

REDES NEURAIIS ARTIFICIAIS NA ENGENHARIA NUCLEAR (PARTE 1)

Benedito Dias Baptista Filho

Instituto de Pesquisas Energéticas e Nucleares - IPEN-CNEN/SP
Av. Prof. Lineu Prestes, 2242 Cidade Universitária
05508-000, São Paulo, SP, Brasil

RESUMO

Redes Neurais Artificiais (RNA) podem ser definidas como "sistemas paralelos compostos por unidades de processamento simples, dispostas em camadas, altamente interligadas e inspiradas no cérebro humano." Se aplicam a problemas complexos de difícil modelagem, em situações em que os dados são falhos ou incompletos, e em problemas complexos de controle. Ainda é necessária a solução de diversos problemas relacionados a treinamento e capacidade de generalização para utilização segura em sistemas de instalações nucleares. Este trabalho, dividido em duas partes, inicia uma discussão sobre a aplicação de três conceitos de RNA, as redes Alimentadas Adiante em Retro-propagação, os Mapas Auto-Organizáveis, e as Redes de Contatos Sinápticos Múltiplos. São discutidas aplicações em controle, aproximação de funções e reconhecimento de padrões, com exemplos e comentários. Esta parte focaliza as Redes Alimentadas Adiante.

Keywords: neural networks, nuclear engineering, backpropagation.

I. INTRODUÇÃO

O campo das *Redes Neurais Artificiais (RNA)*, um dos ramos da *Inteligência Artificial (IA)*, tem despertado grande interesse na Engenharia Nuclear (EN). As RNA se aplicam a problemas complexos de difícil modelagem, em situações em que os dados são falhos ou incompletos e em problemas complexos de controle. Apesar de algumas limitações, oferecem possibilidades inusitadas em termos de eficiência, robustez e confiabilidade, possuindo capacidade de aprender e se adaptar. Por seu lado, a EN tem seu desenvolvimento sustentado em princípios de segurança que inibem a introdução de avanços não comprovados, por isso muitas dificuldades ainda têm que ser superadas para uma ampla aplicação de RNA na EN.

Existem diversas oportunidades. O desenvolvimento de sistemas de controle em que uma falha simples não cause risco às funções de proteção pode permitir a concentração dos esforços da EN nas áreas onde o desempenho seja o mais importante, reduzindo custos e aumentando a competitividade. Essa é uma das áreas em que muito tem que ser feito para se atingir o desempenho esperado das futuras gerações de reatores nucleares. Uma das pesquisas desenvolvidas no Centro de Engenharia Nuclear do IPEN tem por objetivo o desenvolvimento de RNA para controle de instalações nucleares [1]. Outra aplicação das RNA é a simulação de processos complexos. A simulação, que nada mais é do que a construção de modelos numéricos para previsão do

comportamento, tem muita importância em otimização e no desenvolvimento de sistemas de monitoração e diagnóstico, objetivando instalações mais robustas, com sistemas de controle imunes a falhas de sensores e atuadores. Isso permite utilizar o conceito de "redundância por software", muito mais econômico que a "redundância por hardware." Os avanços nas técnicas de modelagem, na solução de sistemas de equações e no hardware de última geração, ainda não permitem que sejam simulados todos os processos, cada dia mais complexos. Aplicações relacionadas a sistemas de informação e de monitoração e diagnóstico requerem respostas rápidas em situações nas quais sistemas de computação de alta capacidade são inviáveis. O IPEN investe nessa área utilizando RNA nas suas aplicações [2].

Como a simulação, a aproximação de funções é outra aplicação típica de RNA. Nessa linha estão a aproximação de correlações para determinação do fluxo crítico de calor [3], a previsão do tempo de queima do combustível como função das temperaturas no núcleo, etc.

