

# Resolução de equações diferenciais por redes neurais artificiais: problemas com gradientes elevados e domínios arbitrários

Luiz Henry Monken e Silva, Ivo Neitzel e Ed Pinheiro Lima\*

Programa de Pós-Graduação em Engenharia Química, Universidade Estadual de Maringá, Av. Colombo, 5790, 87020-900, Maringá, Paraná, Brasil. \*Autor para Correspondência. e-mail: ed\_p\_lima@yahoo.com.br

**RESUMO:** Neste artigo a habilidade das redes neurais perceptron multicamada em interpolar foi utilizada para analisar duas classes de problemas de contorno. A primeira classe é formada por equações diferenciais em que a solução pode apresentar gradientes elevados e a segunda classe é formada de equações diferenciais definidas em domínios arbitrários. As metodologias propostas por Lagaris *et al.* (1998) foram estendidas para casos de equações diferenciais sujeitas às condições de Cauchy e condições de contorno mistas. Os resultados fornecidos pelo método da rede neural se apresentam precisos quando comparados com os resultados analíticos ou por métodos numéricos de resolução de equações diferenciais. A precisão alcançada nos resultados e a facilidade no manuseio do método para resolver estes problemas de contorno encorajaram a continuidade da pesquisa, particularmente no tocante à convergência e estabilidade numérica.

**Palavras chave:** redes neurais, equações diferenciais, gradientes elevados.

**ABSTRACT. Resolution of differential equations with artificial neural networks: high gradients and arbitrary domains problems.** In this paper, the ability of the multilayer perceptron neural network (MLP) in interpolation was used to analyze two classes of boundary value problems. The first class is formed by differential equations, with solutions which can have high gradients and the second are partial differential equations, defined on arbitrary shaped domain. Also, the methodologies proposed by Lagaris *et al.* (1998) were enlarged for differential equations subjected to Cauchy and mix boundary conditions type. The results of the artificial neural network method are very precise when comparison to the analytical ones or those of classical numerical methods to solve differential equations. The precision achieved in the results and the ability to handle the method, to solve those boundary value problems, were encouraging to keep the research, particularly on an important direction, concerning convergence and numerical stability.

**Key words:** neural networks, differential equations, high gradients.

## Introdução

As capacidades de aproximação de funções por meio de redes neurais artificiais são bem conhecidas desde o final da década de 1980, como pode ser constatado pelos resultados alcançados por Funahashi (1989), Girosi e Poggio (1989), Hornik (1989) e Cybenko (1989). Em síntese, qualquer função mensurável pode ser aproximada por uma rede neural progressiva de camada escondida simples.

Apesar desses resultados teóricos e de suas variadas aplicações, são recentes os primeiros trabalhos que usam tais capacidades para resolução de equações ou sistemas de equações diferenciais ordinárias – problema de valor inicial e de valor no contorno – e de equações diferenciais parciais – problemas estacionários, a valor no contorno, como

se pode ver em Lagaris *et al.* (1998).

Este trabalho tem por objetivo demonstrar a aplicação de redes neurais a problemas de interesse na engenharia como, por exemplo, aqueles em que as soluções apresentam gradiente elevado e problemas de equações diferenciais parciais com contornos irregulares. Os resultados obtidos pela utilização de redes neurais na solução destes problemas são comparados com os de métodos de diferenças finitas e de elementos finitos.

## Metodologia

### Contornos Regulares

O método de utilização de redes neurais na solução de equações diferenciais proposto por Lagaris *et al.* (1998) é baseado na construção de uma

função de interpolação global que satisfaça, exatamente, as condições de contorno do problema, na forma

$$\psi_i(\mathbf{x}) = A(\mathbf{x}) + F(\mathbf{x}, N(\mathbf{x}, \mathbf{p})), \quad \mathbf{x} \in \bar{D}, \quad (1)$$

Onde:

$\bar{D} = D \cup B$ ,  $D \subseteq \mathbb{R}^n$  e  $B$  é o contorno de  $D$ ;

$A$  é uma função definida em  $\bar{D}$  que satisfaz as condições de contorno do problema;

$F$  é uma função construída de forma a não contribuir com as condições de contorno e que depende de  $\mathbf{x} \in D$  e de uma rede neural progressiva  $N(\mathbf{x}, \mathbf{p})$ , constituída por  $n$  unidades de alimentação do vetor de entrada  $\mathbf{x} \in D$ ;

$H$  unidades de processamento, resultando  $H \cdot (n+2)$  parâmetros ajustáveis (pesos) em  $\mathbf{p}$  e uma unidade de saída.

A função de interpolação (1) é aplicável à problemas de equações diferenciais ordinárias (EDO), sistemas de EDO e a equações diferenciais parciais (EDP). O termo que satisfaz o contorno, por um lado, melhora qualitativa e quantitativamente a aproximação, mas, por outro lado, é uma forte restrição ao emprego do método a qualquer tipo de condição de contorno e a geometrias arbitrárias.

A rede neural empregada para definir  $F$  presente em (1) conta com pesos e *bias* incógnitos, que serão determinados para minimizar um funcional formado pela soma dos quadrados dos resíduos,  $R(\mathbf{x}_i)$  que resultam da substituição da função definida em (1) à equação diferencial,

$$G(\mathbf{x}, \psi(\mathbf{x}), \nabla \psi(\mathbf{x}), \nabla^2 \psi(\mathbf{x})) = 0, \quad \mathbf{x} \in D \quad (2)$$

colocados nos nós,  $\mathbf{x}_i \in D$ , de uma malha pré-estabelecida em  $D$ , isto é,

$$R(\mathbf{x}_i) = G(\mathbf{x}_i, \psi_i(\mathbf{x}_i), \nabla \psi_i(\mathbf{x}_i), \nabla^2 \psi_i(\mathbf{x}_i)), \quad \forall \mathbf{x}_i \in D \quad (3)$$

Com  $\psi_i$  definida em (1) denotando a solução aproximada que satisfaz as condições de contorno, o problema passa a ser de minimização irrestrita do seguinte funcional:

$$\min_{\mathbf{p}} \sum_{\mathbf{x}_i \in D} \left\{ G[\mathbf{x}_i, \psi_i(\mathbf{x}_i, \mathbf{p}), \nabla \psi_i(\mathbf{x}_i, \mathbf{p}), \nabla^2 \psi_i(\mathbf{x}_i, \mathbf{p})] \right\}^2. \quad (4)$$

A minimização do funcional em (4) pode ser considerada como um procedimento de treinar a rede neural em que o erro correspondente a cada

vetor de entrada  $\mathbf{x}_i$  será o valor  $R(\mathbf{x}_i)$  que terá de se tornar zero. Entretanto, o cálculo do erro envolve, além da saída da rede, as derivadas da saída em relação às suas entradas.

