

Aplicação de Redes Neurais RBF Treinadas com Algoritmo ROLS e Análise Wavelet na Predição de Tráfego em Redes Ethernet

Flávio Henrique Teles Vieira Rodrigo Pinto Lemos
Universidade Federal de Goiás – UFG - Escola de Engenharia Elétrica
Praça Universitária S/N- Setor Universitário 74605-220 Goiânia GO

Lee Luan Ling

Departamento de Comunicações –FEEC-Universidade Estadual de Campinas-UNICAMP
Caixa Postal 6110, 13083-970 Campinas, SP, Brasil

E-mails: flaviotel@decom.fee.unicamp.br, lemos@eee.ufg.br e lee@decom.fee.unicamp.br

Abstract

The use of a wavelet transform technique is investigated for Ethernet traffic prediction by RBF (Radial Basis Function) neural networks based on the combination of the signal approximation and detail predictions in different resolution levels. Then, useful information is extracted resulting in a more appropriate modelling of non-stationary time series by the RBF neural network. The ROLS (Regularised Orthogonal Least Squares) algorithm is applied for the neural networks training to produce smaller nets regarding the number of centers. This contributes to avoid problems of bad conditioning as well as to give an extrapolation capacity improvement verified by the mean-square error reduction of the prediction.

1. Introdução

As redes neurais MLP e RBF formam cada uma do seu modo, a espinha dorsal das redes neurais com aprendizagem supervisionada. As redes neurais podem ser usadas em tomadas de decisões não-lineares complexas e para aproximar o mecanismo gerador de dados de certos sistemas. A rede neural RBF em particular, tem sido aplicada com sucesso na predição de séries temporais não-lineares [1],[2],[3]. Por isso, objetivou-se neste trabalho o seu uso como ferramenta não só para modelar o comportamento de uma rede de computadores, mas também, para prever seu comportamento futuro a fim de propiciarmos aos mecanismos de controle de tráfego informações para tomada de decisões. O problema considerado é o de se prever o próximo valor de uma série temporal correspondente ao tráfego de uma rede Ethernet.

No entanto, a maioria do sucesso de predição das redes neurais está na predição de sinais estacionários, por não caracterizarem bem a variabilidade temporal da série. Uma vez que sinais reais são não apenas não-

lineares, mas também não-estacionários, é necessário desenvolver preditores que lidem com essas características.

Foi provada a capacidade das redes neurais com uma camada de neurônios escondida de aproximar qualquer função contínua [4]. A rede neural RBF também com esta capacidade, possui um treinamento que pode ser visto como uma interpolação e ser resolvido por uma inversão matricial. Entretanto, com essa aproximação pode-se encontrar problemas numéricos. Neste trabalho, utilizamos o algoritmo OLS que tem sido usado em várias aplicações como controle automático [5], redes neurais nebulosas [6] e outras aplicações.

A maioria dos métodos de predição existentes efetuam uma análise global dos dados. Tais métodos levam à perda de detalhes (underfitting) ou a inclusão de ruído (overfitting), por tratar os dados de forma única. Seria interessante levarmos em conta na predição do sinal, a predição de seus detalhes em cada escala, representados por séries com uma maior variação e a predição do sinal com menor riqueza de detalhes. Essa idéia é aplicada utilizando a análise wavelet.

2. Rede Neural RBF

A rede neural RBF é uma rede multicamadas com uma camada de nós de entrada (nós sensoriais), uma camada de nós escondidos e uma camada de saída. Redes RBF com mais de uma camada intermediária também já foram propostas [7]. Nesta rede neural, a ativação de um nó é função da distância entre seus vetores de entrada e os pesos. As funções de transferência entre a camada de entrada e a camada escondida são não-lineares e aquelas conectando a camada escondida com a camada de saída são lineares. Portanto, o mapeamento entrada-saída efetuado pela RBF é obtido por uma transformação não-linear e outra linear. Na saída os parâmetros ajustáveis são os pesos de uma combinação linear. Estes parâmetros podem ser

determinados usando método de mínimos quadrados. As redes RBF constroem aproximações locais para mapeamentos entrada-saída, resultando em um aprendizado mais rápido e sensibilidade reduzida à ordem de apresentação dos dados do conjunto de treinamento.

Seja um número inteiro $P < N$ (N =número de dados de treinamento), a equação 1 fornece a saída da rede neural da Figura 1:

$$y = \sum_{k=0}^P w_k \varphi(\underline{x}; \underline{t}_k) + w_o \quad (1)$$

em que \underline{t}_k representa os centros das funções de base radial.

