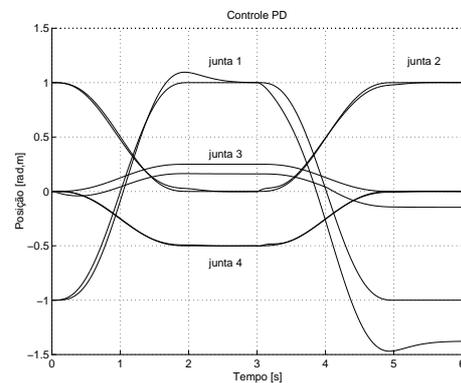


Figura 6: Controle PD: Variação paramétrica e perturbação

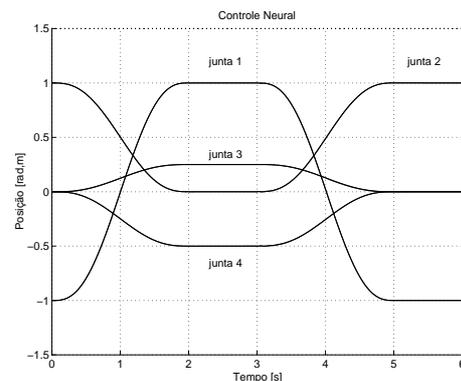


Figura 7: Controle Neural: Variação paramétrica e perturbação

O controlador PD não sofre muita influência das variações paramétricas. Essa característica pode ser vista nos primeiros 3 segundos do seguimento de trajetória (figura 6). Nesse caso, os ganhos do controlador ainda são suficientes para manter uma boa performance de controle frente as variações paramétricas. Entretanto, observa-se

que o erro devido ao efeito de gravidade é agravado pela perturbação existente na terceira junta.

Na estratégia de controle neural, a rede é capaz de compensar as variações paramétricas e as perturbações atuando sobre o manipulador. No caso simulado (figura 7) a influência da perturbação não chega a ser visível, garantindo um seguimento de trajetória similar ao do caso nominal. Isso ocorre, uma vez que a rede atua nas situações em que o controlador PD não consegue garantir uma boa performance de controle, como é o caso da existência de perturbações e incertezas ou variações paramétricas.

### 6.3 Variação de Carga

Procurando executar a trajetória da figura 3, observa-se a performance dos controladores para o caso em que é considerado o transporte de uma carga desconhecida. O valor da carga não se encontra incluído no modelo dinâmico do manipulador. A carga possui uma massa de 5 Kg, e encontra-se acoplada ao manipulador nos primeiros 4 segundos da trajetória.

As figuras 8 e 9 apresentam o desempenho dos controladores para o caso de variação de carga.

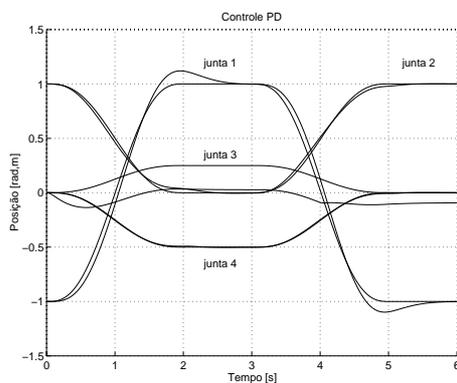


Figura 8: Controle PD: Variação de carga

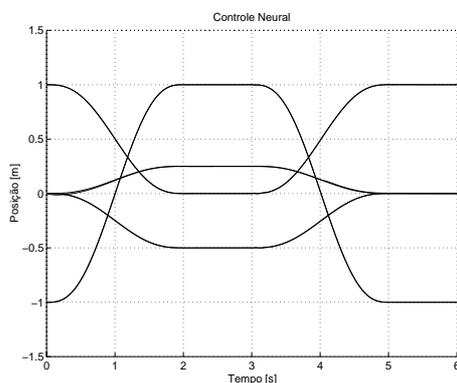


Figura 9: Controle Neural: Variação de carga

O acoplamento de uma carga no efetuador final, corresponde em um acréscimo de massa na terceira junta.

Dessa forma, os efeitos dos torques gravitacionais são mais relevantes, como pode ser visto na figura 8.

