



**COTEQ-051**

**CONTROLE POR APRENDIZADO NEURO - FUZZY PARA INCREMENTAR A  
FREQUÊNCIA DE TESTES DE MÁQUINAS DE ENSAIOS DE FADIGA**

Eleazar Cristian Mejía Sánchez<sup>1</sup>, Juan Gerardo Castillo Alva<sup>2</sup>, Marco Antonio Meggiolaro<sup>3</sup>,  
Jaime Tupiassu Pinho de Castro<sup>3</sup>

*Copyright 2011, ABENDI, ABRACO e IBP.*

*Trabalho apresentado durante a 11ª Conferência sobre Tecnologia de Equipamentos.*

*As informações e opiniões contidas neste trabalho são de exclusiva responsabilidade dos autores.*

---

***SINOPSE***

Neste trabalho apresentam-se técnicas de controle por aprendizado neuro-fuzzy desenvolvidas e utilizadas em máquinas de ensaios de materiais, permitindo a aplicação de carregamentos variáveis ou constantes e o aumento da frequência de trabalho em relação aos métodos tradicionais de controle, por exemplo, PID. Usa-se uma metodologia de controle do tipo bang-bang para restringir a servo-válvula da máquina a trabalhar nos seus limites extremos de operação, procurando mantê-la completamente aberta em uma ou outra direção. Os instantes ótimos de reversão são obtidos pelo algoritmo de aprendizado neuro-fuzzy proposto. Ao usar técnicas de inteligência computacional misturadas com o algoritmo de aprendizado, obtém-se um agente com a capacidade de aprendizado e com um menor custo computacional no armazenamento das variáveis adimensionais, as quais são constantemente atualizadas pelas leis de aprendizado durante o teste, melhorando a resposta do sistema. O sistema de controle é validado experimentalmente em uma máquina servo-hidráulica de ensaios de fadiga com capacidade de 100 kN. Para isso, implementou-se um *software* de controle em tempo real em um sistema computacional CompactRIO. Os resultados experimentais mostram que a frequência dos testes pode ser significativamente aumentada com a técnica de controle proposta, demonstrando também a funcionalidade do sistema de controle para trabalhar com ensaios de carregamentos variáveis.

---

1 Mestre, Engenheiro Mecatrônico – PUC-Rio

2 Mestre, Engenheiro Eletrônico – PUC-Rio

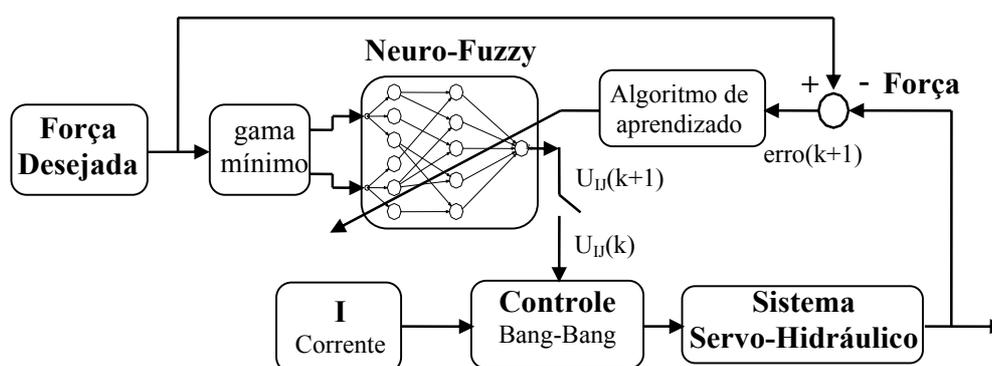
3 PHD, Engenheiro Mecânico – PUC-Rio

## 1. INTRODUÇÃO

Nos últimos anos, sistemas baseados em redes neurais e fuzzy foram encontrando um caminho em aplicações em controle e muitas outras áreas da engenharia. Um esquema de controle por redes neurais não requer a computação da dinâmica não-linear de equipamentos complexos, somente utiliza sinais medidos localmente para fazer a aprendizagem do comportamento do sistema através da atualização dos pesos da rede que minimizem o erro entre a saída desejada e a real. Os resultados obtidos capturaram a atenção de engenheiros que trabalham com sistemas reais. Isto é devido principalmente aos resultados obtidos e frequentemente à facilidade de implementação quando se desenvolvem sistemas de controle baseados em redes neurais e fuzzy (1). Branco e Dente apresentaram o uso de um controlador por aprendizado, utilizando técnicas Neuro-Fuzzy no projeto de um controle de trajetória, modelando o atuador eletro-hidráulico (2). Numa outra aplicação utiliza um controle híbrido adaptativo Neuro-Fuzzy por modelo de referência (“Adaptive Neuro-Fuzzy Model Reference Controller”, ANFMRC). Este modelo foi desenvolvido para melhorar o desempenho do controle em um sistema pneumático. Nele, um controle híbrido é combinado com um controle “bang-bang” aplicado quando o erro é alto, e um ANFMRC aplicado quando o erro é pequeno (3).

