



## Algoritmos para Auto-Ajuste de Sensores Utilizando Redes Neurais

Marcos A. A. de Oliveira<sup>1</sup>, Jorge L. M. do Amaral<sup>2</sup>, José F. M. do Amaral<sup>2</sup>

<sup>1</sup> UERJ / Laboratórios B.Braun S.A., Rio de Janeiro, Brasil, [marcos.oliveira@bbraun.com](mailto:marcos.oliveira@bbraun.com)

<sup>2</sup> UERJ, Rio de Janeiro, Brasil, [jamaral@uerj.br](mailto:jamaral@uerj.br)

<sup>3</sup> UERJ, Rio de Janeiro, Brasil, [franco@uerj.br](mailto:franco@uerj.br)

**Resumo:** Este trabalho propõe o desenvolvimento de um algoritmo para auto-ajuste de sensores inteligentes a partir do uso de Redes Neurais Artificiais. Através de dados gerados em uma calibração, o algoritmo aprende a corrigir os desvios encontrados. A título de exemplo foi utilizado um sensor do tipo Pt-100. O desafio é fazer uma rede neural aprender uma curva com poucos pontos, como os gerados em uma calibração tradicional [1].

**Palavras chave:** Rede Neural, Calibração, Ajuste, Auto-ajuste, sensores inteligentes.

### 1. INTRODUÇÃO

O avanço da micro-eletrônica permitiu o desenvolvimento de sensores inteligentes, isto é, sensores com microcontroladores ou microprocessadores embarcados dotados de *softwares* específicos que são capazes de executar uma grande variedade de funções. Uma das funções mais úteis que pode ser agregada a um sensor inteligente é a capacidade de se auto-ajustar, ou seja, de minimizar os erros sistemáticos de medição sem ajuda de nenhum dispositivo externo. O ajuste (de um instrumento de medição) é a operação destinada a fazer com que um instrumento de medição tenha desempenho compatível com o seu uso [2]. O ajuste pode ser: automático, semi-automático ou manual [2]. O termo auto-ajuste também é conhecido como auto-calibração, ou ainda em Inglês *Auto-Calibration* ou *Self-Calibration*. Todavia, o termo auto-calibração está em desacordo com o VIM (Vocabulário Internacional de Metrologia). Nos instrumentos que a sua curva de calibração não é linear, e/ou sua deriva é elevada em relação ao uso pretendido, ou ainda, quando o número de grandezas e faixas de indicação são muito elevados, o ajuste pode gerar um custo indesejável. Logo, a capacidade de auto-ajuste em um sensor é uma característica interessante, pois facilita a manutenção do seu estado metrológico. Isto é particularmente importante, por exemplo, em medidores de pH.

A pesquisa bibliográfica realizada mostrou várias possibilidades de uso de rede neurais em sensores inteligentes como: Predição da deriva (variação dos erros sistemáticos ao longo do tempo) de instrumentos entre calibrações utilizando redes neurais para previsão de séries temporais [11], técnicas de auto-ajuste (*self-calibration*)

[12] [16], detecção de falhas [13], monitoramento e verificação de calibração [14], linearizar as características de um transdutor ou para compensar os distúrbios causados por outras grandezas de influência [15].

Neste trabalho foi utilizado uma rede neural recorrente do tipo *Elman* e um polinômio padronizado por norma para desenvolver um algoritmo para auto-ajuste de sensores inteligentes.

### 2. TERMÔMETROS DE RESISTÊNCIA

Os termômetros de resistência (termoresistências) são considerados sensores de alta exatidão e boa repetibilidade de leitura, também são conhecidos como RTD que é abreviação inglesa de "*Resistance Temperature Detector*". Vários metais podem ser utilizados, como:

- Cobre (Cu), faixa: -200 a 260 °C;
- Molibdênio (Mo), faixa: -200 a 200 °C;
- Níquel (Ni), faixa: -80 a 260 °C;
- Níquel-ferro (Ni-Fe), faixa: -200 a 200 °C;
- Platina (Pt), faixa: -200 a 650 °C.

O elemento sensor mais utilizado é a platina, preferencialmente com o mais alto grau de pureza e encapsulados em bulbos de cerâmica ou vidro [3]. O primeiro a usar o termômetro de platina foi *William Siemens* em 1871. No entanto, a termometria de precisão com platina só foi iniciada com a publicação do trabalho de *Hugh Callendar* em 1887.

As termoresistências de platina também são conhecidas como PRT que é abreviação inglesa de "*Platinum Resistance Thermometers*", ou ainda TRP que seria a abreviação de Termômetro de Resistência de Platina.

