

UTILIZAÇÃO DE UMA REDE NEURAL ARTIFICIAL PARA INTERPOLAÇÃO DE VALORES DE MEDIÇÃO DE TERMORESISTÊNCIAS (RTD)

Marcos A. A. de Oliveira¹, David R. M. Soares², Luiz Biondi Neto³, Jorge L. M. do Amaral⁴, José F. M. do Amaral⁵

¹ UERJ / Laboratórios B.Braun S.A., Rio de Janeiro, Brasil, marcos.oliveira@bbraun.com

² UERJ, Rio de Janeiro, Brasil, davidrmsoares@gmail.com

³ UERJ, Rio de Janeiro, Brasil, luizbiondi@terra.com.br

⁴ UERJ, Rio de Janeiro, Brasil, jamaral@uerj.br

⁵ UERJ, Rio de Janeiro, Brasil, franco@uerj.br

Resumo: Este trabalho propõe o uso de Redes Neurais para gerar a curva de ajuste de termoresistências de Platina. Os resultados obtidos são satisfatórios apresentando um erro máximo de 0,01 Ohms. 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.

Palavras chave: Termoresistências, Rede Neural, Calibração, Ajuste.

1. INTRODUÇÃO

Um dos problemas clássicos encontrados na área de metrologia é o mapeamento entre o valor indicado pelo elemento sensor e o valor real da grandeza medida (V.V.C. ou seja valor verdadeiro convencional). Um destes elementos é a termoresistência. Os “termômetros de resistência” funcionam baseados no fato de que a resistência elétrica de uma grande gama de materiais varia com a temperatura. De um modo geral, os metais aumentam a resistência com a temperatura [1], sempre de forma não linear, ao passo que os semicondutores diminuem a resistência com a temperatura, conforme mostrado na figura 1.

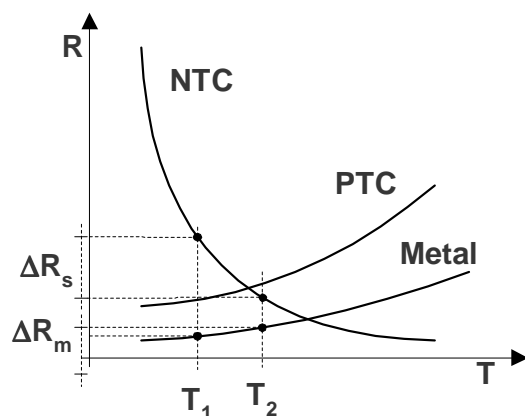


Figura 1

Os termômetros de resistência são considerados sensores de alta exatidão e boa repetibilidade de leitura. Quando metais são usados, o elemento sensor é normalmente confeccionado de platina com o mais alto grau de pureza e encapsulados em bulbos de cerâmica ou vidro [2]. Até mesmo no caso da platina a sua não linearidade deve ser considerada, os erros seriam de até 3,5 °C na faixa de 0 a 300 °C, para um PT100 com um alpha de 0,385 Ohms, ou seja 0,385 Ohms por graus Celsius a 100 °C, ver figura 2.

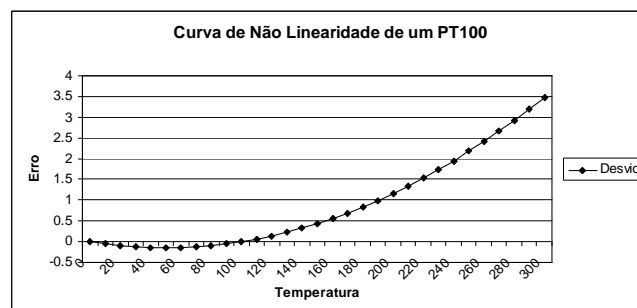


Figura 2

Atualmente, as termoresistências de Platina mais usadas são:

- Pt-25,5Ω
- Pt-100Ω
- Pt-200Ω,
- Pt-1000Ω

A termoresistência mais conhecida e utilizada na indústria é o Pt-100Ω (a 0°C). Sua faixa de uso vai de -200 a 650°C, conforme a norma ASTM E1137 e a norma DIN IEC 751 padronizou sua faixa de -200 a 850°C, sendo que ambas as normas são aceitas.

Normalmente, o bulbo de resistência é montado em uma bainha de aço inox, totalmente preenchida com óxido de magnésio, de tal maneira que haja uma ótima condução térmica e proteção do bulbo com relação a choques mecânicos. A isolamento elétrica entre o bulbo e a bainha obedece à mesma norma IEC 751.