## 2. CONTROLE POR APRENDIZADO NEURO-FUZZY

O processo de aprendizado pode ser visto como um problema de estimação ou aproximações sucessivas de funções ou quantidades desconhecidas (4). Nesse caso, as quantidades desconhecidas que são estimadas ou aprendidas pelo controlador são parâmetros que são governados pelas leis de controle. Na Figura 1 apresenta-se o diagrama de blocos que ilustra o controle por aprendizado Neuro-Fuzzy. Neste modelo de controle, a informação é representada por uma variável adimensional  $U_{IJ}$ , que nesse caso é a saída do sistema Neuro-Fuzzy. Esta variável armazena o instante em que a servo-válvula deve ser revertida durante seu curso, e é um número entre 0 e 1 (0 para reversão no início do curso, até 1 para reversão ao final do curso). A informação que gera o valor de  $U_{IJ}$  é armazenada nos pesos da estrutura do sistema Neuro-Fuzzy. A variável  $U_{IJ}$ , utilizada para mudar o instante da ação de controle sobre a servo-válvula, é atualizada após cada ciclo de operação através do ajuste dos pesos da estrutura Neuro-Fuzzy, utilizando um algoritmo de aprendizado baseado nos erros medidos. O objetivo do sistema Neuro-Fuzzy é fornecer o valor de  $U_{IJ}$  ao controlador, determinando assim o instante de reversão no qual a servo-válvula vai reverter seu sentido, de maneira que a máquina (o sistema servo-hidráulico) seja mantida trabalhando em seus limites de operação.

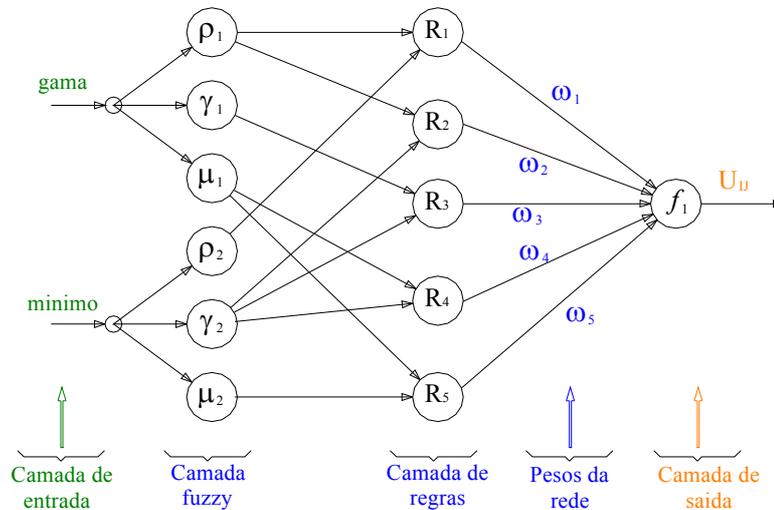


**Figura 1:** Diagrama de blocos do controle por aprendizado Neuro-Fuzzy.

Na Figura 1, o valor de  $U_{IJ}$  é atualizado pela chave com o valor de  $U_{IJ}(k+1)$ , e o erro normalizado  $erro(k+1)$  é obtido em função da força desejada e da força real medida. As entradas do sistema Neuro-Fuzzy são  $gama$  (o dobro da amplitude do carregamento desejado) e o valor mínimo da grandeza controlada, e a saída do sistema é a variável  $U_{IJ}$ .

### 2.1 Sistema Neuro-Fuzzy

O sistema Neuro-Fuzzy é constituído por duas camadas escondidas: a camada Fuzzy e a camada de regras. Os pesos do sistema Neuro-Fuzzy  $\omega_j$  entre a camada de regras e a camada de saída são atualizados utilizando o algoritmo de aprendizado *Backpropagation*, baseado no  $erro(k+1)$  a cada iteração. Na Figura 2 apresenta-se a estrutura do sistema proposto.



**Figura 2:** Estrutura do Sistema Neuro-Fuzzy

### 2.2 Modelagem do Sistema Neuro-Fuzzy

Este controle é baseado em sistemas híbridos Neuro-Fuzzy (SNF), constituído pela combinação de sistemas Fuzzy e Redes Neurais (ANN). Ele combina as vantagens das ANNs tais como a habilidade de aprendizagem, otimização, e conexão em estrutura, com as vantagens dos sistemas Fuzzy, que usa raciocínio semelhante ao humano, com facilidade de incorporar informações de especialistas.