Para chegar às derivadas das redes neurais Lagaris et al. (1998) apresentaram um procedimento onde considera uma rede neural perceptron multicamada com vetor de entrada  $\mathbf{x} = (x_1, \dots, x_n)$ , uma camada escondida com  $H$  unidades sigmoidais e uma saída linear, ou seja

$$N(\mathbf{x}, \mathbf{p}) = \sum_{i=1}^H v_i \sigma \left( \sum_{j=1}^N w_{ij} x_j + u_i \right) \quad (5)$$

Onde:

$w_{ij}$  são pesos referentes à unidade de entrada  $j$  e à unidade escondida  $i$ ;

$v_i$  são pesos referentes à unidade escondida  $i$  e a unidade de saída;

$u_i$  é o *bias* da unidade escondida  $i$  e  $\sigma$  é a função de ativação sigmóide.

Para minimizar o funcional em (4), Lagaris et al. (1998) sugerem o método *quasi-Newton*. Para isso, deve-se calcular o gradiente de  $N$  em relação aos parâmetros da rede original, que utiliza o gradiente para cada problema em estudo, um processo que se mostra trabalhoso. De forma alternativa, podem ser utilizadas rotinas de otimização, como as do pacote IMSL, em que o cálculo do gradiente é numérico, o que diminui a precisão, mas aumenta a generalidade do método. Têm-se ainda métodos de otimização que não necessitam do cálculo do gradiente, como o *simulated annealing*, presentes no livro *Numerical Recipes* que, apesar de não ter a vantagem de ser um pacote otimizado como o IMSL, têm o código aberto e, com isso, muito mais adaptáveis ao problema em consideração.

### Contornos Irregulares

A restrição do método a domínios regulares é crítica, em particular, para problemas de equações diferenciais parciais, que em geral são definidos em domínios com geometria complexa. Levantando esta restrição, Lagaris et al. (2000) propuseram um método aplicável a domínios cuja geometria possa ter contornos complexos, onde a fronteira é definida por um conjunto de pontos pertencentes a ela e se apresentam com pequenas distâncias entre si.

A função de interpolação é construída a partir de duas redes neurais, uma *perceptron* multicamada e outra de funções-base radiais, RBF. As duas redes funcionam numa forma de sinergia. A rede RBF, mais localizada em seus centros, é utilizada para satisfazer as condições de contorno nos pontos que o

define. A MLP, por sua vez, atua principalmente no domínio.

O método é descrito considerando uma equação diferencial parcial

$$L\psi = f, \quad (6)$$

Onde:

$L$  é o operador diferencial e  $\psi = \psi(\mathbf{x})$ ,  $\mathbf{x} \in D \subset \mathbb{R}^n$ , sendo  $B$  sua fronteira, sujeita à condições de contorno que podem ser do tipo Dirichlet, Neumann ou Cauchy, esta última apresentada em Lima (2002), podendo a geometria ser complexa.

O contorno será considerado como um conjunto contendo  $M$  pontos escolhidos para representá-lo. Dessa forma, as condições de contorno serão discretizadas, ficando

$$\psi(\mathbf{r}_i) = b_i, \quad i = 1, 2, \dots, M, \quad (7)$$

$$\mathbf{n}_i \cdot \nabla \psi(\mathbf{r}_i) = c_i, \quad i = 1, 2, \dots, M, \quad (8)$$

$$\mathbf{n}_i \cdot \nabla \psi(\mathbf{r}_i) + \psi(\mathbf{r}_i) = d_i, \quad i = 1, 2, \dots, M, \quad (9)$$

para condições de Dirichlet, Neumann e Cauchy, respectivamente, onde  $\mathbf{n}_i$  é o vetor normal externo à fronteira no ponto  $\mathbf{r}_i$ .

A solução da equação diferencial em (6) tem início pela discretização do domínio  $D$  por meio de uma malha contendo  $K$  pontos, isto é,  $x_i \in D$ ,  $1 \leq i \leq K$ . Em seguida, define-se uma função de interpolação  $\psi_M$ , como aproximação para a solução  $\psi$ , que, substituída na equação a ser resolvida e nas condições de contorno resultam, em geral, resíduos que, pelo método de colocação, são ponderados por distribuições de Dirac definidas nos nós da malha e contorno, resultando o sistema algébrico

$$L\psi_M(x_i, \mathbf{p}) = f(x_i) \quad \forall x_i \in D, \quad (10)$$

e ainda

$$\begin{aligned} \psi_M(\mathbf{r}_j) &= b_j, \\ \mathbf{n}_i \cdot \nabla \psi_M(\mathbf{r}_j) &= c_j \end{aligned} \quad (11)$$

ou

$$\mathbf{n}_i \cdot \nabla \psi_M(\mathbf{r}_j) + \psi_M(\mathbf{r}_j) = d \quad \forall r_i \in B \quad (12)$$

onde  $x_i \in D$ ,  $r_j \in B$ ,  $1 \leq i \leq K$  e  $1 \leq j \leq M$ .

O valor da função de interpolação,  $\psi_M(\mathbf{x}, \mathbf{p})$ , depende do vetor de entrada  $\mathbf{x}$  e de um conjunto de parâmetros ajustáveis  $\mathbf{p}$ . Desta forma, o problema de minimização fica sendo:

$$\min_{\mathbf{p}} E(\mathbf{p}) = \sum_{i=1}^K (L\psi_M(x_i, \mathbf{p}) - f(x_i))^2, \quad (13)$$

sujeito às restrições impostas pelas condições de contorno

$$\psi_M(\mathbf{r}_i, \mathbf{p}) = b_i, \quad i = 1, 2, \dots, M, \quad (14)$$

ou

$$\mathbf{n}_i \cdot \nabla \psi_M(\mathbf{r}_i, \mathbf{p}) = c_i, \quad i = 1, 2, \dots, M, \quad (15)$$

ou ainda

$$\mathbf{n}_i \cdot \nabla \psi_M(\mathbf{r}_i, \mathbf{p}) + \psi_M(\mathbf{r}_i, \mathbf{p}) = d_i, \quad i = 1, 2, \dots, M, \quad (16)$$

para condições de contorno do tipo Dirichlet, Neumann e Cauchy, respectivamente. A forma adotada por Lagaris *et al.* (2000) para resolver este problema de minimização consiste de duas etapas: a primeira usa o método da penalidade e a segunda se baseia na sinergia dentre duas redes neurais progressivas. Do método da penalidade, resultam os parâmetros ajustáveis da rede MLP que serão utilizados como pesos iniciais para o procedimento sinérgico MLP-RBF na segunda etapa.