Atualmente a termoresistência de Platina mais utilizada na indústria é do tipo Pt-100, cujo sufixo numérico corresponde a sua resistência elétrica quando submetida a uma temperatura de 0°C, mas existem outros tipos:

- Pt-0,25;
- Pt-2,5;
- Pt-25;
- Pt-200;
- Pt-1000.

Outro aspecto importante das termoresistências é o seu coeficiente de variação térmica,  $\alpha$  ( $\alpha$ ), que é definido pela equação 1:

$$\alpha = \frac{R_{100} - R_0}{100 \cdot R_0} \quad (1)$$

Onde:

$R_{100}$  = resistência na temperatura de 100 °C;  
 $R_0$  = resistência na temperatura de 0°C.

Os valores de  $\alpha$  mais comuns são:

- 0,00385 °C<sup>-1</sup>;
- 0,00375 °C<sup>-1</sup>;
- 0,00392 °C<sup>-1</sup>.

Os valores especificados de temperatura estão definidos na ITS-90 (“*International Temperature Scale of 1990*”) [4], escala internacional de temperatura do ano de 1990, ou ainda a EIT-90. E segundo a ASTM (“*American Society Testing and Materials*”) E 1137 – 97, para as variações de temperatura a serem medidas são válidas as equações (2) e (3), conhecidas como *Callendar-Van Dusen*, que seria a união da equação original de *Callendar* para a faixa positiva de temperatura (2), mais a contribuição posterior de *Van Dusen* para a faixa negativa (3):

De 0 a 650 °C;

$$R_t = R_0 [1 + A \cdot t + B \cdot t^2] \quad \Omega \quad (2)$$

De -200 a 0 °C;

$$R_t = R_0 [1 + A \cdot t + B \cdot t^2 + C \cdot (t - 100) \cdot t^3] \quad \Omega \quad (3)$$

Onde:

$t$  = temperatura (ITS-90) °C;  
 $R_t$  = resistência na temperatura ( $t$ );  
 $R_0$  = resistência à 0°C;  
 Valores típicos para  $\alpha = 0,00385$  °C<sup>-1</sup> [5]  
 $A = 3,9083 \times 10^{-3}$  °C<sup>-1</sup>;  
 $B = -5,775 \times 10^{-7}$  °C<sup>-2</sup>;  
 $C = -4,183 \times 10^{-12}$  °C<sup>-4</sup>;

A equação *Callendar-Van Dusen* e os seus coeficientes padronizados para cada tipo de Pt-100 é utilizada para gerar as tabelas padrão dos sensores. Os instrumentos que são utilizados para obter as leituras das termoresistências geralmente indicam em ohm, porém a grandeza de interesse é a temperatura e normalmente a unidade é grau Celsius. Desta maneira, necessitamos de equações onde o termo independente seja resistência elétrica e o termo dependente seja temperatura, ou seja, a partir de uma dada resistência, seja possível determinar a temperatura do sensor.

A equação de *Van Dusen* (utilizada na faixa negativa) não permite facilmente a sua dedução inversa, por esse motivo à norma ASTM E 1137, sugere as seguintes equações inversas, para a parte negativa (4) e para a parte positiva (5):

De -200 a 0 °C;

$$t = \sum_{i=0}^4 D_i \left( \frac{R_t}{R_0} - 1 \right)^i \quad (4)$$

De 0 a 650 °C;

$$t = \frac{\sqrt{A^2 - 4B(1 - \frac{R_t}{R_0})} - A}{2B} \quad (5)$$

Onde:

$t$  = temperatura (ITS-90) °C;  
 $R_t$  = resistência na temperatura ( $t$ );  
 $R_0$  = resistência à 0°C;  
 Valores típicos para  $\alpha = 0,00385$  °C<sup>-1</sup> [5]  
 $A = 3,9083 \times 10^{-3}$  °C<sup>-1</sup>;  
 $B = -5,775 \times 10^{-7}$  °C<sup>-2</sup>;  
 $D_1 = 255,819$  °C;  
 $D_2 = 9,14550$  °C;  
 $D_3 = -2,92363$  °C;  
 $D_4 = 1,79090$  °C;

A equação (5) é uma aproximação da *Van Dusen*, por isso um erro de até 0,002 °C é esperado [5].

### 3. REDES NEURAIS ARTIFICIAIS

As principais características das Redes Neurais Artificiais (RNAs) são sua habilidade de aprender certo conteúdo e a capacidade de generalizar, adaptando-se a situações que jamais fizeram parte do conjunto de treinamento.

Enquanto sistemas de softwares convencionais são programados através de um grande número de instruções e regras, a rede neural absorve o conhecimento através da apresentação de exemplos básicos em seu treinamento [6].