De modo geral, o tema abordado é muito extenso, não sendo viável sua apresentação em um ou dois artigos. O objetivo do trabalho foi limitado à apresentação de três conceitos de RNA em aplicações da EN: as redes em retro-propagação, as redes de contatos sinápticos múltiplos e os mapas auto-organizáveis. O trabalho foi então dividido em duas partes; o objetivo desta primeira parte se limitou a uma aplicação em controle, a simulação de processos e a apro-

ximação de funções com redes alimentadas adiante, sendo desenvolvido com exemplos e comentários

II. REDES NEURAIS ALIMENTADAS ADIANTE

As RNA mais utilizadas atualmente são baseadas em unidades simples, arranjadas em camadas e com os sinais seguindo uma única direção: são as redes *alimentadas adiante* (RAA), ou *redes em "feed-forward"*. As interligações são chamadas de "pesos" e representam sinapses. Se possuem funções de transferência sigmóides, as curvas-S, são chamadas "PERCEPTRONS de Múltiplas Camadas." O método de retro-propagação ("back-propagation") [4], ainda é a melhor opção de treinamento para essas redes. A Fig. 1 ilustra uma RAA.

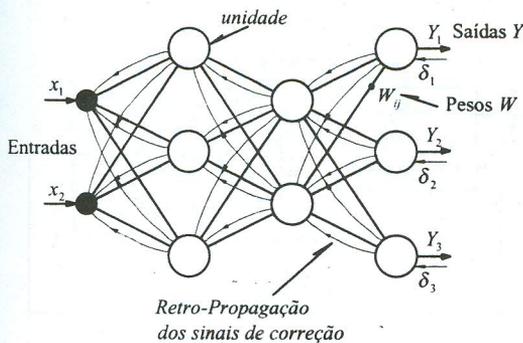


Figura 1. RAA com Retro-Propagação.

O método de retro-propagação tem por objetivo minimizar uma função custo, ou *Função Energia do Erro*, definida como a somatória, sobre todos os padrões, da somatória do quadrado dos erros na saída da rede, ou seja, da diferença entre os valores desejados e obtidos:

$$E(w) = \frac{1}{2} \sum_p \sum_i [y d_i^p - y_i^p]^2 \quad (1)$$

onde w é um peso, p o universo de padrões, i o índice das saídas, yd é o valor desejado, e y é o valor atual na saída.

Diz-se que é um *Método de Gradiente Descendente* pois seu objetivo é ajustar os pesos w no sentido do gradiente negativo de E :

$$\Delta w_{j,i} = -\eta \frac{\partial E}{\partial w_{j,i}} \quad (2)$$

onde η é chamada de *taxa de aprendizado*, com valores entre 0 e 1. A derivada de E na camada de saída é:

$$\frac{\partial E}{\partial w_{j,i}} = -\sum_i [y d_i - y_{j,i}] g'(v_{j,i}) y_{j-1,i} \quad (3)$$

onde $g'(\cdot)$ é a derivada da função de transferência da unidade. Por conveniência, é definido o termo δ :

$$\delta_{j,i} = g'(v_i) [y d_i - y_{j,i}] \quad (4)$$

A função de transferência $g(\cdot)$ geralmente é uma função não-linear, sendo a mais comum a tangente hiperbólica:

$$g(v) = \tanh(\beta v) \quad (5)$$

onde β é uma constante e v a somatória das entradas:

$$v_{j,i} = \sum_k w_{j,i,k} \cdot y_{j-1,k} + w_{0,j,i} \quad (6)$$

onde w_0 é um "bias".

$$g'(v) = \beta [1 - g^2(v)] \quad (7)$$

$$\text{portanto: } \delta_{j,i} = \beta [1 - y_{j,i}^2] [y d_i - y_{j,i}] \quad (8)$$

O treinamento segue os seguintes passos: 1) é feita uma inicialização aleatória de pesos; 2) dado um vetor x de entradas (um exemplo), é realizada a propagação para frente, da primeira à última camada, calculando-se os valores de saída y 's; 3) conhecida a saída calculada e a saída desejada (do exemplo), calcula-se e procede-se à retro-propagação dos δ 's, obtendo-se os incrementos dos pesos w , os Δw . O processo é repetido até a convergência.