Na etapa de penalidade a função de interpolação  $\psi_M$  é definida por

$$\psi_M(\mathbf{x}, \mathbf{p}) = N(\mathbf{x}, \mathbf{p}) \quad (17)$$

e o funcional erro a ser minimizado é construído na forma

$$E(p, \eta) = \sum_{i=1}^K (LN(x_i, \mathbf{p}) - f(x_i))^2 + \eta \sum_{i=1}^M (N(\mathbf{r}_i, \mathbf{p}) - b_i)^2, \quad (18)$$

ou

$$E(p, \eta) = \sum_{i=1}^K (LN(x_i, \mathbf{p}) - f(x_i))^2 + \eta \sum_{i=1}^M (\mathbf{n}_i \cdot \nabla N(\mathbf{r}_i, \mathbf{p}) - c_i)^2, \quad (19)$$

ou

$$E(p, \eta) = \sum_{i=1}^K (LN(x_i, \mathbf{p}) - f(x_i))^2 + \eta \sum_{i=1}^M (\mathbf{n}_i \cdot \nabla N(\mathbf{r}_i, \mathbf{p}) + N(\mathbf{r}_i, \mathbf{p}) - d_i)^2 \quad (20)$$

para condições de contorno do tipo Dirichlet, Neumann e Cauchy, respectivamente. Nas equações (18) a (20), o fator de penalidade  $\eta$  pode tomar valores positivos elevados à medida que se desejar que as condições de contorno sejam precisamente satisfeitas.

Na etapa de sinergia, a função de interpolação é definida a partir de duas redes neurais progressivas, na forma,

$$\psi_M(\mathbf{r}_i, \mathbf{p}) = N(x, \mathbf{p}) + \sum_{l=1}^M q_l e^{-\lambda \|x - \alpha \mathbf{r}_l + \mathbf{h}\|^2} \quad (21)$$

onde  $\|\cdot\|$  denota a norma Euclideana e  $N(\mathbf{x}, \mathbf{p})$  é um perceptron multicamada (MLP) com  $\mathbf{p}$  sendo o vetor de pesos e *bias*. O somatório em (21) representa uma rede RBF com  $M$  unidades escondidas que compartilham o fator exponencial comum  $\lambda$ . Nela aparecem ainda  $\alpha$ , que é um escalar e um vetor constante  $\mathbf{h}$ . Os parâmetros  $\lambda$ ,  $\alpha$  e  $\mathbf{h}$  são escolhidos de forma apropriada para se ter problemas bem condicionados.

Para um dado vetor  $\mathbf{p}$  de parâmetros da rede MLP, os coeficientes  $q_l$  são unicamente determinados fazendo com que as condições de contorno sejam satisfeitas nos pontos do contorno, isto é,

$$b_i - N(\mathbf{r}_i, \mathbf{p}) = \sum_{l=1}^M q_l e^{-\lambda \|\mathbf{r}_i - \alpha \mathbf{r}_l + \mathbf{h}\|^2}; \quad (22)$$

ou

$$c_i - \mathbf{n}_i \cdot \nabla N(\mathbf{r}_i, \mathbf{p}) = -2\lambda \sum_{l=1}^M q_l e^{-\lambda \|\mathbf{r}_i - \alpha \mathbf{r}_l + \mathbf{h}\|^2} \mathbf{n}_i \cdot (\mathbf{r}_i - \alpha \mathbf{r}_l + \mathbf{h}) \quad (23)$$

para condições de contorno do tipo Dirichlet e Neumann, respectivamente, com  $i=1, 2, \dots, M$ . Assim, de forma a obter os parâmetros  $q_l$  que satisfaçam as condições de contorno, deve-se resolver o sistema linear

$$Aq = \gamma, \quad A_{ij} = e^{-\lambda \|\mathbf{r}_i - \alpha \mathbf{r}_j + \mathbf{h}\|^2}, \quad \gamma_i = b_i - N(\mathbf{r}_i, \mathbf{p}) \quad (24)$$

$$Kq = \delta, \quad K_{ij} = -2\lambda e^{-\lambda \|\mathbf{r}_i - \alpha \mathbf{r}_j + \mathbf{h}\|^2} \mathbf{n}_i \cdot (\mathbf{r}_i - \alpha \mathbf{r}_j + \mathbf{h}), \quad \delta_i = c_i - \mathbf{n}_i \cdot \nabla N(\mathbf{r}_i, \mathbf{p}) \quad (25)$$

no caso das condições serem do tipo Dirichlet ou Neumann, respectivamente. Convém ressaltar que os tipos de condições de contorno podem ser estendidos para condições do tipo Cauchy e ainda as condições podem ser do tipo mista, definidas em partes da fronteira.

Segundo Lagaris *et al.* (2000), o modelo baseado na sinergia MLP-RBF satisfaz precisamente as condições de contorno nos pontos que representam o contorno, mas demanda grande esforço computacional devido à necessidade da resolução de um sistema algébrico linear a cada iteração. Além disso, o cálculo do gradiente do funcional objetivo implica na solução de um sistema algébrico linear adicional de mesma ordem para cada componente do gradiente como será visto adiante. Por outro lado, o método da penalidade é muito eficiente, mas não satisfaz precisamente as condições de contorno. Por isso o método da penalidade é utilizado para obter um modelo razoável que satisfaça aproximadamente as condições de contorno para, em seguida, ser

refinado pelo método da sinergia com poucas iterações.

## Resultado e discussão

Para a minimização do funcional erro, neste trabalho foram utilizadas as rotinas DUMCGF e DUMINF da biblioteca IMSL. A utilização destas duas rotinas se deve ao fato da DUMINF não aceitar quaisquer pesos iniciais quando o número de parâmetros chega a 40, ou seja, uma rede 2:10:1, a arquitetura utilizada em praticamente todos os problemas de equações diferenciais parciais que foram resolvidos. A rotina DUMINF, de convergência quadrática, apresenta um desempenho muito superior em relação ao da DUMCGF, tanto de velocidade de convergência, quanto na precisão.