Novamente a estrutura de controle neural foi capaz de satisfazer os requisitos de controle. A rede, nesse caso, funciona como um compensador de gravidade. O torque provido pela rede é capaz de suprir a deficiência da estrutura de controle do tipo PD, que não apresenta a capacidade de compensação dos termos de gravidade (figura 9).

### 6.4 Análise do Esforço de Controle

As figuras 10 e 11 apresentam as ações de controle para o caso onde existem variações paramétricas e perturbações atuando sobre o manipulador.

Observa-se que o controlador neural apresenta uma ação de controle mais oscilatória. Esse inconveniente acontece devido à dois fatores. A primeira influência é decorrente da necessidade de ajuste dos pesos da rede frente às mudanças de referência ou à existência de uma perturbação. O segundo fator provém da falta de conhecimento sobre a interação entre o controlador clássico e a rede neural, não sendo conhecida uma forma de estabelecer critérios de projeto e análise de estabilidade do sistema manipulador/controlador + rede neural, de forma a incluir a otimização da lei de controle durante a fase de projeto do controlador. Além disso, o treinamento da rede neural encontra-se baseado somente no erro de posicionamento, e não leva em conta o esforço da ação de controle.

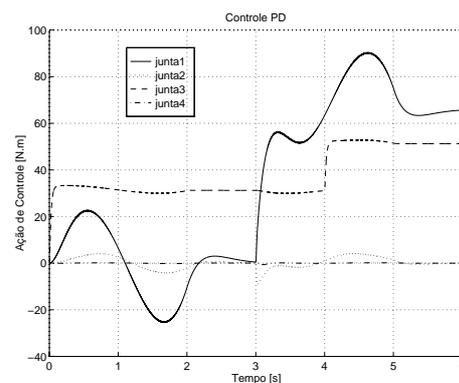


Figura 10: Controle PD: Ação de controle

## 7 Conclusões

O emprego de estruturas adaptativas em controle de robôs é uma alternativa às técnicas de controle convencional que necessitam do conhecimento analítico do modelo do manipulador. A vantagem do emprego de redes neurais reside justamente na não necessidade do conhecimento da planta, sendo capaz de aprender o modelo dinâmico do manipulador, incluindo as não-linearidades, e também as incertezas.

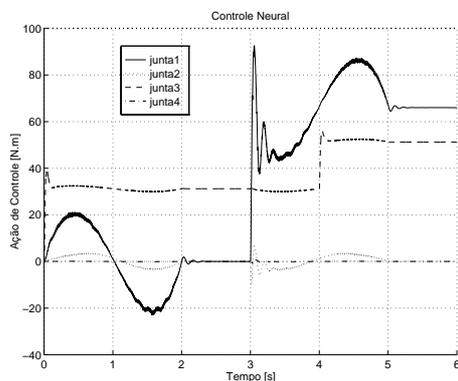


Figura 11: Controle Neural: Ação de controle

Entretanto, a necessidade de um treinamento prévio *off-line*, com o emprego de um conjunto de exemplos obtidos do sistema real, pode ser um entrave a sua utilização em robótica. Primeiro porque, de acordo com a qualidade do conjunto de exemplos utilizado e com a forma de treinamento empregada, não é possível prever o comportamento da rede para situações em que ela não foi treinada. Isto é, a capacidade de generalização da rede pode não ser suficiente para garantir uma performance satisfatória do sistema de controle. Segundo, o processo de treinamento geralmente é lento e exige um custo computacional elevado. Em manipuladores robóticos de mais de três graus de liberdade, a quantidade de dados é relativamente alta, aumentando o tempo de treinamento.

Nas tarefas de seguimento de trajetória de posição os controladores clássicos do tipo PD/PID não são suficientes pois não apresentam boas propriedades frente à existência de perturbações e variações paramétricas. Estratégias de controle que incluem um modelo paramétrico do robô em sua estrutura, como o controle inverso de juntas ou o controle híbrido de força/posição, procuram aumentar a performance da sua estrutura de controle. Porém, essas estratégias necessitam de um conhecimento mais preciso sobre os parâmetros do modelo do manipulador. A falta de precisão no modelo estimado empregado na lei de controle, pode levar o sistema a apresentar um desempenho fora das especificações. Dessa forma, uma rede neural que complementa o ação de um controlador principal, assumindo o papel de compensar não-linearidades e perturbações não-modeladas atuantes sobre a dinâmica do manipulador, apresentou-se como uma solução adequada à falta de conhecimento sobre essa dinâmica.