Uma rede neural consiste em uma ou várias unidades processadoras básicas interligadas dispostas em uma ou mais camadas. Cada camada pode conter diversos elementos processadores. A conexão entre as unidades processadoras é feita através do peso sináptico, que define a influência entre as unidades processadoras interligadas [6]. Os pesos sinápticos, conhecidos como parâmetros internos, têm seu valor atualizado durante o processo de treinamento. Cada unidade processadora executa uma operação simples sobre a informação recebida da entrada ou de outro processador [9].

O resultado é então transmitido para outros processadores que, por sua vez, repetem a operação. O processo de atualização dinâmica dos pesos é feito de forma paralela e distribuída por toda a rede.

Quando as saídas diferem dos alvos por um valor inferior à tolerância de erro pré-estabelecida, está concluído o processo de aprendizagem. Neste ponto, pode-se dizer que a rede está devidamente treinada e a matriz de pesos, que será usada na fase de execução, retém o conhecimento que a rede foi capaz de absorver durante a fase de aprendizado [8].

A Figura 1 mostra a arquitetura de uma rede neural genérica com uma camada, onde  $P_1, P_2, P_3, \dots, P_R$  representam os elementos do vetor de treinamento  $P$  da rede [7]. O bloco somatório realiza a soma ponderada das entradas  $P_1, P_2, P_3, \dots, P_R$  com os pesos  $W_{ij}$ , normalmente conhecida, sob a forma matricial, como  $NET = W \cdot P$

A saída  $a = F(\text{NET})$  é o resultado da aplicação da soma ponderada NET na função de ativação F, que pode ser linear ou não [6].

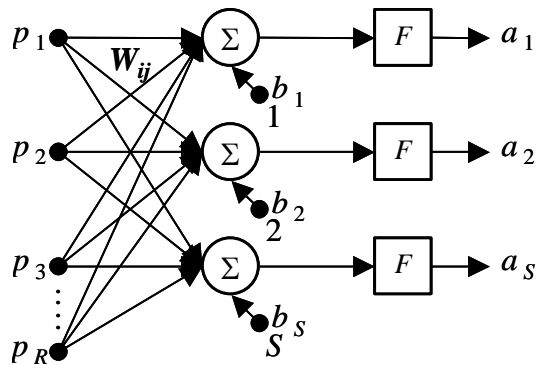


Figura 1 - Arquitetura Genérica de uma Rede Neural Artificial.

A regra de *Widrow-Hoff* [9] é a base fundamental dos diversos métodos de treinamento das RNAs. Ela avalia o erro médio quadrático a cada iteração através da derivada parcial do erro quadrático em relação ao peso W e ao limiar (bias) b.

Terminado o treinamento, a rede está pronta para ser executada. Nesta fase são fornecidos à RNA novos padrões, denominados padrões de teste, e a rede responde, rapidamente, de acordo com o que lhe foi imposto durante o treinamento.

As MLP (*Multi Layer Perceptron*) são redes em que os neurônios são dispostos em múltiplas camadas, conforme pode ser visto na figura 2.

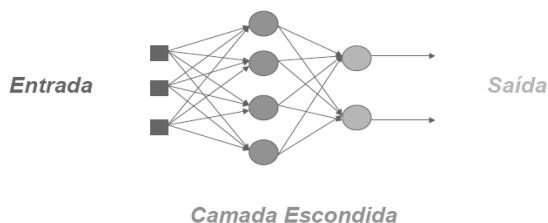


Figura 2 – Rede MLP

O algoritmo de treinamento das MLPs é chamado de *backpropagation*. Conforme pode ser visto na figura 3, ele é composto de duas etapas: na primeira, chamada de passo direto, a rede é ativada por um exemplo e o erro de cada neurônio da camada de saída é calculado. Na segunda etapa, chamada de passo reverso (*backward*), o erro da rede é usado para atualizar os pesos. Os neurônios da camada oculta não estão relacionados ao erro de forma direta, mas através de neurônios da próxima camada. A partir da camada de saída, o erro é retropropagado (*backpropagation*) através da rede de camada a camada.

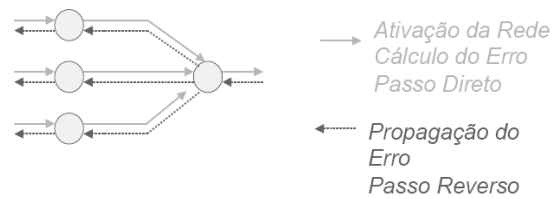


Figura 3 - *backpropagation*

O *backpropagation* é um dos algoritmos de treinamento de Redes Neurais mais utilizado em aplicações de previsão, de classificação e de reconhecimento de padrões em geral. Toda função limitada contínua por partes pode ser aproximada com um erro arbitrariamente pequeno por uma rede de camada escondida.