A vantagem da DUMCGF é que, ao contrário da DUMINF, pesos arbitrários podem ser utilizados como pesos iniciais para qualquer número de parâmetros. Para aproveitar as melhores características computacionais de cada uma dessas sub-rotinas utilizou-se o processo de executar poucas iterações, realizadas em pouco tempo e, com os pesos finais, iniciar o programa novamente, mas com a rotina DUMINF.

## Problemas com Gradientes Elevados

Com a finalidade de averiguar o comportamento numérico e computacional do método em problemas que podem apresentar gradientes elevados no domínio da solução, considera-se o seguinte problema de equação diferencial ordinária a valor no contorno, assim expresso:

$$-\frac{d}{dx} \left[ a(x) \frac{d\psi(x)}{dx} \right] + b(x)\psi(x) = f(x), \quad 0 < x < 1 \quad (26)$$

sendo,

$$a(x) = \frac{1}{\alpha} + \alpha(x - \bar{x})^2,$$

$$b(x) = 0,$$

$$f(x) = 2 \left\{ 1 + \alpha(x - \bar{x}) \left[ \operatorname{tg}^{-1}(\alpha(x - \bar{x})) + \operatorname{tg}^{-1}(\alpha\bar{x}) \right] \right\}$$

sujeito às condições de contorno:  $\psi(0)=0$  e  $\psi(1)=0$ , sendo  $\alpha$  e  $\bar{x}$  dois parâmetros dados. A solução exata é:

$$\psi_a(x) = (1-x) \left[ \operatorname{tg}^{-1}(\alpha(x - \bar{x})) + \operatorname{tg}^{-1}(\alpha\bar{x}) \right] \quad (27)$$

Para um problema de EDO de ordem dois, o funcional erro a ser minimizado é definido por:

$$E[\mathbf{p}] = \sum_i \left\{ \frac{d^2 \psi_i(x_i)}{dx^2} - f\left(x_i, \psi_i(x_i), \frac{d\psi_i(x_i)}{dx}\right) \right\}^2. \quad (28)$$

A função de interpolação para este problema é definida construindo a função  $A$  de (1). Para um problema de EDO de ordem dois com condições de Dirichlet a função de interpolação pode ter a forma:

$$\psi_i(x) = A(1-x) + Bx + x(1-x)N(x, \mathbf{p}) \quad (29)$$

onde  $A = \psi(0)$  e  $B = \psi(1)$ . Para o problema em consideração, tem-se

$$\psi_i(x) = x(1-x)N(x, \mathbf{p}) \quad (30)$$

Dois situações foram analisadas, variando-se os parâmetros da equação, a saber:

- i)*  $\alpha = 5, \quad \bar{x} = 0,2;$
- ii)*  $\alpha = 100, \quad \bar{x} = 0,36388.$

Para o caso *i*, a solução analítica sobreposta pela resposta da rede está mostrada na Figura 1, juntamente com os erros na solução obtida, que ficaram na sétima casa decimal.

Pode-se notar na Figura 2 que o erro aumenta na vizinhança do ponto onde a solução apresenta um gradiente em forma de degrau. O erro máximo é menor do que o que procede da solução deste mesmo problema por elementos finitos.

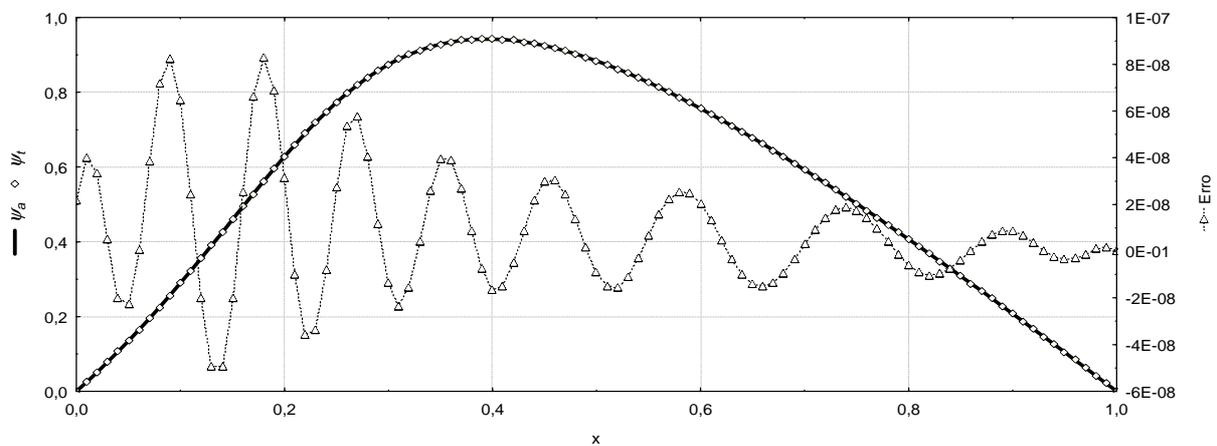


Figura 1. Solução analítica, resultado da função de interpolação e o erro na solução obtida pela função de interpolação (30).

No caso *ii* existe um gradiente elevado em torno do ponto  $\bar{x}$ , como é mostrado na Figura 2.

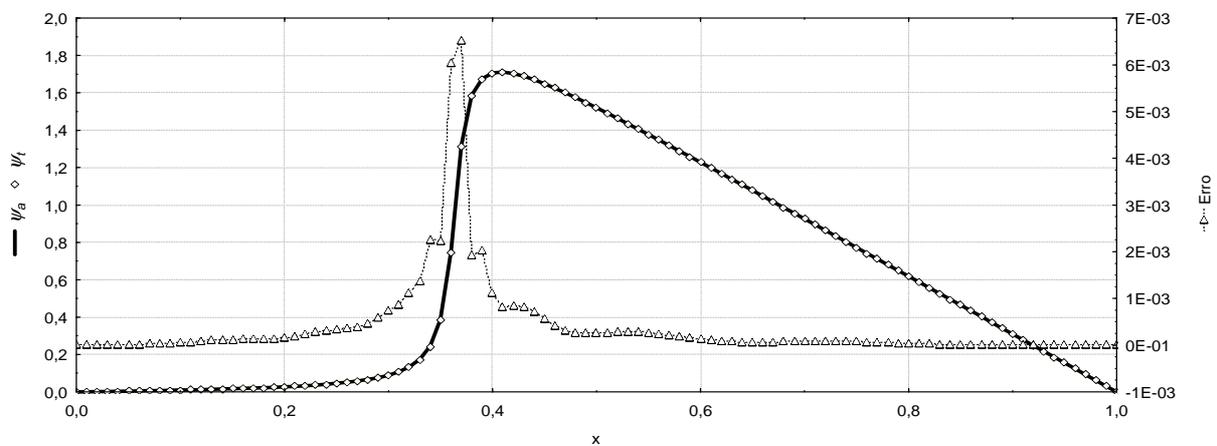


Figura 2. Solução analítica e resultado da função de interpolação Erro na solução obtida pela função de interpolação (30).