#### 2.2.1 Modelagem Fuzzy

A modelagem das características Fuzzy é determinada pela configuração dos parâmetros das seguintes 4 categorias:

**Modelo Fuzzy:** O modelo de inferência Fuzzy (formato das regras) implementado foi o modelo *Takagi-Sugeno*, com conjuntos Fuzzy de saída do tipo "singleton" para cada uma das regras. As regras fuzzy são da seguinte forma:

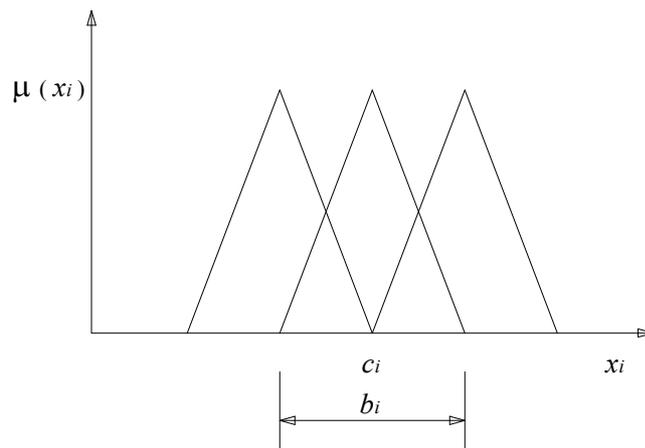
$$\text{Regra } j: \text{ Se "gama" é } \rho_1 \text{ e "mínimo" é } \mu_2 \text{ Então } U_{IJ} = \omega_j \quad (1)$$

**Formato das Funções de Pertinência:** As funções de pertinência na camada Fuzzy são geralmente funções simétricas, tais como funções triangulares, trapezoidais, e sigmoidais. Neste trabalho, escolheram-se oito funções de pertinência do tipo triangular para cada uma

das variáveis de entrada, pela simplicidade para sua implementação experimental. Na Figura 3 ilustra-se a função de pertinência triangular, cuja equação é expressa por:

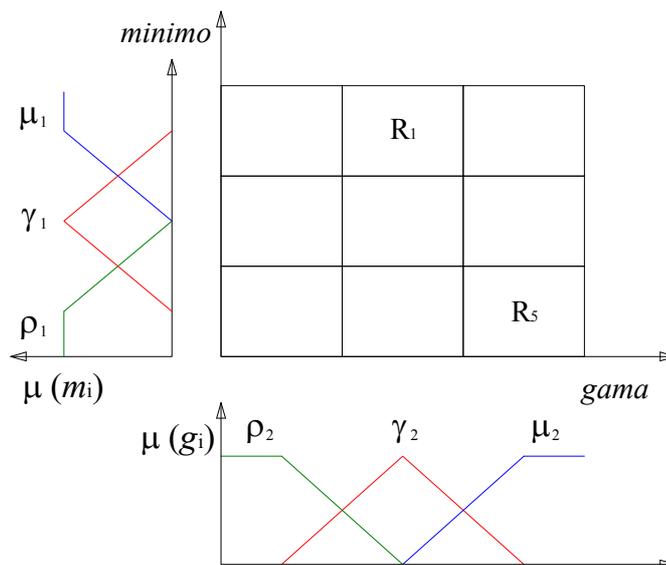
$$\mu(x_i) = 1 - \frac{2|x_i - c_i|}{b_i} \quad (2)$$

onde  $x_i$  é o valor da variável de entrada,  $c_i$  é o centro do triângulo da função de pertinência, e  $b_i$  é a largura da base do triângulo.



**Figura 3.** Função de pertinência triangular do *SNF*.

*Particionamento do Espaço de E/S:* O particionamento do espaço das variáveis de entrada e saída (E/S) é do tipo fuzzy grid, o qual mapeia internamente regiões fuzzy relacionadas através de suas regras. Na Figura 4 apresenta-se o particionamento fuzzy grid, onde  $\rho_1, \gamma_1, \mu_1$  e  $\rho_2, \gamma_2, \mu_2$  são os graus de pertinência dos conjuntos Fuzzy das variáveis “*mínimo*” e “*gama*”, respectivamente.



**Figura 4.** Particionamento Fuzzy Grid.

*Método de Defuzzificação:* O consequente das regras é uma função do tipo “*singlenton*”, e a saída do sistema,  $U_{IJ}$ , é obtida da média ponderada dos graus de disparo de cada regra. A saída  $U_{IJ}$  é calculada por

$$U_{IJ} = f(P(\mu_i, \mu_i), \omega) \quad (3)$$

onde  $P(\mu_i, \mu_i)$  é o nível de disparo correspondente a cada regra, e  $\omega$  é o peso da estrutura (saída “*singlenton*”).

### 2.2.2 Modelagem da Parte Neural

A modelagem das características da parte neural está relacionada com a capacidade de aprendizado do *SNF*, e determinada pela configuração dos parâmetros das seguintes subclasses:

*Tipo de aprendizado:* O aprendizado utilizado é do tipo *online*, isto devido a que ao longo da operação da máquina o sistema tem que ter a capacidade de mudar os instantes de reversão, de maneira que os pesos do *SNF* têm que ser atualizados a cada iteração.