Utilizando regularização no treinamento desta rede neural a função custo \mathcal{E}_R a ser minimizada é descrita por:

$$\mathcal{E}_R = \sum_{i=1}^N (d_i - f(x_i))^2 + \sum_{j=1}^P \lambda_j w_j^2 \quad (2)$$

Temos que $\underline{d} = [d_1, d_2, \dots, d_N]^T$ é o vetor de respostas desejadas, $\underline{w} = [w_0, w_1, w_2, \dots, w_P]^T$ é o vetor de pesos da rede neural RBF, λ_j é o parâmetro de regularização e $f(x_i)$ é a saída da rede neural para um vetor de entrada x_i .

A rede neural segundo essas equações, possui P nós na camada escondida. Em seu treinamento são encontradas as localizações dos centros \underline{t}_k das funções de base radial.

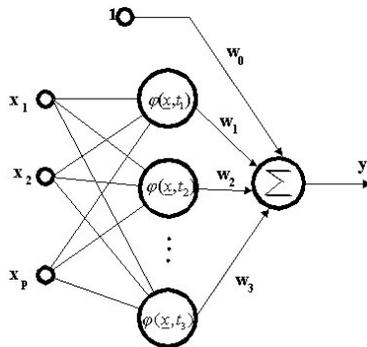


Figura 1: Rede neural RBF

Escolheu-se como função de base radial a função gaussiana:

$$\varphi(\underline{x}, \underline{t}_k) = \exp\left(-\frac{1}{\sigma_k^2} \|\underline{x} - \underline{t}_k\|^2\right), \quad k = 1, 2, 3, \dots, P \quad (3)$$

onde σ_k representa a largura da função radial e \underline{t}_k o seu centro. No entanto, investigações teóricas e resultados práticos sugerem que a escolha da não-linearidade não é crucial para o desempenho da rede neural [8].

$$y = \sum_{k=0}^P w_k \exp\left(-\frac{1}{\sigma_k^2} \|\underline{x} - \underline{t}_k\|^2\right) + w_o \quad (4)$$

Definindo a matriz de interpolação Φ , cujo tamanho é $N \times (P+1)$, onde N é o número de exemplos de treinamento e P é o número de funções de base radial.

$$\Phi = \begin{bmatrix} 1 & \varphi(\underline{x}_1, \underline{t}_1) & \varphi(\underline{x}_1, \underline{t}_2) & \dots & \varphi(\underline{x}_1, \underline{t}_P) \\ 1 & \varphi(\underline{x}_2, \underline{t}_1) & \varphi(\underline{x}_2, \underline{t}_2) & \dots & \varphi(\underline{x}_2, \underline{t}_P) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \varphi(\underline{x}_N, \underline{t}_1) & \varphi(\underline{x}_N, \underline{t}_2) & \dots & \varphi(\underline{x}_N, \underline{t}_P) \end{bmatrix} \quad (5)$$

Em termos matriciais, o vetor de pesos que minimiza a função custo \mathcal{E}_R (equação 2) é [9]:

$$\underline{w} = (\Phi^T \cdot \Phi + Q)^{-1} \Phi^T \cdot \underline{d} \quad (6)$$

onde tem-se:

$$Q = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_P \end{bmatrix} \quad (7)$$

Pode-se definir o número de nós na camada intermediária como sendo igual ao número de padrões de entrada, assim cada centro é situado sobre um vetor de entrada e a rede RBF mapeia com exatidão o vetor de entrada para a saída. Porém, esta interpolação exata pode ser indesejável principalmente quando se tem ruído, pois pode levar ao “overfitting”, em que não se consegue obter da rede neural uma boa generalização, ou seja, extrapolação para novos dados. Um dos motivos de se utilizar um número de centros menor que o número de padrões de entrada. A regularização também “desencoraja” a rede neural a realizar “overfitting” com o conjunto de treinamento.

O desempenho da rede RBF depende sensivelmente dos centros escolhidos. Mal condicionamento ocorre pela escolha de centros próximos. O algoritmo ROLS pode ser usado para seleção de um conjunto de centros a partir de candidatos fixos e evitar problemas de mal condicionamento, além de obter redes com um menor número de centros.