Para variações de temperatura a serem medidas é válida a equação (1):

$$R_T = R_o [1 + A.T + B.T^2 + C.(t-100).T^3] \Omega \quad (1)$$

Onde:

T = temperatura (ITS-90), °C,

R_T = resistência na temperatura (T),

R_o = resistência a 0°C,

A = 3,9083 x 10⁻³ °C⁻¹,

B = -5,775 x 10⁻⁷ °C⁻² e

C = -4,183 x 10⁻¹² °C⁻⁴.

Existem diversas técnicas para gerar a curva de ajuste de termoresistências, sendo que o método mais utilizado é o estatístico para se encontrar os coeficientes A, B e C reais.

Este trabalho propõe o uso de Redes Neurais para interpolar os valores de uma termoresistência e está dividido da seguinte forma: na seção 2, faz-se uma breve apresentação das Redes Neurais; na seção 3 são apresentados os resultados obtidos nos experimentos e a seção 4, com as conclusões, encerra este trabalho.

2. REDES NEURAIS ARTIFICIAIS

As principais características das RNAs são sua habilidade de aprender certo conhecimento 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 [3].

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 [2]. 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 [6].

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 distribuídamente por toda a rede.

Quando as saídas diferem dos alvos por um valor inferior à tolerância de erro preestabelecida, 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 [5].

A Figura 3 mostra a arquitetura de uma rede neural genérica com uma camada, onde $P_{1,2,3,\dots,R}$ representam os elementos do vetor de treinamento P da rede [4]. O bloco somatório realiza a soma ponderada das entradas $P_{1,2,3,\dots,R}$ com os pesos W_{ij} , normalmente conhecida, sob a forma matricial, como $NET = W \cdot P$

A saída $a = F(NET)$ é o resultado da aplicação da soma ponderada NET na função de ativação F, que pode ser linear ou não [3].

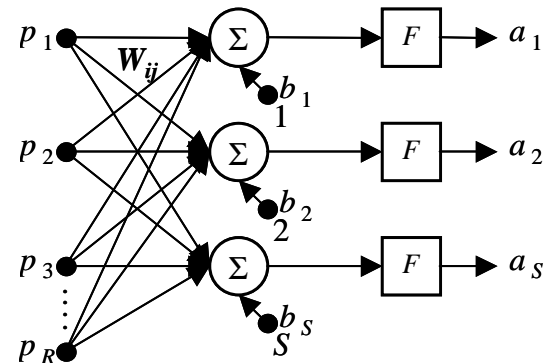


Figura 3 - Arquitetura Genérica de uma Rede Neural Artificial.

A regra de Widrow-Hoff [6] 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 e conseguem representar funções não linearmente separáveis, ver figura 4.

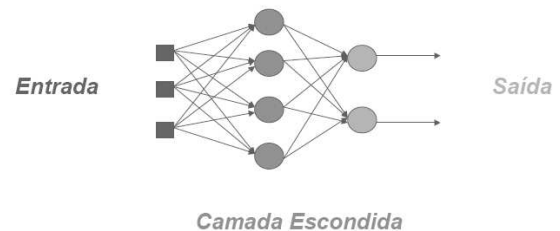


Figura 4

O algoritmo de treinamento das MLPs é chamado de *backpropagation*. Conforme pode ser visto na figura 5, 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 escondida 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. .

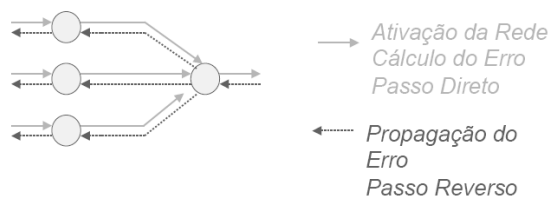


Figura 5

O *backpropagation* é um dos algoritmos de Rede Neural mais utilizado em aplicações práticas 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 padrões novos, 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, evitando o super treinamento (“overfitting”), que diminui a capacidade de generalização.

Para se utilizar esta técnica, devem-se dividir os padrões em três conjuntos, a saber:

- Treinamento: padrões utilizados para modificar os pesos;
- Validação: padrões utilizados para verificar o overfitting;
- Teste: padrões utilizados para testar o desempenho do modelo final.

A seguir, a rede é treinada com os padrões de treinamento, observando-se o erro obtido pela rede quando os padrões de validação são apresentados. O treinamento termina quando o erro para os padrões de validação deixa de diminuir.

3 EXPERIMENTOS

Neste trabalho, os pontos da curva resistência x temperatura, foram retirados da norma ASTM E1137 e foram divididos em conjunto de pontos para treinamento, validação e teste. Foram utilizados 100 padrões de treinamento, 100 padrões de validação e 100 padrões de teste.