*Identificação da estrutura:* O número de regras é determinado pela combinação dos conjuntos fuzzy das variáveis de entrada.

*Identificação dos parâmetros:* Neste modelo de *SNF* só apresentam aprendizado nos parâmetros do consequente de cada regra, baseado na medida do erro normalizado.

## 3. CONTROLE POR APRENDIZADO NEURO-FUZZY

### 3.1 Cálculo do valor de $U_{IJ}$

O valor da variável adimensional  $U_{IJ}$  é obtido como o resultado da avaliação do *SNF* para cada combinação de *mínimo* e *gama*. Na aplicação considerada, os valores da variável de entrada, *mínimo* e *gama*, estão na faixa de [-100, 100]MPa e [-200, 200]MPa respectivamente. Os valores de *gama* são positivos quando o sistema esta indo de um vale para um pico e negativos quando vai de um pico para um vale. Estas variáveis de entradas do *SNF* (*mínimo* e *gama*) são normalizadas na faixa de [-1,1] utilizando

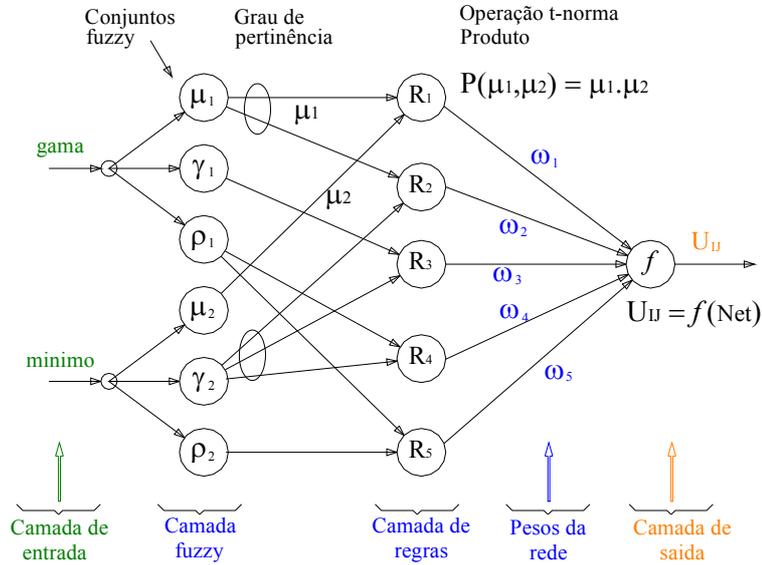
$$x_n = \frac{2 \cdot (x - \min)}{\text{Max} - \min} - 1 \quad (4)$$

onde  $x_n$  é o valor normalizado da variável  $x$ , e *min* e *Max* são os valores de mínimo e máximo da variável  $x$ . A normalização das variáveis de entrada, *mínimo* e *gama*, é feita substituindo-se na equação acima

$$\min_n = \frac{\min}{100} \quad (5)$$

$$\text{gama}_n = \frac{\text{gama}}{200} \quad (6)$$

Depois da normalização das variáveis de entrada na camada de entrada, vide Figura 5, na camada fuzzy calcula-se o grau de pertinência com que as entradas satisfazem aos conjuntos fuzzy associado a cada entrada.



**Figura 5.** Cálculo de  $U_{ij}$  e descrição das camadas do SNF.

Na camada de regras, calcula-se o nível de disparo correspondente a cada uma das regras, executando a operação *t-norm* (produto).

$$P_k(\mu_i, \mu_j) = \mu_i \cdot \mu_j \quad (7)$$

Na camada de saída calcula-se o valor de  $U_{ij}$  em função do  $P_k(\mu_i, \mu_j)$  e  $\omega_k$ :

$$U_{ij} = f(\text{Net})$$

$$\text{Net} = \frac{\sum_{k=1}^M P_k(\mu_i, \mu_j) \cdot \omega_k}{\sum_{k=1}^M P_k(\mu_i, \mu_j)} \quad (8)$$

Neste trabalho, considerou-se  $f$  como função de ativação linear pela facilidade na implementação experimental, mas pode-se aplicar outras funções do tipo sigmoidais (e.g. logsig ou tansig). Tem-se então

$$U_{ij} = \frac{\sum_{k=1}^M P_k(\mu_i, \mu_j) \cdot \omega_k}{\sum_{k=1}^M P_k(\mu_i, \mu_j)} \quad (9)$$

onde  $P_k(\mu_i, \mu_j)$  é o resultado da operação *t-norm* na camada de regras e  $\omega_k$  é o peso de conexão da regra  $k$  e o neurônio de saída.