A rede utilizada para ambos os problemas foi 1:10:1, colocada numa malha com 101 pontos em  $[0,1]$  e minimizada com a DUMINF. Estes mesmos problemas, quando resolvidos pelo método de elementos finitos utilizando funções de interpolação Lagrangeanas, apresentam resultados distantes dos exatos. Esta dificuldade é, em parte, contornada pelo emprego de funções de interpolação assimétricas, com todas as desvantagens em se ter que adaptar o método para essas funções. Mesmo assim, os resultados são apenas aceitáveis. Já o emprego de redes neurais não apresentou esta dificuldade, mesmo no caso de elevado gradiente em  $\bar{x}$ .

O próximo problema estudado apresenta comportamento de camada limite no início do intervalo de interesse para a solução. Nesse sentido, tomou-se o problema tratado por Latta (1964, *apud* Rice e Do, 1995), expresso pela equação diferencial:

$$\varepsilon \frac{d^2 \psi}{dx^2} + \frac{d\psi}{dx} + \psi = 0, \quad \varepsilon \ll 1, x \in (0,1) \quad (31)$$

sujeita as condições de contorno  $\psi(0)=0$  e  $\psi(1)=2$ . A solução exata é:

$$\psi_a(x) = \frac{-2e^{s_1 x} + 2e^{s_2 x}}{e^{s_2} - e^{s_1}}, \quad (32)$$

Sendo

$$s_{1,2} = \frac{-1 \pm \sqrt{1 - 4\varepsilon}}{2\varepsilon}.$$

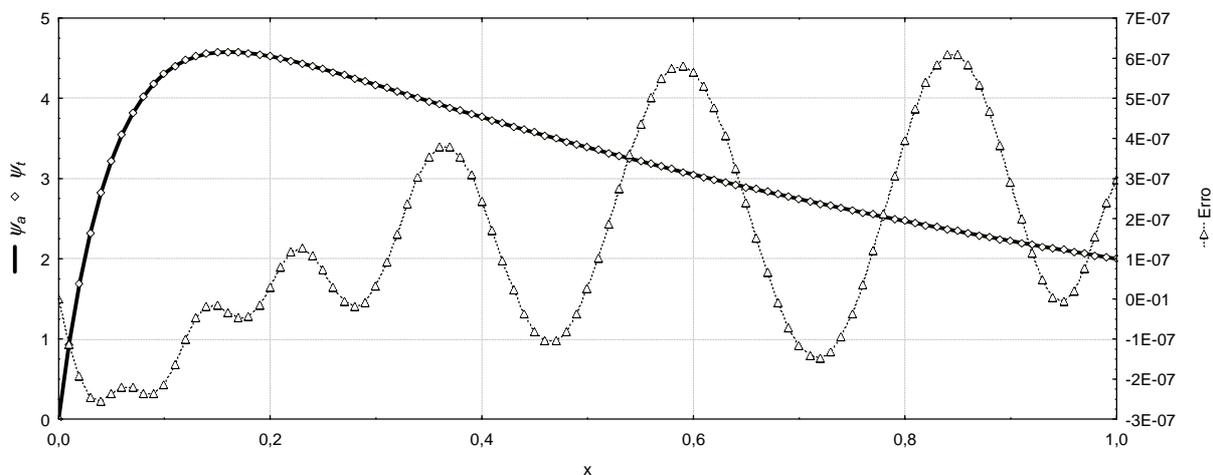


Figura 3. Solução analítica e resultado da função de interpolação Erro na solução obtida pela função de interpolação (33).

Conforme Rice e Do (1995), quando o valor do parâmetro  $\varepsilon$  é pequeno nesta equação, o método de perturbação é indicado para resolver este problema. Contudo, os resultados obtidos são distantes dos esperados na região em que há elevado gradiente. Neste caso, a alternativa a aproximação em duas etapas: uma denominada interna – que trata do comportamento de camada limite – e outra externa – que trata da solução longe da região de elevado gradiente na solução para, em seguida, as duas aproximações serem compatibilizadas e assim resultar numa aproximação aceitável do problema, que é conhecido como método das múltiplas escalas.

Para o emprego do método de rede neural, uma função de interpolação, partindo de (29), é tomada na forma:

$$\psi_t(x) = x + x(1-x)N(x, \mathbf{p}) \quad (33)$$

Novamente, a malha de pontos de colocação contém 101 pontos equidistantes em  $[0,1]$ . Na Figura 3 estão mostradas a solução analítica, exata, sobreposta pela solução aproximada obtida pela rede neural 1:10:1 e o erro entre as soluções exata e obtida pela função de interpolação.

Como se pode constatar, a precisão está na sexta casa decimal e, inclusive, na região de camada limite os erros são menores do que aqueles nos pontos mais afastados dela.

**Contornos Irregulares**

Seja uma EDP homogênea, com contorno irregular e condições de Dirichlet, expressa por

$$\nabla^2 \psi(x, y) = 0, \quad x, y \in \Omega \subseteq \square^2 \quad (34)$$

sendo  $\Omega$  a região triangular mostrada na Figura , sujeita às condições de Dirichlet  $\psi(0, y) = 0$ ,  $\psi(x, 1-x) = 0$  e  $\psi(x, 0) = \text{sen}(\pi x)$ . A função de interpolação a ser utilizada pelo método da penalidade é:

$$\psi_t(x, y, \mathbf{p}) = N(x, y, \mathbf{p}) \quad (35)$$

enquanto que para o método da sinergia é:

$$\psi_t(x, y, \mathbf{p}) = N(x, y, \mathbf{p}) + \sum_{l=1}^M q_l e^{-\lambda \|x - \alpha_l + \mathbf{h}\|^2} \quad (36)$$

Para a comparação de resultados obtidos pelo método da penalidade e da sinergia, o problema foi resolvido pelo método de diferenças finitas usando

diferenças centrais. As numerações dos pontos da malha do domínio discretizado utilizadas na rede neural e no método de diferenças finitas podem ser vistas nas Figuras 4 e 5 **Figura** , respectivamente.