3. Algoritmo ROLS

As redes neurais MLPs possuem aprendizagens baseadas em técnicas de otimização não-linear e pode-se ter problemas de mínimos locais. Outras técnicas de otimização como algoritmos genéticos, apesar de serem capazes de encontrar um mínimo global, podem requerer extensa computação.

A seleção direta é um algoritmo não-linear que procura em um espaço discreto de conjuntos um subconjunto com o menor erro de predição. Começa-se

com um subconjunto vazio e acrescenta-se a cada iteração uma função base (centro), aquela que reduz a soma dos erros quadráticos, até que algum critério seja alcançado.

O algoritmo de mínimos quadrados ortogonais reduz a complexidade computacional da seleção direta. Consiste de uma ortogonalização de Gram-Schmidt que assegura que cada nova coluna a ser acrescentada na matriz de projeto será perpendicular a todas as outras colunas.

O ROLS se baseia na solução ótima para os pesos (equação 6):

$$\underline{w} = (H^T H + \lambda H)^{-1} H^T \underline{d} \quad (8)$$

onde H é chamada de matriz de projeto:

$$H = \begin{bmatrix} h_1(x_1) & h_2(x_1) & \cdots & h_m(x_1) \\ h_1(x_2) & h_2(x_2) & \cdots & h_m(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ h_1(x_p) & h_2(x_p) & \cdots & h_m(x_p) \end{bmatrix} \quad (9)$$

Esta matriz é fatorada em $H_m = \tilde{H}_m U_m$, em que \tilde{H}_m possui colunas ortogonais entre si e U_m é uma matriz triangular superior.

Seja Z uma matriz cujas colunas $\{z_f\}_{f=1}^M$ correspondem aos M possíveis candidatos:

$$Z = [z_1 \quad z_2 \quad \cdots \quad z_M] \quad (10)$$

A cada passo m , um vetor \tilde{z}_i é acrescentado na matriz de projeto ortogonalizada $\tilde{H}_m = [\tilde{H}_{m-1} \quad \tilde{z}_i]$ dado por:

$$\tilde{z}_f = z_f - \sum_{j=1}^m \frac{z_f^T \tilde{h}_j}{\tilde{h}_j^T \tilde{h}_j} \tilde{h}_j \quad (11)$$

tal que leva à maior diminuição do erro quadrático médio e que maximiza a equação:

$$\varepsilon r_m - \varepsilon r_{m+1} = \frac{(\underline{d}^T \tilde{z}_f)^2}{\lambda + \tilde{z}_f^T \tilde{z}_f} \quad (12)$$

O vetor de pesos ortogonalizado é calculado inicialmente:

$$\tilde{w}_m = \frac{\underline{d}^T \tilde{h}_j}{\lambda + \tilde{h}_j^T \tilde{h}_j} \quad (13)$$

para então calcularmos o vetor de pesos regular:

$$\underline{w}_m = U_m^{-1} \tilde{w}_m \quad (14)$$

em que se utiliza a matriz triangular superior U_m :

$$U_m = \begin{bmatrix} U_{m-1} & (\tilde{H}_{m-1}^T \tilde{H}_{m-1})^{-1} \tilde{H}_{m-1}^T z_f \\ 0_{m-1}^T & 1 \end{bmatrix} \quad (15)$$

Para calcularmos o valor do parâmetro de regularização, precisamos da matriz de projeção P :

$$\tilde{P}_{m+1} = I_p - \sum_{j=1}^m \frac{\tilde{h}_j \tilde{h}_j^T}{\lambda + \tilde{h}_j^T \tilde{h}_j} \quad (16)$$

O parâmetro de regularização é otimizado a cada adição de novo centro e é escolhido baseado na minimização do GCV (Generalised Cross Validation) [10]. Um novo valor é re-estimado depois de cada seleção de centro usando o λ prévio. As equações de re-estimação do parâmetro de regularização, que é iniciado com zero são:

$$\lambda = \frac{\left[\frac{\partial \text{traço}(\tilde{P}_m)}{\partial \lambda} \right] (\underline{d}^T \tilde{P}_m \underline{d})}{\text{traço}(\tilde{P}_m) \tilde{w}_m^T (\tilde{H}_m^T \tilde{H}_m + \lambda I_m)^{-1} \tilde{w}_m} \quad (17)$$

$$\frac{\partial \text{traço}(\tilde{P}_m)}{\partial \lambda} = \sum_{j=1}^m \frac{\tilde{h}_j \tilde{h}_j^T}{(\lambda + \tilde{h}_j^T \tilde{h}_j)^2} \quad (18)$$

A decisão de interromper o acréscimo de funções base pode ser feita monitorando-se algum critério de seleção, utilizamos o BIC (Bayesian Information Criteria) [10]:

$$\sigma_{BIC}^2 = \frac{n + (\ln(n) - 1)\gamma}{n - \gamma} \frac{\underline{d}^T P^2 \underline{d}}{n} \quad (19)$$

onde $\gamma = n - \text{traço}(P)$.