As equações apresentadas acima determinam o valor de  $U_{IJ}$  para qualquer tipo de carregamento. Uma vez calculado o valor de  $U_{IJ}$ , os pontos (instantes) de reversão da servoválvula são calculados por

$$Ponto\_Reversão = \begin{cases} \min + U_{IJ} \cdot gama & (na\ subida) \\ (\min + gama) - U_{IJ} \cdot gama & (na\ descida) \end{cases} \quad (10)$$

### 3.2 Lei de aprendizado do SNF

No processo de aprendizagem, são feitas atualizações dos valores de  $U_{IJ}$  seguindo uma lei de aprendizado. O erro normalizado é o erro obtido entre o pico (ou vale) desejado  $x_d$  e o pico (ou vale) atingido  $x$ , dividido pela gama obtida entre o pico (ou vale) desejado  $x_d$  e o valor do vale (ou pico) atingido na reversão anterior  $x'$ , ou seja:

$$erro = \frac{x_d - x}{x_d - x'} \quad (11)$$

O aprendizado do SNF é feito pelas atualizações dos pesos  $\omega_k$  em um instante seguinte com o valor atual. Todos os valores de  $\omega_k$  são inicializados com 0,5 e depois atualizados em função do erro normalizado, a taxa de aprendizado e o nível de disparo de cada uma das regras. O erro normalizado é o erro obtido entre o pico (ou vale) desejado  $x_d$  e o pico (ou vale) atingido  $x$ , dividido pela *gama* obtida entre o pico (ou vale) desejado e o valor de vale (ou pico) atingido na reversão anterior  $x'$ , como foi definido na equação (11).

Se o carregamento aplicado ao sistema for de um vale para um pico (subida), os valores de  $x$  e  $x_d$  são picos próximos entre si e  $x'$  terá sido um vale, portanto  $x_d - x'$  se torna positivo. Assim, se  $x < x_d$  então o erro é positivo ( $erro > 0$ ), apresentando um caso de *undershoot*. No entanto, se  $x > x_d$  o erro é negativo ( $erro < 0$ ), apresentando um caso de *overshoot*.

No caso em que o sistema for de um pico para um vale (descida), os valores de  $x$  e  $x_d$  são vales próximos entre si e  $x'$  terá sido um pico, portanto  $x_d - x'$  se torna negativo. Neste caso, como o carregamento está diminuindo, se  $x > x_d$  o erro é positivo ( $erro > 0$ ), um caso de *undershoot*, enquanto que se  $x < x_d$  o erro é negativo ( $erro < 0$ ), um caso de *overshoot*.

Conclui-se assim que erros positivos estão associados ao *undershoot*, e negativos ao *overshoot*, tanto na subida (vale – pico) como na descida (pico-vale).

Em geral, o valor de *erro* esta na faixa de [-1, 1] e o algoritmo de atualização dos pesos  $\omega_k$  do SNF é dado pela lei de aprendizado

$$\omega_k(t+1) = \omega_k(t) + \Delta\omega_k(t) \quad (12)$$

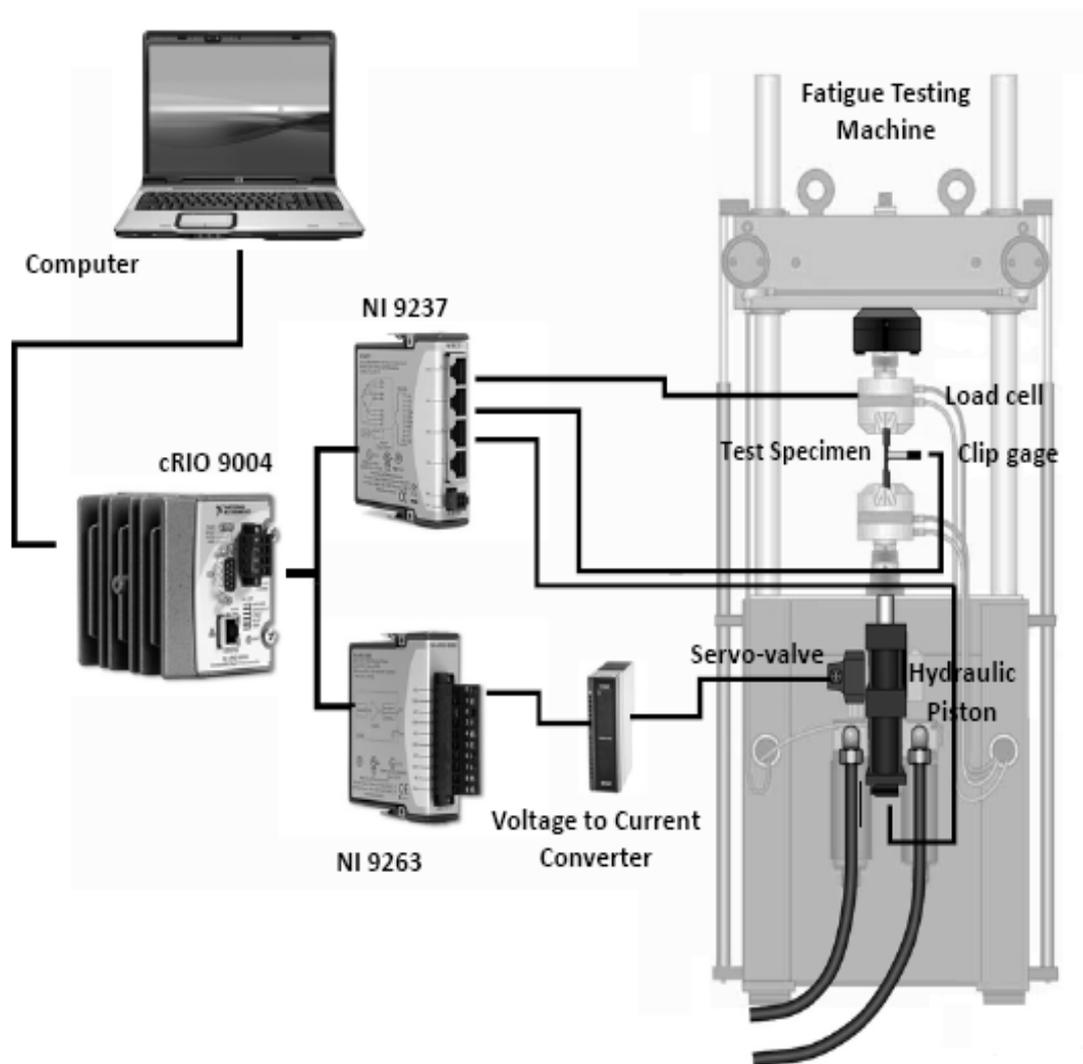
$$\Delta\omega_k(t) = \eta \cdot erro \cdot \frac{P_k(\mu_i, \mu_j)}{\max[P_k(\mu_i, \mu_j)]} \quad (13)$$