Os métodos da penalidade e da sinergia utilizaram uma rede 2:20:1, já que com 10 neurônios os resultados não foram precisos. Os resultados para os métodos da penalidade e da sinergia estão mostrados na Tabela 1. Constatase, ao se comparar os resultados aproximados obtidos por rede neural com os de diferenças finitas, na coluna MDF, que o método da sinergia não foi mais preciso que o da penalidade, apesar de que os pesos iniciais da sinergia foram os finais da penalidade. Tanto para a penalidade quanto para a sinergia foi utilizada a rotina DUMCGF, o que inclui o cálculo do gradiente, aumentando o tempo de otimização e, no caso do método da sinergia, insere erros de truncamento no cálculo do vetor  $q_l$ , além dos erros de arredondamento presentes na decomposição LU.

**Tabela 1.** Resultados dos métodos da penalidade e sinergia.

Ponto	x	y	MDF	Penalidade	Dif. Relativa	Sinergia	Dif. Relativa
1	0,1	0,1	2,1720E-01	2,1779E-01	2,7000E-03	2,1762E-01	1,9000E-03
2	0,1	0,2	1,4836E-01	1,4835E-01	-4,6100E-05	1,4549E-01	-1,9300E-02
3	0,1	0,3	9,7267E-02	9,7196E-02	-7,3300E-04	9,4514E-02	-2,8300E-02
4	0,1	0,4	6,0155E-02	5,9774E-02	-6,3300E-03	5,9469E-02	-1,1400E-02
5	0,1	0,5	3,4169E-02	3,3693E-02	-1,3900E-02	3,7188E-02	8,8300E-02
6	0,1	0,6	1,7036E-02	1,6820E-02	-1,2700E-02	2,4882E-02	4,6100E-01
7	0,1	0,7	6,8077E-03	6,9354E-03	1,8700E-02	1,9657E-02	1,8900E+00
8	0,1	0,8	1,7019E-03	2,0292E-03	1,9200E-01	1,8342E-02	9,7800E+00
9	0,2	0,1	4,1145E-01	4,1157E-01	2,9700E-04	4,0641E-01	-1,2200E-02
10	0,2	0,2	2,7895E-01	2,7850E-01	-1,6300E-03	2,7348E-01	-1,9600E-02
11	0,2	0,3	1,8056E-01	1,8006E-01	-2,7300E-03	1,7692E-01	-2,0200E-02
12	0,2	0,4	1,0918E-01	1,0867E-01	-4,7200E-03	1,0894E-01	-2,2600E-03
13	0,2	0,5	5,9486E-02	5,9003E-02	-8,1200E-03	6,3839E-02	7,3200E-02
14	0,2	0,6	2,7166E-02	2,6797E-02	-1,3600E-02	3,6853E-02	3,5700E-01
15	0,2	0,7	8,4934E-03	8,3065E-03	-2,2000E-02	2,3536E-02	1,7700E+00
16	0,3	0,1	5,6184E-01	5,6163E-01	-3,7500E-04	5,4923E-01	-2,2500E-02
17	0,3	0,2	3,7545E-01	3,7463E-01	-2,1900E-03	3,6564E-01	-2,6100E-02
18	0,3	0,3	2,3683E-01	2,3605E-01	-3,2800E-03	2,3099E-01	-2,4600E-02
19	0,3	0,4	1,3654E-01	1,3596E-01	-4,2100E-03	1,3566E-01	-6,3900E-03
20	0,3	0,5	6,7426E-02	6,7009E-02	-6,1800E-03	7,2431E-02	7,4200E-02
21	0,3	0,6	2,3648E-02	2,3193E-02	-1,9200E-02	3,5167E-02	4,8700E-01
22	0,4	0,1	6,5145E-01	6,5087E-01	-8,9500E-04	6,3089E-01	-3,1600E-02
23	0,4	0,2	4,2419E-01	4,2314E-01	-2,4800E-03	4,0956E-01	-3,4500E-02
24	0,4	0,3	2,5476E-01	2,5391E-01	-3,3400E-03	2,4612E-01	-3,3900E-02
25	0,4	0,4	1,3271E-01	1,3229E-01	-3,1500E-03	1,3051E-01	-1,6600E-02
26	0,4	0,5	5,0033E-02	4,9964E-02	-1,4000E-03	5,5080E-02	1,0100E-01
27	0,5	0,1	6,6872E-01	6,6775E-01	-1,4500E-03	6,4157E-01	-4,0600E-02
28	0,5	0,2	4,1510E-01	4,1398E-01	-2,6900E-03	3,9625E-01	-4,5400E-02
29	0,5	0,3	2,2531E-01	2,2452E-01	-3,4900E-03	2,1377E-01	-5,1200E-02
30	0,5	0,4	8,9504E-02	8,9424E-02	-8,9000E-04	8,5343E-02	-4,6500E-02
31	0,6	0,1	6,0833E-01	6,0724E-01	-1,8000E-03	5,7751E-01	-5,0700E-02
32	0,6	0,2	3,4217E-01	3,4114E-01	-3,0100E-03	3,2069E-01	-6,2800E-02
33	0,6	0,3	1,4187E-01	1,4118E-01	-4,8300E-03	1,2780E-01	-9,9100E-02
34	0,7	0,1	4,7138E-01	4,7070E-01	-1,4400E-03	4,4100E-01	-6,4400E-02
35	0,7	0,2	2,0339E-01	2,0274E-01	-3,1800E-03	1,8183E-01	-1,0600E-01
36	0,8	0,1	2,6479E-01	2,6481E-01	8,5600E-05	2,3944E-01	-9,5800E-02

No caso da EDP expressa por:

$$\nabla^2 \psi(x, y) = 0, \quad x, y \in \Omega \subseteq \mathbb{R}^2 \quad (37)$$

sendo  $\Omega$  a região triangular mostrada na Figura 4, sujeita à condição de contorno mista, isto é, de Dirichlet

$\psi(0,y)=0$ ,  $\psi(x,1-x)=0$  e Neumann  $\frac{\partial}{\partial y} \psi(x,0)=sen(\pi x)$ . Para a comparação dos resultados da penalidade, o problema foi resolvido pelo método de diferenças finitas, com diferenças centrais. Neste problema não foi utilizado o método da sinergia pelas razões descritas no problema anterior. A Figura 6 mostra os pontos de colocação para a entrada da rede assim como os vetores normais, utilizados no método da penalidade no caso de condições de Neumann, como visto em (19).