Uma importante característica das Redes Neurais é a sua capacidade de generalizar o seu conhecimento a partir de exemplos de treinamento. Isto permite que a rede seja capaz de lidar com ruídos e distorções, e também responder corretamente a novos padrões, isto é, padrões que não foram apresentados no treinamento.

Para que a rede tenha uma boa capacidade de generalização, utiliza-se o treinamento com validação cruzada (*cross validation*), evitando o super treinamento (*overfitting*), que diminui a capacidade de generalização.

Para se utilizar essa técnica, os padrões devem ser divididos em dois conjuntos:

Treinamento: padrões utilizados para modificar os pesos;

Teste: padrões utilizados para testar o desempenho do modelo final.

A seguir, a rede é treinada com os padrões de treinamento, observando o erro obtido pela rede quando os padrões de teste são apresentados.

### 3.1 Redes Recorrentes

As redes recorrentes possuem laços de realimentação, ou seja, a saída de um neurônio pode ser uma entrada para outro de uma camada precedente ou, no caso de auto-realimentação, para o próprio neurônio. As redes recorrentes, chamadas de redes com memória, não possuem organização rígida e seus neurônios possuem liberdade para se ligar a qualquer outro neurônio [9]. A rede de *Elman* é uma estrutura de rede neural com características recorrentes [17], isto é, pode apresentar uma ou mais malhas de realimentação, fazendo com que suas conexões permitam à rede memorizar as informações atuais na camada de contexto no momento seguinte à sequência investigada, ver figura 4.

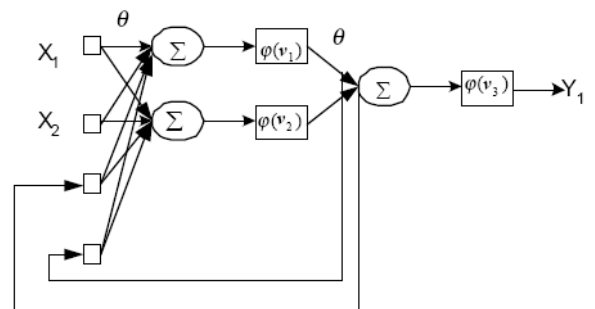


Figura 4 – Rede Elman

A Rede Elman é empregado tipicamente nas situações onde sequência dos dados de entrada da rede é importante, isto é, quando se deseja que a rede, de algum modo, utilize as entradas precedentes, levando-as em consideração juntamente com a entrada atual para gerar uma resposta. Isto permite que a rede execute a aprendizagem das tarefas que estendem sobre o tempo. De fato, é devido à natureza da realimentação em torno das unidades escondidas, estes neurônios escondidos continuam reciclando a informação através da rede sobre as etapas de tempo múltiplas, e descobrem deste modo a representação interna abstrata do tempo, ou ainda no nosso caso a sequência dos pontos de calibração. Esta característica é muito interessante para que a curva de correção da calibração seja uniforme, assumindo valores fisicamente coerentes entre os pontos de calibração.

### 3.2 Leave-One-Out

Foi utilizado o algoritmo de treinamento *Leave-One-Out* que é um caso especial de validação cruzada. O que é computacionalmente dispendioso e frequentemente é usado em amostras pequenas. Para uma amostra de tamanho  $n$  ( $n > 2$ ), uma hipótese é induzida utilizando  $(n-1)$  exemplos; a hipótese é então testada no único exemplo remanescente. Esse processo é repetido  $n$  vezes, cada vez induzindo uma hipótese deixando de considerar um único exemplo[10].

## 4. ALGORITMOS PARA AUTO-AJUSTE

Para esse trabalho foi escolhida uma topologia utilizando os polinômios para termoresistências e um rede neural *Elman*. Nesta topologia, durante a fase de treinamento, os erros que os pontos de calibração possuem em relação ao polinômio padronizado, são apresentados para o treinamento da rede neural. Durante a fase de utilização o valor de entrada é apresentado ao polinômio e à rede neural simultaneamente e a saída do sistema é o somatório da saída de ambas, conforme figura 5.

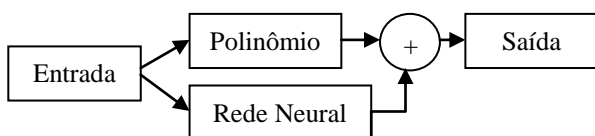


Figura 5 – Topologia do Algoritmo

A partir do *Leave-One-Out* a nossa rede foi treinada cinco (número de pontos disponíveis) vezes, deixando a cada treino um ponto diferente fora do treinamento, para testar o erro da rede. A rede que apresentou um menor erro em relação ao ponto de teste é automaticamente selecionada pelo algoritmo, ver figura 7.