O método de retro-propagação apresenta diversos problemas e para minimizá-los existem muitas heurísticas e muitos artificios. Um dos principais artificios, introduzido por Rumelhart et al. [5], é conhecido como "momentum". Como a função de energia do erro pode ser muito complexa, possuindo mínimos locais e platôs, foi idealizada a introdução desse termo para acelerar o processo de convergência e a saída de platôs. O "momentum" considera a variação temporal dos pesos e é adicionado ao termo de correção dos pesos:

$$M = \alpha (w_{ij}(t) - w_{ij}(t-1)) \quad (9)$$

onde α é uma constante que deve ser ajustada caso a caso.

$$\Delta w_{ij}(t) = \eta \delta_j x_i(t) + M \quad (10)$$

Aplicação em Controle. Existem inúmeros trabalhos sobre controladores baseados em RAA. Psaltis et. al. [6] discutem três métodos diferentes. Em geral é necessário o aprendizado do comportamento do sistema a ser controlado. Muitos abordam o uso das RNA para representar a dinâmica inversa do processo a ser controlado. O uso de um método supervisionado implica em que a rede não pode controlar nada antes de ser treinada para isso: *a ação só é possível após o treinamento*. A instalação é utilizada para treinar o *estimador* na dinâmica inversa.

Um exemplo de processo que pode ser controlado por uma RAA é o manipulador bi-articulado no plano da Fig. 3. É um típico problema não-linear com grande grau de

dificuldade pelas singularidades que surgem no problema inverso, que não podem ser resolvidas por métodos analíticos tradicionais. Existem dois objetivos clássicos de controle relacionados a um manipulador: o *controle de posição* e o *controle de trajetória* [7]. O controle de posição consiste no fornecimento de dois sinais para os acionadores dos motores elétricos, uma para cada segmento, com o objetivo de posicionar a extremidade em um ponto desejado. Além da dinâmica dos segmentos, o atrito nas articulações dificulta o controle.

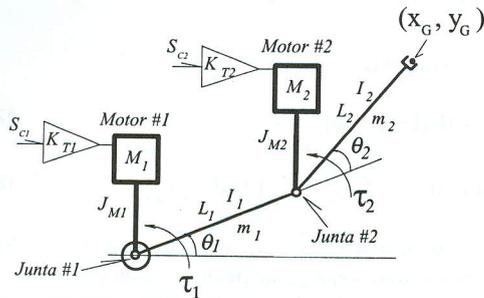
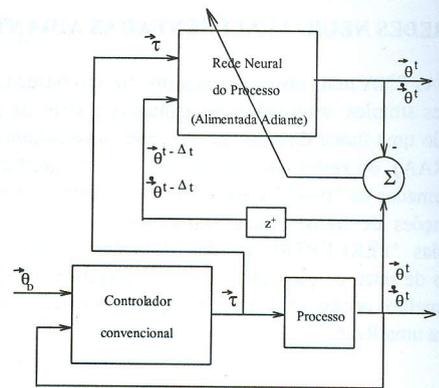


Figura 2. Manipulador Bi-articulado no Plano.

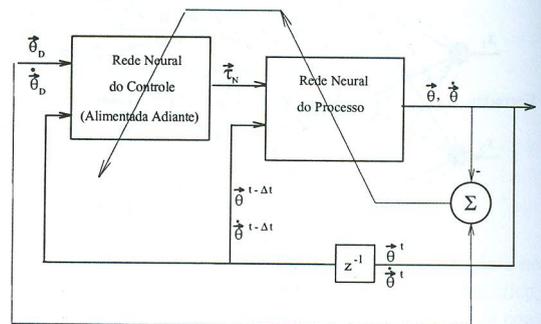
Diferentes métodos já foram empregados para resolver problemas de controle altamente não-lineares como esse. Kawato et al [8] resolvem o problema com RNA hierárquicas utilizando controle inverso. Nguyen e Widrow [9] utilizam uma rede para aprender as características dinâmicas do sistema e outra para aprender a controlar a rede que emula o processo. Esta última aproximação desmembra o problema em duas etapas. Segundo os autores, uma rede com duas camadas ocultas é capaz de representar o processo simulado. No exemplo por eles apresentado foram necessárias milhares de sessões de treinamento para as redes, consumindo muitas horas de CPU na "Workstation" que dispunham. Handelman, Lane e Gelfand [10] apresentam uma metodologia que integra um sistema baseado no conhecimento ("knowledge based system") a uma RNA de modo a habilitar o treinamento robótico. Eles concluem que, apesar das redes neurais terem mostrado eficiência no aprendizado, o treinamento deve ser rigidamente supervisionado pelo operador externo, ou seja, pelo sistema baseado no conhecimento. Um esquema para a solução desse problema em dois estágios está ilustrado na Fig. 3.