Para o projeto da rede e o treinamento, foram utilizados os dados da tabela 1, e a distribuição dos erros de Treinamento, Validação e Teste em função do número de neurônios na camada escondida, pode ser visto na tabela 2:

Parâmetro	Valor
Taxa de aprendizado	0,05
Número máximo de épocas de aprendizado	5000
Objetivo do erro médio quadrático	6×10^{-6}
Números de neurônios na camada de saída	1

Função de ativação da camada escondida	Tansig (sigmóide)
Função de ativação da camada de saída	Purelin (linear)
Método de treinamento e validação	Levenberg-Marquardt

Tabela 1

Números neurônios na camada escondida	Erros		
	Treinamento	Validação	Teste
10	$8,2 \times 10^{-6}$	$1,4 \times 10^{-5}$	$1,2 \times 10^{-5}$
11	$7,2 \times 10^{-6}$	$1,6 \times 10^{-5}$	$1,5 \times 10^{-5}$
12	$7,5 \times 10^{-6}$	$1,2 \times 10^{-5}$	$1,1 \times 10^{-5}$
13	$8,0 \times 10^{-6}$	$1,2 \times 10^{-5}$	$1,3 \times 10^{-5}$
14	$6,0 \times 10^{-6}$	$1,4 \times 10^{-5}$	$1,4 \times 10^{-5}$

Tabela 2

Após a execução do programa no Matlab® com 12 neurônios na camada escondida, obtivemos os resultados apresentados nas figuras 6 e 7. Na figura 6 podemos ver a evolução dos erros de treinamento, validação e teste por épocas de aprendizado (mostra a evolução da rede durante o aprendizado), a figura 7 mostra a evolução do erro da rede neural em toda a faixa de temperatura.

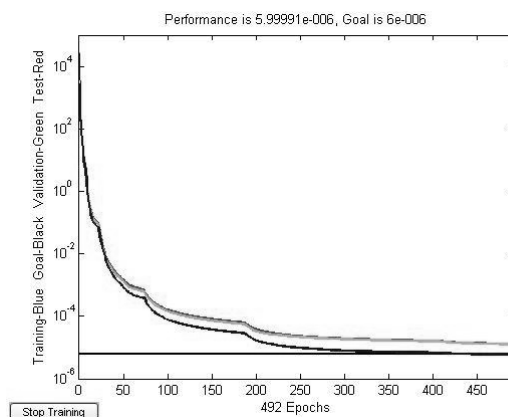


Figura 6

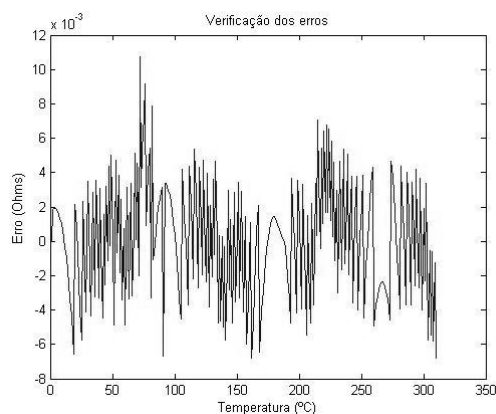


Figura 7

4. CONCLUSÃO

Observando-se os resultados obtidos nas simulações no Matlab®, vemos que esta rede teve um excelente comportamento, convergindo para um resultado muito próximo da curva apresentada no aprendizado, sendo que o maior erro apresentado foi de 0,01 Ohms (0,026 °C para um PT100 de alpha 0,0385). A vantagem que este método pode apresentar em relação ao método tradicional, estatístico, é a facilidade de implementação deste algoritmo em sistemas embarcados, como transmissores e termômetros, criando instrumentos auto-ajustáveis.

Os próximos passos para aperfeiçoamento deste sistema incluem:

- Verificar o desempenho de técnicas de treinamento que utilizem um número reduzido de amostras (pontos de calibração), como por exemplo, *leave-one-out*.
- Investigar a utilização de pontos da função resistência versus temperatura, ao invés da temperatura versus resistência, permitindo assim uma atualização da calibração ao se reprogramar um novo valor de resistência medido em uma temperatura de referência.
- Implementar esta técnica em um microcontrolador e comparar os resultados com aqueles obtidos no Matlab.

REFERÊNCIAS

- [1] HERMINI, Helder Anibal, Apostila MEDIDAS DE TEMPERATURA (Aspectos Gerais)
- [2] NICHOLAS, J. V. and WHITE, D.R., Traceable Temperatures, JOHN WILEY & SONS, 1997
- [3] 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.
- [4] 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.
- [5] DAYHOFF, Judith E.. Neural Network Architectures: An Introduction. New York: Van Nostrand Reinhold, 1990.
- [6] HAYKIN, Simon. Neural Networks a Comprehensive Foundation, Englewood Cliffs: Macmillan College Publishing Company, 1994.