Com o objetivo de aumentar a repetibilidade do sistema este procedimento foi repetido  $n$  vezes (tentativas) e o melhor resultado, em relação ao maior erro, em módulo, foi guardado, ver figura 6.

A fim de minimizar os erros, o próximo passo foi refinar a rede escolhida, treinando a mesma por  $x$  épocas, salvando este treinamento caso apresente erros menores que o anterior. Repetindo este procedimento por  $y$  vezes (refino) e ficando com o melhor resultado, como pode ser visto na figura 8.

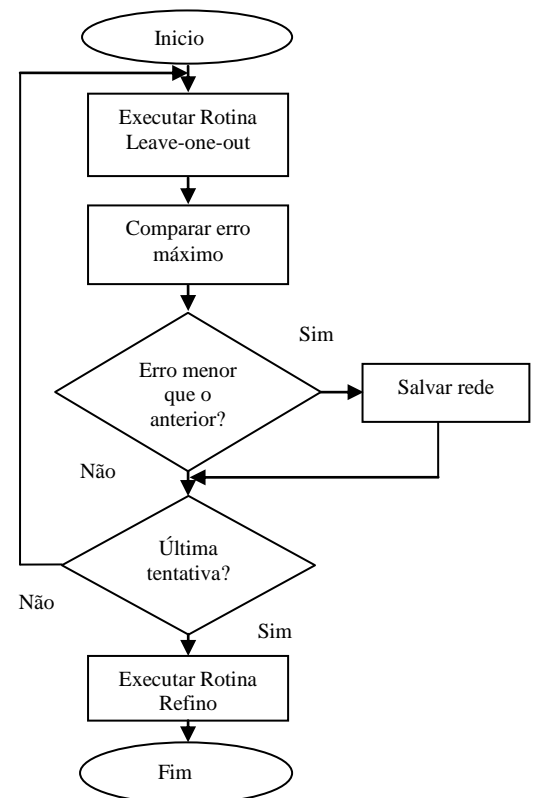


Figura 6 – Fluxograma Geral

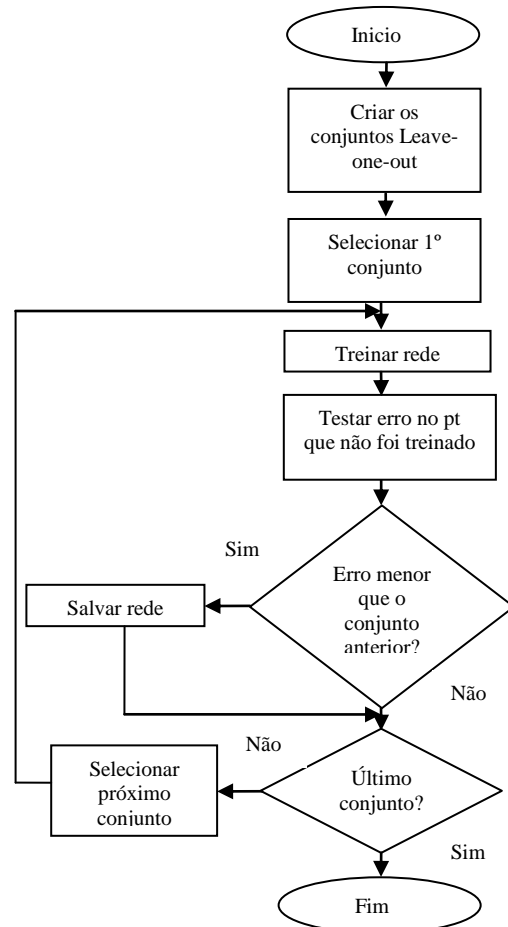


Figura 7 – Fluxograma *Leave-one-out*

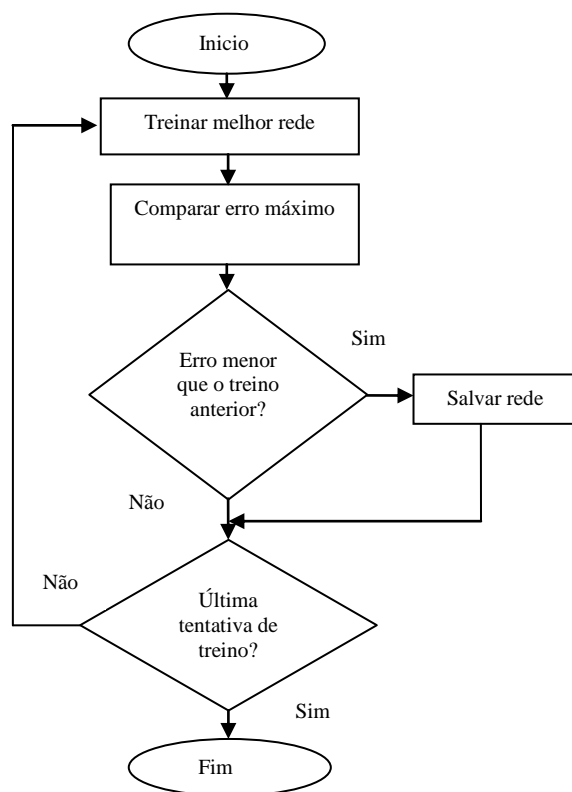


Figura 8 – Fluxograma de Refino

## 5. EXPERIMENTOS

Neste trabalho, os pontos da curva resistência x temperatura, foram retirados de uma calibração feita em um laboratório da RBC (Rede Brasileira de Calibração) em um Pt-100 padrão (Isothech T100) com uma incerteza máxima na faixa calibrada de 0,11°C, calibração em 05 pontos na faixa de -20 a 300 °C.

Para o projeto da rede *Elman* e o treinamento, foram utilizados os dados da tabela 1:

Tabela 1. Parâmetros da Rede

Parâmetro	Valor
Taxa de aprendizado	0,1
Número máximo de épocas de aprendizado (Leave-one-out)	1000
Número máximo de épocas de aprendizado (Refino)	20 x 500 (20 épocas com 500 testes intermediários)
Objetivo de erro (máximo absoluto)	0
Números de neurônios na camada de saída	1
Função de ativação da camada escondida	Tansig (Sigmóide Tangente Hiperbólica)