A função de interpolação para o método da penalidade é definida por:

$$\psi_i(x, y, \mathbf{p}) = N(x, y, \mathbf{p}) \quad (38)$$

Os resultados obtidos pelo método da penalidade para duas redes, 2:10:1 e 2:20:1, estão mostrados na Tabela 2, onde são comparados relativamente com os de diferenças finitas, na coluna MDF. O total de incógnitas na resolução por diferenças finitas foi de 66 parâmetros, enquanto que para a rede neural foi de apenas 40 parâmetros para 10 neurônios e 80 para 20 neurônios. Pôde-se observar que o maior número de neurônios não traz melhora significativa se for levado em consideração o tempo maior para treinamento, o que indica que um aumento no número de neurônios não implica uma melhora nos resultados.

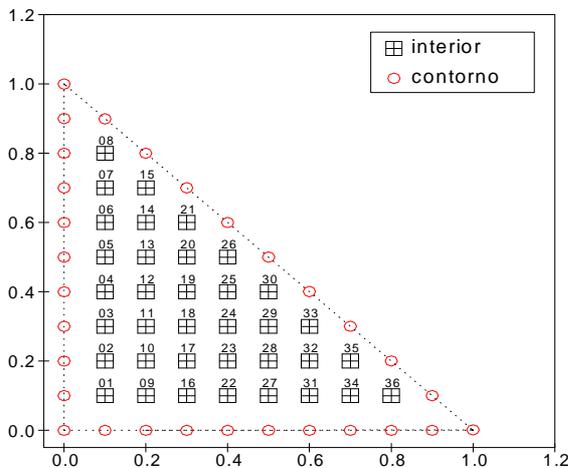


Figura 4. Discretização do domínio triangular para rede neural.

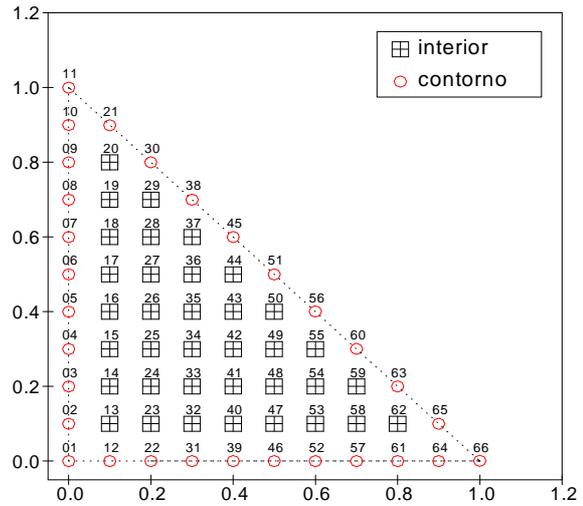


Figura 5. Discretização do domínio triangular com pontos numerados para MDF.

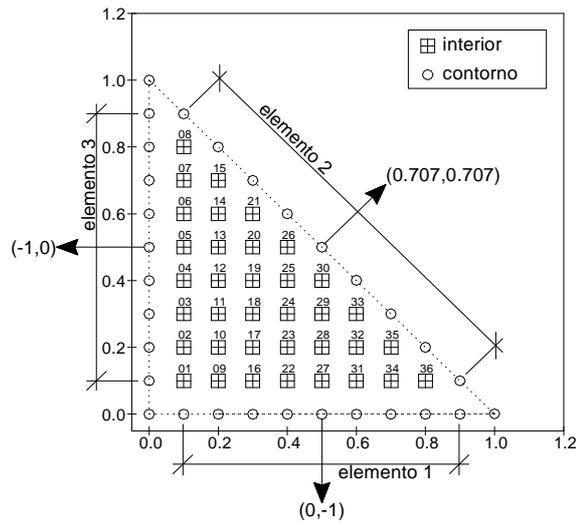


Figura 6. Discretização do domínio triangular com vetores normais para rede neural.

**Conclusão**

No presente trabalho, tanto a metodologia da penalidade para inserção das condições de contorno no funcional, onde se tem um problema de otimização sem restrições, como a de sinergia MLP-RBF, em que as condições de contorno são satisfeitas num conjunto de pontos representativos do contorno, foram testadas numericamente. Nesta última, o cálculo do gradiente do funcional objetivo implica na solução de um sistema algébrico linear para cada componente do gradiente e de ordem igual ao número de pontos do contorno. Isto se mostra inviável de ser realizado computacionalmente pelos atuais equipamentos com as configurações usuais de mercado.

Tabela 2. Comparação dos resultados do método da penalidade e MDF.