Entretanto, as estruturas de controle neural apresentaram ações de controle mais oscilatórias, em comparação com as estratégias convencionais de controle de posição e controle híbrido força/posição. Outra desvantagem, consiste na necessidade de existir um controlador principal (controlador PD). Além disso, ainda não se encontram estabelecidas provas e critérios de estabilidade para o sistema robô/controlador + rede neural.

A estratégia de controle aqui apresentada deverá ser implementada a seguir em um robô real disponível no

Laboratório de Robótica da UFSC. A estratégia está também sendo alterada de forma a permitir o controle de força, que será tema de outro artigo em futuro próximo.

## Referências

- [1] K. J. Hunt, D. Sbarbaro, R. Zbikowski, and P. J. Ganthrop. Neural networks for control systems - a survey. *Automatic*, 28(6):1385–1406, 1992.
- [2] I. H. S. Filho and M. R. Stemmer. *Redes Neurais*. LCMI - UFSC - Florianópolis, 1993.
- [3] K. S. Narendra. Neural networks for control: Theory and practice. *Proceedings of IEEE*, 84(10):1385–1406, October 1996.
- [4] T. Fukuda and T. Shibata. Theory and applications of neural networks for industrial control systems. *IEEE Transactions on Industrial Electronics*, 39(6):472–489, December 1992.
- [5] M. Agarwal. A systematic classification of neural network-based control. *IEEE Control Systems*, pages 75–93, April 1997.
- [6] L. S. Wilfinger. A comparison of force control algorithms for robots in contact with flexible environments. Master's thesis, Rensselaer - Polytechnic Institute, Troy, New York, 1992.
- [7] C. C. de Wit, B. Siciliano, and G. Bastin. *Theory of Robot Control*. Springer-Verlag, 1996.
- [8] F. L. Lewis, C. T. Abdallah, and D. M. Dawson. *Control of Robot Manipulators*. MacMillan Publishing Co., 1993.
- [9] H. Asada and J. J. E. Slotine. *Robot Analysis and Control*. John Wiley and Sons, 1986.
- [10] A. Ishiguro, T. Furuhashi, S. Okuma, and Y. Uchikawa. A neural network compensation for uncertainties of robotics manipulators. *IEEE Transactions on Industrial Electronics*, 39(6):565–569, December 1992.
- [11] D. Katic and M. Vukobratovic. Connectionist based robot control: An overview. *IFAC - 13<sup>th</sup> Triennial World Congress*, pages 169–174, 1996.
- [12] S. M. Zlauddin and A. M. S. Zalzal. Model-based neural network compensation for uncertainties of robot arms. *IFAC - 13<sup>th</sup> Triennial World Congress*, pages 157–162, 1996.
- [13] A. M. S. Zalzal and A. S. Morris. *Neural Networks for Robotic Control - Theory and Applications*. Ellis Horwood, 1996.
- [14] M. J. Er and K. C. Liew. Control of adept one scara robot using neural networks. *IEEE Transactions on Industrial Electronics*, 44(6):762–768, December 1997.
- [15] A. A. Neto and L. C. S. Goes. Projeto de um controlador neural para o robô ita-iemp. *III Congresso Brasileiro de Redes Neurais - IV Escola de Redes Neurais - Florianópolis*, pages 384–389, Julho 1997.
- [16] M. R. Stemmer, E. R. de Pieri, and F. A. P. Borges. Um controlador neural adaptativo para o robô manipulador puma 560. *Proceedings of XIII Brazilian Automatic Control Conference - XII CBA*, 5:1633–1638, Setembro 1998.
- [17] M. R. Stemmer, E. R. de Pieri, and F. A. P. Borges. Comparação de performance entre controladores clássicos e um controlador torque computado neural aplicados ao robô puma 560. *III Congresso Brasileiro de Redes Neurais - IV Escola de Redes Neurais - Florianópolis*, pages 390–395, Julho 1997.