Função de ativação da camada de saída	Purelin (linear)
Método de treinamento	Traingdx (Gradiente de descida com momento e adaptativo)
Números neurônios na camada escondida	11

Outro grande desafio para esse algoritmo foi torná-lo repetitivo, desta forma, melhorar o seu grau de concordância entre os resultados de edições sucessivas de um mesmo mensurando efetuadas sob as mesmas condições de medição [2]. Os ajustes de tentativas (repetições do *leave-one-out*), refino (repetições da rotina de refino) e épocas (quantidade de épocas de treinamento em cada refino), que foram mais decisivos na evolução da repetitividade e redução do erro máximo, podem ser vistos na figura 9. Sendo então escolhido os valores de 20 tentativas de *leave-one-out* com 500 repetições de refino e 20 épocas de aprendizagem para cada refino. Com este ajuste conseguimos com 5 repetições, uma média de erros máximos de 0,0024 °C com um desvio padrão de 0,0007 °C.

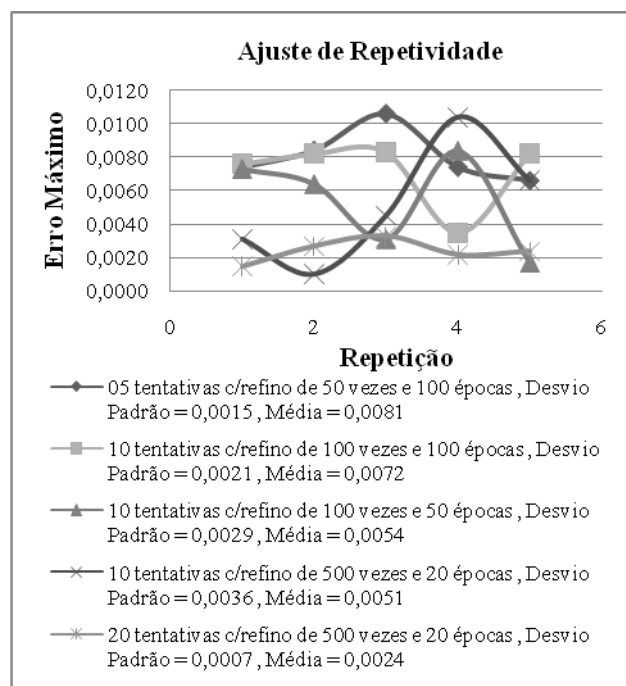


Figura 9 – Repetitividade

Após execução do programa no *Matlab®*, obtivemos os resultados apresentados nas figuras 10 e 11. Na figura 10 podemos ver o erro em relação polinômio, esse erro corrigido pela rede neural (0,0026°C), e a correção final com o refinamento da rede (0,0022 °C). Na figura 11 podemos ver a curva de correção da rede e da rede refinada em toda a faixa ajustada, com intervalo de simulação de 1 ohm, a fim de comprovar a ausência de *over-fitting*.