Ponto	x	y	MDF	10 neurônios		20 neurônios	
				Valor	Dif. Relativa	Valor	Dif. Relativa
1	0,1	0,1	6,2872E-02	6,6457E-02	5,7008E-02	5,8324E-02	7,2345E-02
2	0,1	0,2	4,1218E-02	4,3650E-02	5,9008E-02	3,9419E-02	4,3651E-02
3	0,1	0,3	2,6199E-02	2,8009E-02	6,9106E-02	2,5624E-02	2,1950E-02
4	0,1	0,4	1,5869E-02	1,6936E-02	6,7200E-02	1,5747E-02	7,6980E-03
5	0,1	0,5	8,9046E-03	9,4904E-03	6,5790E-02	8,9030E-03	1,6964E-04
6	0,1	0,6	4,4132E-03	4,7919E-03	8,5815E-02	4,4176E-03	-1,0135E-03
7	0,1	0,7	1,7600E-03	2,0045E-03	1,3890E-01	1,7564E-03	2,0667E-03
8	0,1	0,8	4,4001E-04	5,9464E-04	3,5142E-01	4,3652E-04	7,9346E-03
9	0,2	0,1	1,1650E-01	1,2160E-01	4,3785E-02	1,1034E-01	5,2829E-02
10	0,2	0,2	7,5801E-02	8,0471E-02	6,1601E-02	7,4002E-02	2,3740E-02
11	0,2	0,3	4,7708E-02	5,1333E-02	7,5964E-02	4,7520E-02	3,9465E-03
12	0,2	0,4	2,8374E-02	3,0675E-02	8,1109E-02	2,8581E-02	-7,2887E-03
13	0,2	0,5	1,5336E-02	1,6562E-02	7,9936E-02	1,5517E-02	-1,1847E-02
14	0,2	0,6	6,9880E-03	7,5239E-03	7,6684E-02	7,0689E-03	-1,1568E-02
15	0,2	0,7	2,1870E-03	2,3380E-03	6,9012E-02	2,2047E-03	-8,0905E-03
16	0,3	0,1	1,5204E-01	1,6066E-01	5,6726E-02	1,4941E-01	1,7314E-02
17	0,3	0,2	9,7782E-02	1,0575E-01	8,1494E-02	9,8877E-02	-1,1201E-02
18	0,3	0,3	6,0459E-02	6,6100E-02	9,3298E-02	6,1978E-02	-2,5124E-02
19	0,3	0,4	3,4582E-02	3,7858E-02	9,4754E-02	3,5604E-02	-2,9567E-02
20	0,3	0,5	1,7077E-02	1,8614E-02	9,0051E-02	1,7552E-02	-2,7827E-02
21	0,3	0,6	6,0162E-03	6,4632E-03	7,4306E-02	6,1515E-03	-2,2496E-02
22	0,4	0,1	1,6094E-01	1,7967E-01	1,1637E-01	1,7066E-01	-6,0394E-02
23	0,4	0,2	1,0283E-01	1,1607E-01	1,2882E-01	1,1035E-01	-7,3153E-02
24	0,4	0,3	6,1766E-02	6,9507E-02	1,2533E-01	6,6058E-02	-6,9488E-02
25	0,4	0,4	3,2416E-02	3,6243E-02	1,1807E-01	3,4390E-02	-6,0890E-02
26	0,4	0,5	1,2373E-02	1,3757E-02	1,1181E-01	1,2980E-02	-4,9042E-02
27	0,5	0,1	1,3284E-01	1,7660E-01	3,2942E-01	1,7123E-01	-2,8900E-01
28	0,5	0,2	9,0824E-02	1,0974E-01	2,0831E-01	1,0608E-01	-1,6795E-01
29	0,5	0,3	5,1360E-02	5,9929E-02	1,6684E-01	5,7677E-02	-1,2298E-01
30	0,5	0,4	2,0944E-02	2,4135E-02	1,5234E-01	2,3003E-02	-9,8319E-02
31	0,6	0,1	1,3676E-01	1,5195E-01	1,1108E-01	1,5071E-01	-1,0201E-01
32	0,6	0,2	7,6269E-02	8,6726E-02	1,3711E-01	8,5342E-02	-1,1896E-01
33	0,6	0,3	3,1907E-02	3,6589E-02	1,4674E-01	3,5730E-02	-1,1983E-01
34	0,7	0,1	1,0606E-01	1,0909E-01	2,8513E-02	1,1139E-01	-5,0171E-02
35	0,7	0,2	4,5583E-02	4,8831E-02	7,1250E-02	4,9120E-02	-7,7593E-02
36	0,8	0,1	5,4947E-02	5,4828E-02	-2,1796E-03	5,8398E-02	-6,2790E-02

Também se observou que, apesar de se ter utilizado os pesos resultantes do método da penalidade, o método da sinergia não apresentou uma melhora significativa. Os valores obtidos são próximos aos dos do método da penalidade. Pode-se atribuir isso à rotina de otimização, que faz o cálculo do gradiente numericamente, o que pode acumular erros na solução do sistema linear.

O método da penalidade, no entanto, se mostrou de adaptação rápida e simplificada a novos problemas. Para a resolução de equações diferenciais parciais com contornos irregulares, o método da penalidade se apresentou preciso, assim como para contornos regulares.

Nos problemas de equações diferenciais ordinárias os resultados foram precisos, até mesmo para casos de gradiente elevado, onde apresentaram comportamento melhor do que outros métodos de aproximação. Para contornos irregulares, os resultados foram precisos para penalidade e para a sinergia, mas com as observações feitas acima.

O que se pode constatar do método de

resolução de equações diferenciais por redes neurais artificiais é que é um método de interpolação global. Após o treinamento da rede neural, tem-se uma função de interpolação que descreve, para qualquer ponto do domínio, a solução aproximada da equação diferencial, sem a necessidade de procedimentos de pós-processamento, que em alguns métodos chegam a ser complexos. Além disso, do que foi observado, a restrição na aplicabilidade do método pode ser associada à rotina de otimização, que no caso deste trabalho foi adotado o pacote fechado IMSL. De forma alternativa, a utilização de rotinas disponíveis gratuitamente, como o caso do Numerical Recipes, pode dar a flexibilidade de ajustes necessários para vários tipos de problemas.

Por fim, uma característica importante em relação à utilização de redes neurais para a resolução de equações diferenciais é em relação à linearidade ou não dos problemas. As redes neurais tratam, por assim dizer, problemas lineares e não lineares da mesma forma, como se fossem do mesmo tipo.

**Referências**

- CYBENKO, G. *Approximation by superpositions of a sigmoidal function*. *Math. Control Signals Syst.*, New York, v. 2, n. 4, p. 303-314, 1989.
- FUNAHASHI, K. *On the approximate realization of continuous mappings by neural networks neuro-control systems theory and applications*. *Neural Netw.*, New York, v. 2, n. 3, p. 183-192, 1989.
- GIROSI, F.; POGGIO, T. *Networks and the best approximation property*. *Biol. Cybern.*, New York, v. 63, p. 169-176, 1990.
- HORNIK, K. et al. *Multilayer feedforward networks are universal approximators*. *Neural Netw.*, New York, v. 2, n. 5, 1989, p. 359-366.
- LAGARIS, I. E. et al. *Artificial neural network for solving ordinary and partial differential equations*. *IEEE Trans. Neural Netw.*, New York, v. 9, n. 5, p. 987-1000, 1998.
- LAGARIS, I.E. et al. *Neural-Network methods for boundary value problems with irregular boundaries*. *IEEE Trans. Neural Netw.*, New York, v. 11, n. 5, p. 1041-1049, 2000.
- LIMA, E.P. *Soluções de modelos diferenciais discretos ou contínuos por redes neurais artificiais: contornos irregulares*. 2002. Dissertação (Mestrado)–Universidade Estadual de Maringá, Maringá, 2002.
- RICE, R.G.; DO, D.D., *Applied Mathematics and Modeling for Chemical Engineers*. New York: John Wiley & Sons, 1995.

Received on November 26, 2004.

Accepted on April 29, 2005