Quando este critério pára de diminuir, e começa a aumentar, interrompe-se o acréscimo de funções de base radial.

4. Transformada Wavelet

A transformada wavelet proporciona uma representação em multiresolução do sinal, tendo como idéia a utilização de diversas escalas e independência a escala. Uma função φ que satisfaz a condição de admissibilidade [11] é uma função wavelet. Desta condição, tira-se que a função wavelet mãe φ tem suporte compacto, ou seja, não tem componentes espectrais fora de um faixa de frequências e é oscilatória. Por dilatações e translações da função wavelet mãe obtemos uma família de funções:

$$\varphi_{s,t}(u) = |s|^{-p} \varphi\left(\frac{u-t}{s}\right) \quad (20)$$

onde $s \in \mathfrak{R}$, $p \geq 0$, s é o parâmetro de dilatação e t é o parâmetro de translação. Podemos definir a Transformada Wavelet Contínua (CWT) de uma função $f(t)$:

$$W(s, t) = \int_{-\infty}^{+\infty} f(u) \varphi_{s,t}(u) du = \langle f, \varphi_{s,t} \rangle \quad (21)$$

Obtemos a transformada discreta pela discretização temporal e em escala da transformada contínua. Assim, temos um conjunto enumerável de funções utilizando

uma discretização diádica em que $s = 2^m$, $t = n2^m t_0$ e $m, n \in \mathbb{Z}$:

$$\varphi_{m,n}(u) = |2|^{-\frac{m}{2}} \varphi(2^{-m}u - n) \quad (22)$$

Os coeficientes da expansão wavelet discreta são:

$$c_{m,n} = \langle f, \varphi_{m,n} \rangle \quad (23)$$

O conjunto $\varphi_{m,n}$ forma uma base ortonormal de $L^2(\mathfrak{R})$, espaço de funções onde \mathfrak{R} é o conjunto de números reais e cujas funções possuem energia finita [12]. Portanto, podemos reconstruir uma função da seguinte forma:

$$f = \sum_{m,n} c_{m,n} \varphi_{m,n}^* \quad (24)$$

Seja uma função $\phi \in L^2(\mathfrak{R})$, que será chamada função de escala, tal que a família de funções:

$$\phi_{j,k}(u) = |2|^{-\frac{j}{2}} \phi(2^{-j}u - k) \quad j, k \in \mathbb{Z} \quad (25)$$

é base ortonormal do subespaço V_j chamado de espaço de escala, formado pelas funções cujos detalhes estão na escala 2^j . Podemos representar uma função $f \in L^2(\mathfrak{R})$ por projeção ortogonal em V_j :

$$P_{V_j}(f) = \sum_k \langle f, \phi_{j,k} \rangle \phi_{j,k} \quad (26)$$

Quando j decresce, a largura de $\phi_{j,k}$ diminui, aumentando portanto, a frequência de resolução. Os detalhes que aparecem na escala 2^j estão presentes na escala 2^{j-1} , assim:

$$V_j \subset V_{j-1} \quad (27)$$

O espaço V_{j-1} é obtido acrescentando-se todas as funções de $L^2(\mathfrak{R})$ com frequências na faixa $[\alpha_j, \alpha_{j-1}]$, chamaremos esse espaço de W_j , que é gerado por uma base ortonormal wavelet $\{\varphi_{j,k} | k \in \mathbb{Z}\}$.

A representação de um sinal f na escala V_{j-1} é:

$$P_{V_{j-1}}(f) = \sum_k \langle f, \phi_{j,k} \rangle \phi_{j,k} + \sum_k \langle f, \varphi_{j,k} \rangle \varphi_{j,k} \quad (28)$$

5. Simulações e resultados

A análise wavelet pode ser utilizada para melhorar o desempenho de predição das redes neurais RBF pela decomposição do sinal em séries correspondentes a aproximações e detalhes em várias escalas. Para cada série obtida da decomposição wavelet, uma rede neural RBF fica responsável por prover uma regra de predição a um passo para esta série. A soma dessas predições pode ser vista como a predição da série original.