Para nosso propósito de estudo do desempenho de uma RNA tratando um problema desses, e dispondo-se de um controlador convencional, não é necessário utilizar o esquema de dois estágios. A rede de processo é dispensada e o treinamento da rede de controle é supervisionado pelo controlador convencional, de um modo semelhante ao apresentado por Handelman et al, [10]. Esse esquema, ilustrado na Fig. 4, simplifica a solução permitindo a obtenção de resultados mais rápidos. A rede nada mais fará do que aprender o que faz o controlador em situações para as quais seja habilitado e generalizará esse aprendizado para situa-

ções nas quais o próprio controlador não apresente bons resultados.



Primeiro Estágio - Treinamento da Rede de Processo



Segundo Estágio - Treinamento da Rede de Controle

Figura 3. Treinamento em Dois Estágios.

Uma rede para lidar com esse problema de uma maneira razoável precisa de três camadas [2]. No exemplo deste trabalho, a primeira camada possui seis unidades, a segunda sessenta e a terceira apenas as duas unidades de saída que geram os sinais correspondentes aos torques para controlar os dois motores (Fig. 5). A rede com 60 unidades na segunda camada, foi treinada para posicionar a garra em 18 pontos distribuídos na circunferência máxima do manipulador. Durante o treinamento foram dados seis segundos de tempo para o trajeto entre cada ponto e para parada da extremidade do manipulador após cada comando, permitindo-se a consolidação dos sinais estáticos de manutenção do manipulador em cada posição. Os padrões da dinâmica do manipulador alimentaram a rede a cada 3 ms. No início foi utilizada uma taxa de aprendizado $\eta = 0,02$, modificada posteriormente para $\eta = 0,05$. Após 4088 mudanças de alvo a rede foi capaz de posicionar, com grande imprecisão, a garra do manipulador nos pontos em que foi realizado o treinamento.

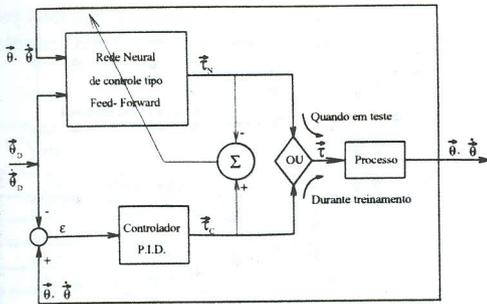


Figura 4. Esquema Utilizado no Exemplo.

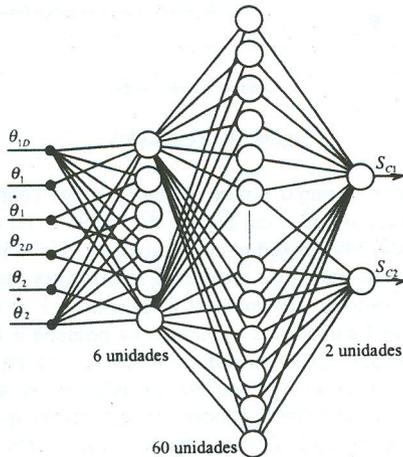


Figura 5. Arquitetura da RNA Alimentada Adiante.