onde  $\omega_k(t)$  é o peso de conexão correspondente à regra  $k$ ,  $\eta$  é a taxa de aprendizado,  $erro$  é o erro normalizado, e  $P_k(\mu_i, \mu_j)$  é o nível de disparo correspondente à regra  $k$ .

#### **4. SISTEMA EXPERIMENTAL**

A metodologia proposta foi testada em uma máquina de ensaios de fadiga INSTRON Modelo 8501. Ela é constituída de uma bomba hidráulica que fornece uma pressão de 190 bar, um atuador hidráulico cilindro-pistão com capacidade de 100 kN, e é comandada por uma servo-válvula MOOG modelo D562, que tem como entrada de controle um sinal de corrente de  $\pm 40$  mA. Além disso, a máquina possui três sensores: um LVDT, que mede o deslocamento do atuador na faixa de  $\pm 50$  mm a partir de uma posição central, clip gages que medem a deformação do corpo de prova, e uma célula de carga de capacidade de 100 kN tanto na tração como compressão.

O controle Neuro-Fuzzy foi implementado em um sistema computacional CompactRIO cRIO9004 da National Instruments. Este sistema inclui módulos de entradas analógicas (NI9201), saídas analógicas (NI9263), módulo excitador de *strain gage* (NI9237), e um conversor de tensão elétrica para corrente, vide Figura 6.

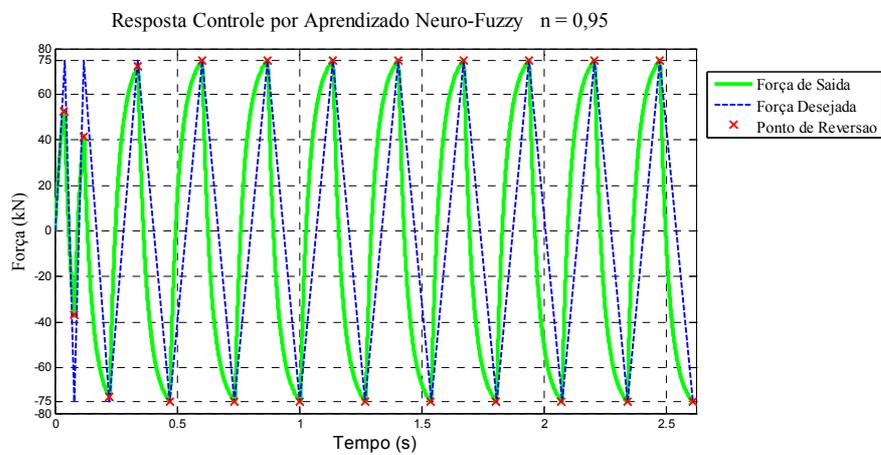
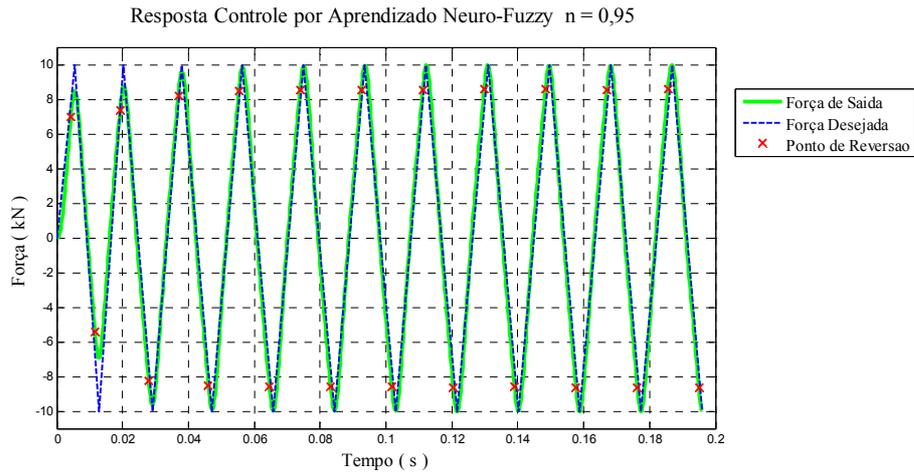