As séries temporais utilizadas neste trabalho foram retiradas das medições da Bellcore, arquivos BC-pOct89 e BC-Oct89Ext que podem ser obtidas no site: <http://www.acm.org/sigcomm/ITA/> e tem sido utilizadas em diversos estudos [13][14].

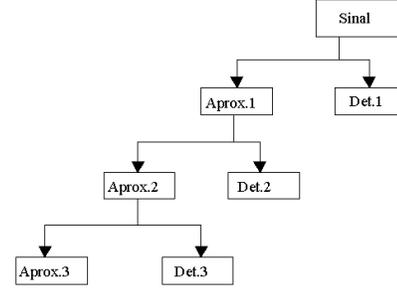


Figura 2: Decomposição em escala do sinal original

Para se avaliar a qualidade das predições foi usado o erro quadrático médio normalizado EQMN dado por:

$$EQMN = \frac{1}{\sigma^2 p} \sum_{k=1}^p [y(k) - \hat{y}(k)]^2 \quad (27)$$

onde $y(k)$ é o valor real da série, $\hat{y}(k)$ é o valor predito, σ^2 é a variância da seqüência real sobre o intervalo de duração da predição e p é o número de amostras do teste.

Seja a decomposição wavelet de um sinal $f(x)$ até uma escala, digamos $j=3$ (Figura 2). Obtemos a reconstrução do sinal naquela escala, a aproximação 3, que é uma versão do sinal original suavizada, com menos detalhes. Se acrescentarmos à aproximação 3, os sinais de detalhes nas escalas 3, 2 e 1, reconstruímos o sinal original. Ou seja, pode-se escrever para o sinal $f(x)$:

$$f(x) = P_{V_3}(f(x)) + \sum_{j=1}^3 P_{W_j}(f(x)) \quad (28)$$

Esta equação provê uma fórmula de reconstrução para a série original, levando em conta a decomposição da série até a escala 3. Como é aditiva, podemos da mesma maneira, adicionar predições para obter a predição da série original.

Utilizou-se redes neurais RBF na predição da série Bc-Octext com cinco elementos de entrada e otimizadas para cada série (Figura 3). As redes neurais foram treinadas com 1000 exemplos de treinamento da série Bc-Octext e 1000 pontos da mesma foram usados como teste. Obteve-se os seguintes erros médios quadráticos, escala relativa à largura da função de base radial e número de centros escolhidos:

- Aproximação 3: EQMN=0,1645, escala 5 e centros=4;
- Detalhe 3: EQMN=0,2480, escala 2 e centros=19;
- Detalhe 2: EQMN= 0,3275, escala 1 e centros=38;

•Detalhe 1: EQMN=0,5727, escala 0,5 e centros=17.

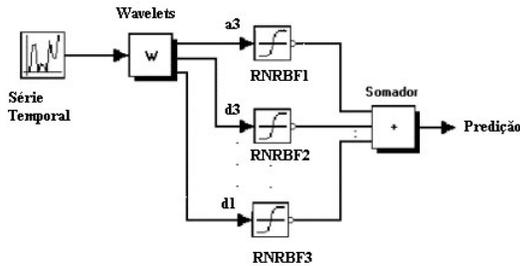


Figura 3: Redes neurais RBF otimizadas na predição total por soma de predições

A Figura 4 mostra os valores do BIC na escolha de centros para a aproximação 3 da série Bc-Octext:

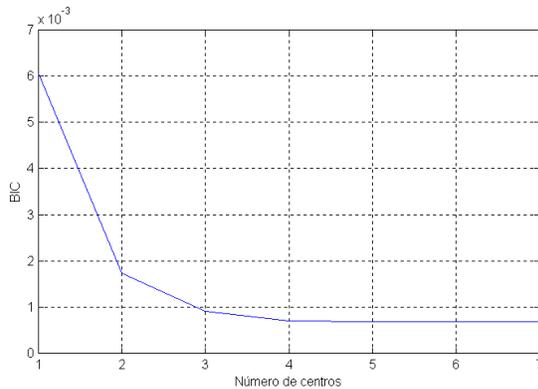


Figura 4: Decaimento do BIC com o acréscimo de centros no treinamento de rede neural RBF para a aproximação do sinal na escala 3.