A rede não foi capaz de atender a comandos para pontos interiores ao círculo descrito pela trajetória do braço totalmente estirado, ou seja, não teve essa capacidade de generalização. Uma fase de treinamento adicional foi conduzida sobre outros pontos e, após isso a rede foi capaz de controlar grosseiramente o manipulador em pontos internos ao círculo. O treinamento total representou 7 horas de tempo simulado, com a apresentação de 8.464.000 padrões à rede. A fase adicional de treinamento, de apenas 720 mudanças de alvo, deixou a rede apta a controlar o manipulador, ainda com grande imprecisão, como mostram as Fig. 6 e 7, onde pode ser verificada a trajetória e a evolução da distância da extremidade do manipulador ao alvo.

Identificação do Comportamento de um Sistema. Para aprender o comportamento de um processo, uma RNA recebe os mesmos sinais de entrada e é treinada com as saídas do sistema. A aplicação aqui exemplificada consiste no treinamento de uma RNA para prever o comportamento de regime permanente do sistema de bombeamento de água ilustrado na Fig. 9. Um operador atua na abertura da válvula (A), variando-a de 5% a 100%. Sempre após o sistema estabilizar são registrados os valores de nível nos 2 tanques, vazão nas duas linhas, altura manométrica da bomba (Δp), e a abertura da válvula (Tab. 1).

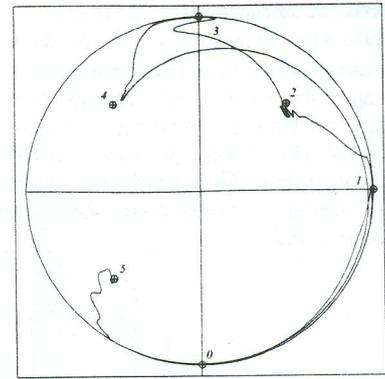


Figura 6. Trajetória do Manipulador no Teste.

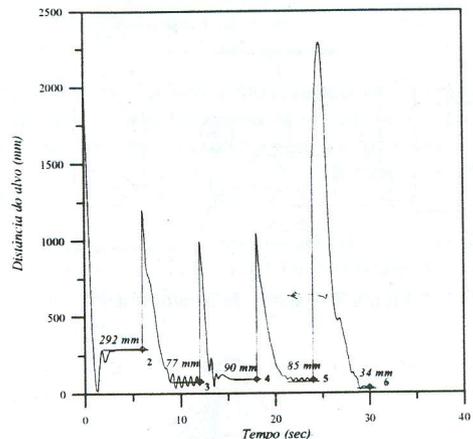


Figura 7. Distância da Garra ao Alvo.

TABELA 1. Dados Obtidos sobre o Processo

| Abertura A | Nível TQ - 1 | Nível TQ - 2 | Vazão Bomba | Altura Manométrica HB (m) | Vazão Tanque |
|------------|--------------|--------------|-------------------|---------------------------|-------------------|
| % | m | m | m ³ /h | | m ³ /h |
| 5 | 2,79 | 0,21 | 0,97 | 38,97 | 0,97 |
| 10 | 2,44 | 0,56 | 1,57 | 37,28 | 1,57 |
| 15 | 2,14 | 0,86 | 1,96 | 35,78 | 1,96 |
| 20 | 1,92 | 1,08 | 2,20 | 34,70 | 2,20 |
| 30 | 1,60 | 1,40 | 2,49 | 33,15 | 2,49 |
| 40 | 1,44 | 1,56 | 2,64 | 32,36 | 2,64 |
| 50 | 1,34 | 1,66 | 2,72 | 31,88 | 2,72 |
| 60 | 1,27 | 1,73 | 2,77 | 31,54 | 2,77 |
| 70 | 1,22 | 1,78 | 2,82 | 31,27 | 2,82 |
| 80 | 1,19 | 1,81 | 2,84 | 31,15 | 2,84 |
| 90 | 1,18 | 1,82 | 2,84 | 31,10 | 2,84 |
| 100 | 1,17 | 1,83 | 2,85 | 31,05 | 2,85 |

Neste exemplo foi utilizada uma RNA muito mais simples do que no caso de controle, com apenas duas camadas, 10 unidades na primeira camada e 5 na segunda (Fig. 9). O treinamento foi feito com base nos dados em negrito destacados na Tabela 1, representando 80% do universo de dados. A rede, recebendo como entrada o valor de A, entre