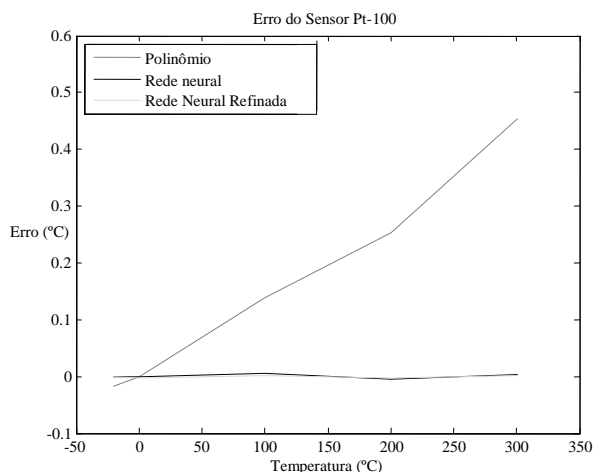


Figura 10 – Erros do Sistema

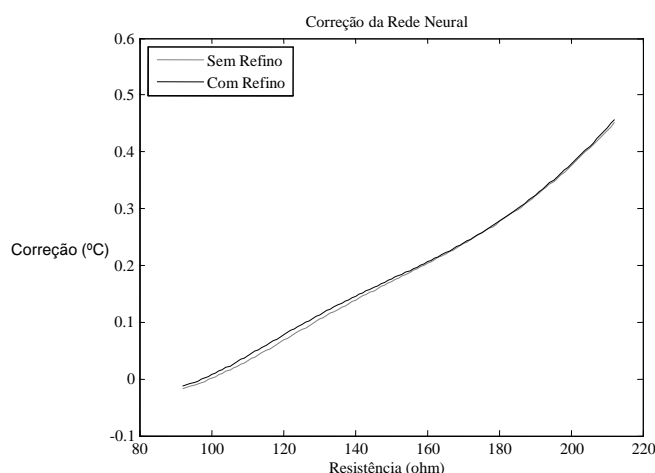


Figura 11 – Correção da Rede Neural

Observando os resultados obtidos nas simulações no *Matlab*®, vimos que essa rede teve um excelente comportamento, convergindo para um resultado muito próximo da curva apresentada no aprendizado. Na figura 12 e na tabela 2, podemos ver um teste de reprodutibilidade a partir do histórico das calibrações do nosso sensor, ao mesmo tempo comparando-se com os ajustes obtidos com *software* comercial para *fit-curve* muito utilizado em metrologia o *TableCurve*®. De acordo com a tabela podemos notar que a nossa rede neural se comporta melhor que o método tradicional principalmente quando temos poucos pontos de calibração.

Reprodutibilidade dos resultados de medição é o grau de concordância entre os resultados das medições de um mesmo mensurando efetuadas sob condições variadas de medição[2].

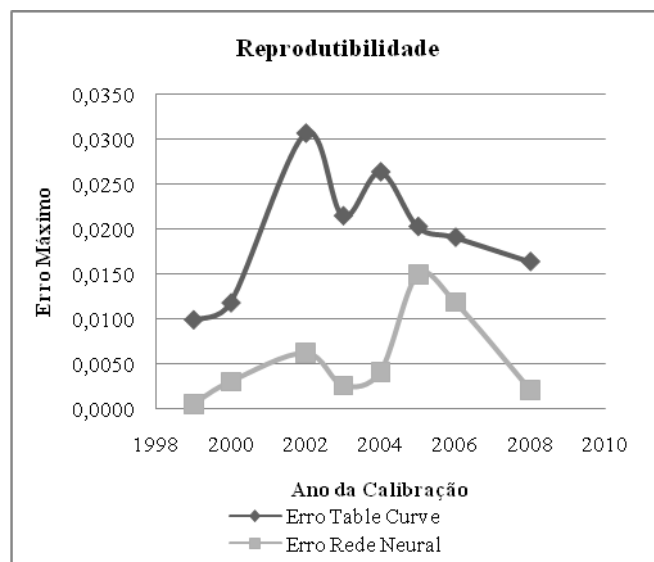


Figura 12 – Reprodutibilidade

Tabela 2. Reprodutibilidade

Levantamento de Reprodutibilidade			
Ano da Calibração	Erro Table Curve (°C)	Erro Rede Neural (°C)	Pontos da Calibração
2008	0,0164	0,0022	5
2006	0,0191	0,0119	7
2005	0,0203	0,0150	7
2004	0,0264	0,0042	7
2003	0,0215	0,0027	7
2002	0,0307	0,0063	7
2000	0,0118	0,0031	5
1999	0,0099	0,0006	7

## 6. CONCLUSÃO

A vantagem que este método pode apresentar em relação ao método estatístico tradicional, é a facilidade de implementação deste algoritmo em sistemas embarcados como: transmissores e termômetros, criando instrumentos auto-ajustáveis. Somando-se a isto, também a capacidade de absorver desvios de modelagem dos polinômios utilizados, devido à capacidade de generalização das redes neurais artificiais.