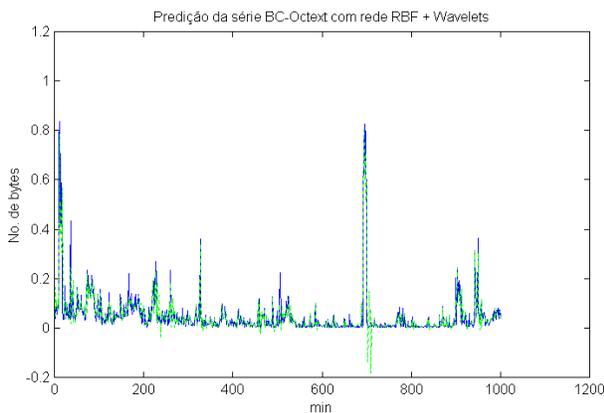


Figura 5: Predição (tracejado) da série Bc-Octext pela soma de predições multiescala. EQMN=0,2068

A partir do quarto centro selecionado o BIC se mantém estável, portanto quatro centros são escolhidos.

Obteve-se um EQMN total de 0,2068 para a série Bc-Octext (Figura 5).

A série Bc-Octint por apresentar um salto estatístico em torno do instante 1100 oferece uma maior dificuldade de predição pelas redes neurais. A simulação para a série Bc-Octint utilizou 30 elementos normalizados para o vetor de entrada com 900 pontos de treinamento e 800 pontos de teste, obtendo um EQMN de 0,1331 pela decomposição do sinal até a escala 3 (Figura 6).

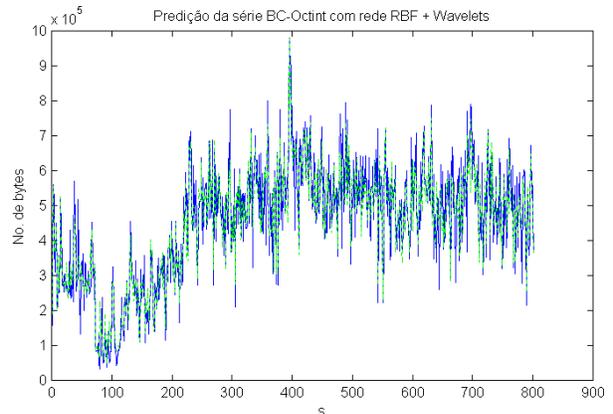


Figura 6: Predição (tracejado) da série Bc-Octint pela soma de predições multiescala. EQMN=0,1331

A técnica atingiu tal resultado devido entre outras coisas, a capacidade de predição da rede RBF de variações bruscas ‘retangulares’ que surgem com as aproximações do sinal com menos detalhes. Dificuldade esta encontrada por exemplo, pelas rede neurais FIR MLP [15]. Além do uso de redes neurais RBF otimizadas para cada série a ser predita.

Comparando-se com o desempenho de predição de outras redes neurais, vê-se uma redução significativa do EQMN de predição (Tabela 1). Para a rede neural MLP aplicada na predição da série Bc-Octext, utilizou-se os pontos de 1 a 800 para treinamento, de 801 a 1000 para validação e 1001 a 2000 para teste de predição, e para a série Bc-Octint os pontos de 1001 a 1700 para teste. Com a rede FIR MLP utilizou-se o mesmo número de pontos de treinamento e teste usado para a rede RBF.

Tabela 1: EQMN de predição de redes neurais

| Série | EQMN | | | |
|-----------|--------|---------|---------|------------------|
| | MLP | FIR MLP | RBF+OLS | RBF+OLS+Wavelets |
| Bc-Octext | 0,4037 | 0,4260 | 0,3962 | 0,2068 |
| Bc-Octint | 1,21 | 0,7408 | 0,5092 | 0,1331 |

Mantendo o mesmo número de entradas, simulou-se a predição da série Bc-Octint utilizando diferentes níveis de detalhes (Tabela 2).

Tabela 2: EQMN dos experimentos 1, 2, 3 e 4.

| EQMN | | | | | | |
|------|--------|--------|--------|--------|--------|--------|
| | Aprox1 | Det1 | Det2 | Det3 | Det4 | EQMN |
| Exp1 | 0,2301 | 0,6058 | - | - | - | 0,2552 |
| Exp2 | 0,0875 | 0,6058 | 0,3262 | - | - | 0,1657 |
| Exp3 | 0,0294 | 0,6058 | 0,3262 | 0,1896 | - | 0,1331 |
| Exp4 | 0,0185 | 0,6058 | 0,3262 | 0,1896 | 0,1896 | 0,1319 |

Nota-se uma diminuição do EQMN a medida que detalhes em escalas menores vão sendo considerados e uma maior facilidade de predição pela rede neural RBF das aproximações com menos detalhes.