0% e 100%, foi treinada para reproduzir as demais variáveis do processo. O treinamento que produziu melhores resultados considerou uma constante de momentum $\alpha = 0,5$ e uma taxa de aprendizagem $\eta = 0,3$. Os resultados de previsão para todas as condições da Tabela 1 estão na Fig. 10. O desvio médio entre valores previstos e observados foi de 2,4%, o pior resultado foi na previsão da vazão na condição de 5% de abertura, com um desvio máximo na ordem de 20% (valor circulado).

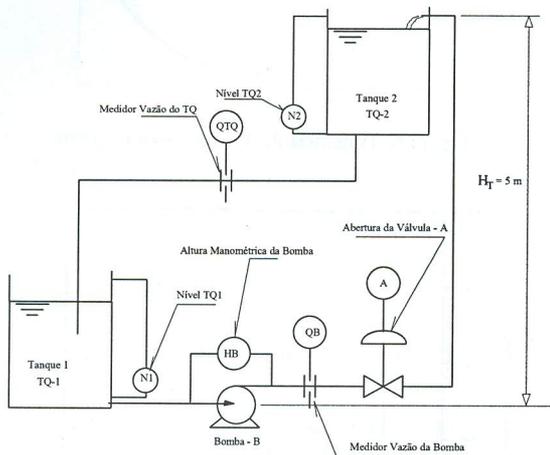


Figura 8. Sistema de Bombeamento.

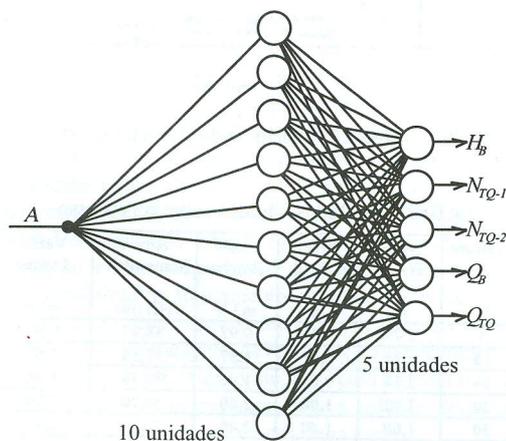


Figura 9. Arquitetura da RNA Alimentada Adiante.

Aproximação de Funções. O exemplo analisado é o de um reator PWR que possui um sistema de instrumentação monitorando a temperatura da água em 10 pontos igualmente espaçados no núcleo. São disponíveis distribuições de temperatura em intervalos de 60 dias (Tab. 2). Sabe-se que essa distribuição é função da queima. O objetivo é utilizar uma RNA para estimar o tempo de queima como função da distribuição de temperatura no núcleo, ou seja, a função inversa.

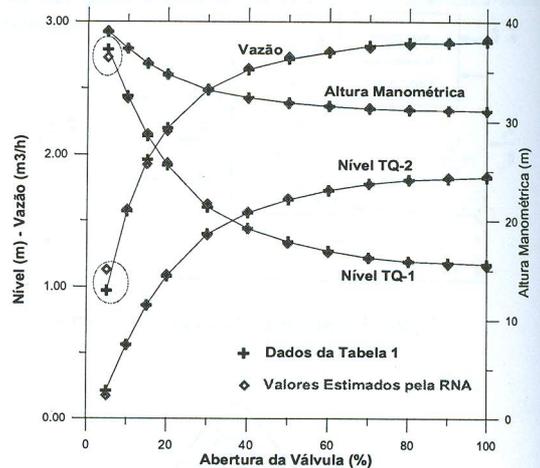


Figura 10. Resultados.

Neste exemplo foram utilizados, como conjunto de treinamento, os dados em negrito da Tabela 2. Os demais, de 150, 480, 540 e 600 dias como tabela de validação.