**Figura 6:** Sistema Experimental.

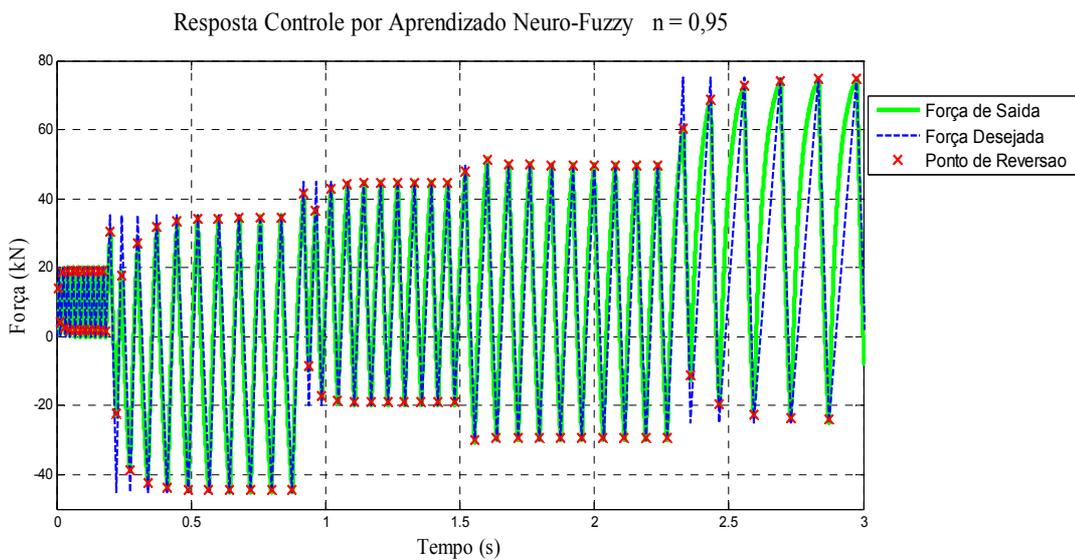
## 5. SIMULAÇÕES

As simulações do sistema de controle proposto aplicado em uma máquina de ensaios servo-hidráulica foram desenvolvidas no *software* MATLAB<sup>TM</sup>. As simulações incluem a modelagem da máquina servo-hidráulica de 100 kN, incluindo um modelo da servo-válvula (5), (6). O modelo do sistema não é apresentado neste trabalho, mas pode ser observado em detalhes em Alva (7).

Nas Figuras 7 e 8 mostram-se as simulações para uma amplitude de carga constante de  $\pm 10\text{kN}$  e  $\pm 75\text{kN}$ , respectivamente, com uma taxa de aprendizado  $\eta = 0,95$ . Os pontos de reversão de pico/vale são modificados a cada novo ciclo de carregamento apresentado, convergindo para um ótimo valor de reversão. Assim, para futuros carregamentos do mesmo valor, o controlador responderá de forma satisfatória sem a necessidade de re-aprendizado.



**Figura 7:** Respostas do controle por aprendizado Neuro-Fuzzy para uma entrada de amplitude constante de  $\pm 10$  kN e  $\pm 75$  kN, respectivamente de cima para baixo.



**Figura 8:** Resposta do controle por aprendizado Neuro-Fuzzy para uma entrada de amplitude variável.

## 6. RESULTADOS EXPERIMENTAIS

Os ensaios são feitos para cargas médias nulas e amplitudes de força de 10kN, 20kN, 30kN e 40kN, todos eles utilizando uma corrente entre  $\pm 20\text{mA}$  e  $\pm 30\text{mA}$  na servo-válvula. Os testes são feitos usando corpos de prova  $\epsilon\text{N}$  com 12mm de diâmetro.