6. Conclusões

Na rede neural RBF a camada de nós que utiliza funções de base radial transforma um conjunto de entrada não linearmente separável em um conjunto linearmente separável, agrupando os dados de entrada em "clusters". A camada de saída procura classificar os padrões recebidos da camada anterior. Dessa forma a rede neural RBF encontra padrões existentes nos dados de tráfego. Entretanto, este é bastante variável. A análise de detalhes em diversas escalas então é empregada, trazendo uma melhoria significativa no desempenho de predição.

Mostrou-se que o algoritmo proposto é uma ferramenta eficiente na modelagem e predição do comportamento de sistemas complexos e é estável, ou seja, independe da escolha inicial dos pesos. Observou-se que o mapeamento estático entre pontos presentes e futuros do tráfego pode não ser capaz de lidar com grandes variações estatísticas do sinal. Porém, a análise em escala e dos níveis de detalhes incorpora mais informações no tratamento da não-estacionaridade de séries temporais reais. Este método também dá oportunidade de uso de várias redes neurais, até mesmo de tipos de diferentes no tratamento de séries que irão compor a série original. Pode-se então, atingir o limite de predicabilidade [16] da série temporal pelo custo computacional de inclusão de redes neurais RBF em sua predição.

Referências

- [1] S.Chen. "Orthogonal least square learning algorithm for radial basis function networks", IEEE Trans. Neural networks, vol.2, no.2, pp.335-356,1989.
- [2] M.A.S.Potts e D.S.Broomhead. "Time series prediction with a radial basis function neural network", SPIE Adaptive Signal Processing, vol.1565, pp.255-266,1991.
- [3] D.S.Broomhead e D.Lowe. "Multivariable functional interpolation and adaptive networks", Complex Systems, vol.2, pp.321-355,1988.

[4] G.Cybenco. "Approximations by superposition of a sigmoidal function", Math. Control Signal Systems, vol.2, no.4, pp.303-314,1989.

[5] S.Chen e S.A.Billings. "Neural networks for nonlinear dynamic system modeling and identification". International Journal of Control, vol.56, no.2, pp.319-346,1992.

[6] J.-S.R.Jamg e C.-T.Sun. "Functional equivalence between radial basis function networks and fuzzy inference systems", IEEE Transactions on Neural Networks, vol.4, pp156-159, 1993.

[7] X. He; A. Lapedes. "Nonlinear modeling and prediction by successive approximations using radial basis functions" , Technical report, Los Alamos National Laboratory, 1991.

[8] M.J.D.Powell. "Radial basis function approximations to polynomials", in Proc.12th Biennial Numerical Analysis Conf.(Dundee), 1987, pp.223-241.

[9] S.Haykin. "Neural Networks - A Comprehensive Foundation". Prentice Hall,1994.

[10] M.J.L.Orr. "Regularization in the Selection of Radial Basis Function Centres", Neural Computation, 7(3):606-623, 1995.

[11] C.K.Chui. "An Introduction to wavelets", Department of Mathematics Texas, A&M University College Station, Texas, 1992.

[12] J.Gomes; L.Velho; S.Goldstein. "Wavelets: Teoria, Software e Aplicações". Rio de Janeiro: IMPA 1997.

[13] D.Veitch; P.Abry. "A wavelet based joint estimator of the parameters of long-range dependence," IEEE Trans. Inform. Theory-Special Issue on Multiscale Statistical Signal Analysis and Its Applications, vol. 45, no. 3, Apr. 1999.

[14] D.Veitch; P.Abry. "Wavelet analysis of long-range dependent traffic," IEEE Trans. Info. Theory, vol. 44, no. 1, pp. 2-15, 1998.

[15] F.H.T.Vieira. "Predição de tráfego em redes de comunicações utilizando redes neurais e análise wavelet-Alocação dinâmica de largura de faixa". Dissertação de mestrado. Universidade Federal de Goiás, Goiânia, Goiás, Brasil.

[16] A.Sang e S.Q.Li. "A predictability analysis of network traffic". Conference on Computer Communications,IEEE Infocom, New York, Mar. 2000.