Foram analisadas redes com as seguintes características: a) 2 camadas, 10 unidades na primeira e 1 na segunda camada; b) 2 camadas, 20 unidades na primeira e 1 na segunda camada; e, c) 3 camadas, 10 unidades na primeira e segunda, e 1 na terceira camada. Inicialmente as redes foram treinadas por 30000 épocas, ou seja, 30000 apresentações de toda a tabela de treinamento. A rede com 20 unidades na primeira camada não forneceu resultados adequados com esse treinamento, necessitando de mais 10000 épocas de treinamento para apresentar resultados próximos das outras. A distribuição temporal de temperaturas se comporta como uma função contínua em todo o intervalo, desse modo só seriam necessárias duas camadas, mesmo assim foi avaliada uma rede com 3 camadas que se mostrou melhor para a previsão do comportamento fora da região de treinamento, ou seja, generalizou melhor o aprendizado. Para esse tipo de problema observou-se que não são necessárias mais do que 10 unidades na primeira camada, e que uma rede com três camadas apresenta resultados mais precisos (Fig. 11).

III. CONCLUSÕES

Redes Neurais Artificiais Alimentadas Adiante (RAA) treinadas pelo método de retro-propagação podem ter muitas aplicações na EN. Neste trabalho foram abordadas apenas algumas: controle, simulação de processos e aproximação de funções. Os exemplos apresentados mostram que as RAA, apesar de alguns problemas, como dependência da configuração para serem capazes de generalizar, podem ser aplicadas com muita facilidade e sucesso.

Os resultados indicam que as RAA ainda não se apresentam como uma solução ótima para problemas de controle. Comportamento de sistemas e aproximação de funções são problemas facilmente resolvidos e com precisão,

redes com apenas 15 unidades demonstraram excelente capacidade. Outros problemas são discutidos na segunda parte

deste trabalho comentando sobre o uso de RAA no reconhecimento de padrões.

TABELA 2. Dados de Distribuição de Temperatura no Núcleo do Reator PWR

| z (m) | T (dias) | | | | | | | | | | | |
|-------|----------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | 0 | 60 | 120 | 150 | 180 | 240 | 300 | 360 | 420 | 480 | 540 | 600 |
| 3.14 | 310,8 | 310,6 | 310,6 | 310,8 | 310,5 | 310,4 | 310,3 | 310,3 | 310,2 | 310,2 | 310,1 | 310,2 |
| 2.826 | 310,8 | 310,5 | 310,5 | 310,8 | 310,4 | 310,3 | 310,1 | 310 | 309,8 | 309,6 | 309,4 | 309,2 |
| 2.512 | 310,6 | 310,3 | 310,2 | 310,5 | 310 | 309,7 | 99 | 309 | 308,5 | 307,9 | 307,1 | 306,3 |
| 2.198 | 310,2 | 309,8 | 309,5 | 309,6 | 309 | 308,5 | 307,8 | 306,9 | 305,9 | 304,7 | 303,3 | 301,9 |
| 1.884 | 309,3 | 308,6 | 308 | 307,9 | 307,1 | 306,1 | 305 | 303,6 | 302 | 300,3 | 298,3 | 296,3 |
| 1.57 | 307,3 | 306,2 | 305,2 | 304,9 | 303,9 | 302,4 | 300,7 | 298,9 | 296,8 | 294,7 | 292,3 | 290 |
| 1.256 | 303,7 | 302,1 | 300,7 | 300,1 | 298,9 | 297 | 295 | 292,8 | 290,6 | 288,3 | 286 | 283,9 |
| 0,942 | 297,5 | 295,6 | 293,8 | 293,1 | 291,9 | 289,9 | 287,8 | 285,7 | 283,7 | 281,8 | 279,9 | 278,2 |
| 0,628 | 288,2 | 286,5 | 284,9 | 284,2 | 283,2 | 281,6 | 280 | 278,6 | 277,2 | 275,9 | 274,8 | 273,8 |
| 0,314 | 277,2 | 276,2 | 275,4 | 275 | 274,6 | 273,9 | 273,2 | 272,6 | 272,1 | 271,7 | 271,3 | 271 |
| 0 | 270 | 270 | 270 | 270 | 270 | 270 | 270 | 270 | 270 | 270 | 270 | 270 |