O próximo passo para aperfeiçoamento deste sistema seria: Implementar essa técnica em um microcontrolador e ou com um *software* com recursos específicos para instrumentação e controle como o *LabView*® e comparar os resultados com aqueles obtidos no *Matlab*®, ou ainda desafiar o sistema com um número menor de pontos (3 ou 4).

## REFERÊNCIAS

- [1] OLIVEIRA, M. A. A. ; AMARAL, J. L. M. ; AMARAL, J. F. M. ; SOARES, D.R.M ; BIONDI NETO, L. . Utilização de uma Rede Neural Artificial para a Interpolação de Valores de Medição de Termoresistências (RTD). Anais do Primeiro Congresso Internacional de Metrologia Mecânica, Rio de Janeiro, 2008.
- [2] INMETRO, Vocabulário Internacional de Termos Fundamentais e Gerais de Metrologia, 4a edição, Rio de Janeiro, 2005.
- [3] NICHOLAS, J. V. and WHITE, D.R., Traceable Temperatures, JOHN WILEY & SONS, 1997
- [4] PRESTON-THOMAS, H., The International Temperature Scale of 1990 (ITS-90), Metrologia, Vol 27, No. 1, 1990, pp 3-10
- [5] ASTM E 1137-97, Standard Specification for Industrial Platinum Resistance Thermometers, 1997
- [6] BIONDI, L. N.;PACHECO, M. A. C.; VELLASCO, M. M. B. R.; PASSOS, E. P. L. e CHIGANER, L.. Sistema Híbrido de Apoio à Decisão para Detecção e Diagnóstico de Falhas em Redes Elétricas. Anais do III Simpósio Brasileiro de Redes Neurais, Recife, p. 197-204, 1996.
- [7] COELHO, P. H. G.. A Complex EKF - RTRL Neural Networks. Proceedings of the IJCNN'2001, New Jersey, IEEE Press, v.1, p. 120-125, 2001.
- [8] DAYHOFF, Judith E.. Neural Network Architectures: An Introduction. New York: Van Nostrand Reinhold, 1990.
- [9] HAYKIN, Simon. Neural Networks a Comprehensive Foundation, Englewood Cliffs: Macmillan College Publishing Company, 1994.
- [10] MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre Aprendizado de Máquina. Capítulo 4. In: REZENDE, S. O. Sistemas Inteligentes: Fundamentos e Aplicações. Barueri: Manole, 2002.
- [11] GOLOVKO, V.; SAVITSKY, J.; SACHENKO, A.; KOCHAN, V.; TURCHENKO, V.; LAOPOULOS, T.; GRANDINETTI, L.. Intelligent System for Prediction of Sensor Drift. International Conference on Neural Networks and Artificial Intelligence ICNNAI'99, 1999.
- [12] PATRA, J. C.; GOPALKRISHNAN, V.; ANG, E. L.; DAS, A.. Neural Network-Based Self-Calibration/ Compensation of Sensors Operating in Harsh Environments. Sensors, 2004. Proceedings of IEEE Volume , Issue , 24-27 Oct. 2004 Page(s): 425 - 428 vol.1.
- [13] NAJAFI, M.; CULP, C.; LANGARI, R.. Performance Study of Enhanced Auto-Associative Neural Networks For Sensor Fault Detection. Energy Systems Laboratory, 2004
- [14] PINTO, J. O. P.; PINTO, A. M. A. C; TATIBANA, G. S.; SANCHES, R. O.; CARRASCO, B. N.; FLORA, N.; HINES , J. W.. Uma Nova Estratégia de Modelagem para Monitoramento de Instrumentos e Verificação de Calibração Usando Redes Neurais Artificiais. Anais do 3º Congresso Brasileiro de P&D em Petróleo e Gás., 2005
- [15] PEREIRA, J.M. D.; POSTOLACHE, O.; GIRÃO, P. S.. Using a Dual-Channel FDC Device and ANN Techniques to Improve Measurements Accuracy. Sensors & Transducers Magazine (S&T e-Digest), Vol.62, Issue 12, December 2005, pp.462-472
- [16] RIVERA, J.; CARRILLO, M.; CHACÓN, M.; HERRERA, G.; BOJORQUEZ, G.. Self-Calibration and Optimal Response in Intelligent Sensors Design Based on Artificial Neural Networks. sensors ISSN 1424-8220, 7, 1509-1529, 2007
- [17] ELMAN, J. L.. Finding Structure in Time. Cognitive Science, 14, 179-211. 1990.