A Figura 9 compara o desempenho do controle proposto e o controle tradicional da Instron. O controle por aprendizado Neuro-Fuzzy é implementado usando correntes entre  $\pm 20\text{mA}$  e  $\pm 30\text{mA}$ , enquanto que o controle tradicional da Instron usa  $\pm 40\text{mA}$  e correntes superiores (modo *overdrive*). É possível observar o melhor desempenho do controle por aprendizado Neuro-Fuzzy para baixas amplitudes e um desempenho igual ou similar para altas amplitudes, mesmo utilizando correntes bem menores. Espera-se que com a mesma corrente de  $\pm 40\text{mA}$  se obtenham resultados ainda melhores com o controle proposto.

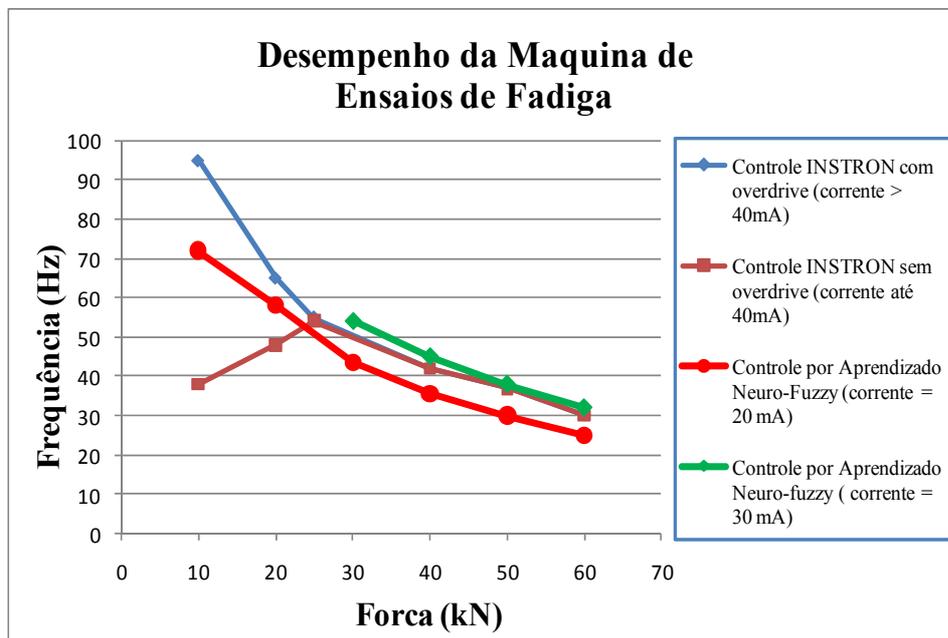


Figura 9: Desempenho do sistema Neuro-Fuzzy numa máquina de ensaio de fadiga.

## 7. CONCLUSÕES

Neste trabalho, mostra-se que é possível aumentar a frequência de trabalho de uma máquina de ensaios de fadiga usando controle por aprendizado misturado com técnicas Neuro-Fuzzy aplicadas a sistemas servo-hidráulicos. O controle proposto foi simulado e posteriormente implementado em uma máquina de ensaios de fadiga, através de um sistema computacional CompactRIO. Os resultados mostram que o controle proposto é capaz de gerar frequências maiores de trabalho que as obtidas pelo controlador original, mesmo utilizando correntes mais baixas para operar as servo-válvulas.

## ***REFERÊNCIAS BIBLIOGRÁFICAS***

- (1) Lewis, F. L.; Campos, J. e Selmic, R. “Neuro-Fuzzy Control of industrial System with Actuator Nonlinearities”. Philadelphia, 2002.
- (2) Branco, P. J.; Dente, J. A. “Design of an Electro-Hydraulic System Using Neuro-Fuzzy Techniques”. Mechatronics Laboratory. Department of Electrical and Computer Engineering. Instituto superior técnico, Lisboa. Portugal – 1998. pp. 190-206.
- (3) Kaitwanidvilai, S.; Parnishkun, M. “Force Control in a Pneumatic System using Hibrid Adaptive Neuro-Fuzzy Model Reference Control”. School of Advanced Technologies, Asian Institute of Technology. Thailand – 2004. pp. 23-41.
- (4) King-Sun F., “Learning Control System,” IEEE Transactions on Automatic Control, pp. 210-221, 1970.
- (5) Viersma T.J., Analysis, Synthesis and Design of Hydraulic Servosystems and Pipelines. Elsevier, 1980.
- (6) Thayer W.J., “Transfer Functions for MOOG Servovalves,” Technical Bulletin. New York, NY, USA, 1965.
- (7) Alva J.G.C., “Controle por Aprendizizado de Sistemas Servo-Hidráulicos de Alta Frequência,”M.Sc. thesis, Pontifical Catholic University of Rio de Janeiro, in Portuguese, 2008.