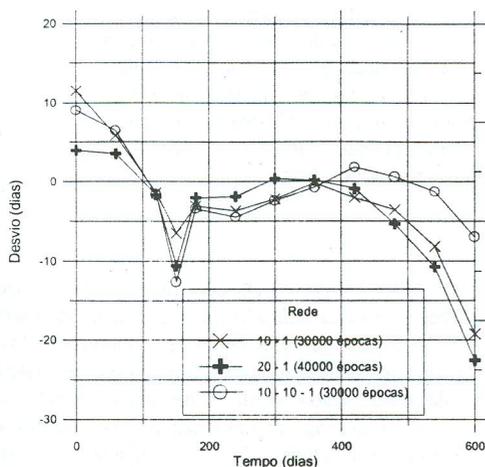


Figura 11. Desvio das Estimativas.

REFERÊNCIAS

- [1] Baptista F., B.D., Cabral, E.L.L., Soares, A.J. (1998) **A New Approach to Artificial neural Networks**, IEEE Transactions on neural Networks, Vol. 9, No. 6, pp.1167-1179.
- [2] Gonçalves, I. M. P. (2000), **Detection and Isolation of Multiple Faults in Nuclear Plant System**, Proceedings of the ANS 2000 (American Nuclear Society Meeting), June 4-8, 2000, San Diego, California.
- [3] Kim, H-K, Lee, S-H, and Chang, S-H, **Neural Network Model for Estimating Departure from Nucleate Boiling Performance of a Pressurized Water Reactor Core**, *Nuclear Technology*, vol. 1001, pp. 111-122, Feb. 1993.
- [4] Werbos, P.J. (1974) **Beyond regression: New Tools for Prediction and Analysis in the Behavioral Sciences**. Cambridge, MA: Nov. 1974. Ph.D. dissertation, Committee on Appl. Math., Harvard Univ.
- [5] Rumelhart, D.E., Hinton, G.E., Williams, R.J., (1986) **Learning Representations of back-propagation errors**, Nature (London), vol. 323, pp. 533-536.

[6] Psaltis, D., Sideris, A., Yamamura, A.A., (1988) **A Multi-layered Neural Network Controller**, IEEE Control Systems Magazine, p. 17-21, April 1988.

[7] Baptista F., B.D. (2000), **Utilização do Conceito de Redes Neurais de Contatos Sinápticos Múltiplos no Controle de Trajetória de um Manipulador Bi-Articulado**, Ref. AC9828, Congresso Nacional de Engenharia Mecânica, CONEM2000, Agosto de 2000.

[8] Kawato, M.; Uno, Y.; Isobe, M.; Susuki, R. (1988) **Hierarchical Neural Network Model for Voluntary Movement with Application to Robotics**. IEEE Control Systems Magazine, p. 8-16, April 1988.

[9] Nguyen, D. H.; Widrow, B. (1990) **Neural Networks for Self-Learning Control Systems**. IEEE Control Systems Magazine, p. 18-23, April 1990.

[10] Handelman, D.A.; Lane, S.H.; Gelfand, J.J. (1990) **Integrating Neural Networks and Knowledge-Based Systems for Intelligent Robotic Control**. IEEE Control Systems Magazine, p. 77-87, April 1990.

ABSTRACT

Artificial Neural Networks (ANN) can be defined as "parallel systems composed of layers of simple processing units highly interconnected and inspired in the human brain." ANN can be used to solve problems of difficult modeling, when the data are fail or incomplete and in problems of control of high complexity. Several problems related with network training and generalization are to be solved to a safe utilization in nuclear plants systems. This work, divided into two parts, intends to begin a discussion on three ANN concepts: feed-forward neural networks, self-organized maps (SOM), and multi-synaptic neural networks. The discussion will cover control applications, approximation of functions and pattern recognition. A few set of samples are commented. This first part focus on feed-forward neural networks with the back-propagation algorithm.