

# Previsão de Vazões Médias Mensais: Uma Aplicação para as Usinas do Rio Paranapanema

M. Cicogna, M. Kadowaki, R. Ballini e S. Soares, Unicamp

## RESUMO

Este trabalho apresenta um modelo de rede neural nebulosa para previsão de vazões médias, juntamente com uma interface a qual possibilita ajudar o ajuste do modelo ao problema de previsão, além de uma visualização das previsões feitas através de gráficos ou de tabelas, de todo o histórico ou de períodos específicos. O modelo de rede neural nebulosa é baseado em um método de aprendizado construtivo onde grupos de neurônios competem quando a rede recebe uma nova entrada. A rede aprende os parâmetros fundamentais para definir as regras nebulosas e funções de pertinência para cada variável de entrada. O modelo vem sendo aplicado para o problema de previsão de vazões médias mensais das usinas hidroelétricas situadas no Rio Paranapanema. Os resultados vêm mostrando que a rede neural nebulosa apresenta um bom desempenho para previsão um passo à frente.

## PALAVRAS-CHAVE

Lógica Nebulosa; Modelagem Estocástica; Previsão de Vazões; Redes Neurais; Sistema Híbrido Nebuloso.

## I. INTRODUÇÃO

Análise e previsão de séries de vazões sazonais são de fundamental importância no planejamento da operação de sistemas de recursos hídricos. Uma das dificuldades na previsão dessas séries é a natureza não estacionária devido aos períodos de cheia e seca do ano. Estas previsões são realizadas atualmente por metodologias baseadas nos modelos de [7]. Como as séries de vazões mensais são séries sazonais com períodos de 12 meses, as quais, geralmente apresentam correlações periódicas, modelos de séries temporais que vêm sendo sugerido são os modelos auto-regressivos periódicos [1].

Em anos recentes muitas pesquisas vêm sendo desenvolvidas para aplicação de modelos de redes neurais artificiais em previsões de séries temporais por sua habilidade de aprendizado e capacidade de generalização, asso-

ciação e computação paralela. Estas qualidades as tornam capazes de identificar e assimilar as características das séries, tais como sazonalidade, periodicidade e tendência, na maioria das vezes camufladas por ruídos, sem necessitar da laboriosa formulação teórica, imprescindível para os procedimentos estatísticos [2].

Contudo, a capacidade de mapeamentos complexos das redes neurais cresce com o número de camadas e neurônios. Este aumento acarreta maior tempo de processamento bem como, uma considerável soma de dados. Na prática, os parâmetros devem ser ajustados rapidamente e somente uma pequena quantidade de dados é disponível [3]. Além disso, frequentemente, dados do mundo real apresentam ruídos, podendo conter contradições e imperfeições. Assim, modelos de análise de dados baseados em conjuntos nebulosos vêm sendo especialmente utilizados para tratar estas hipóteses [6].

Em anos recentes, surgiu um método promissor unindo os benefícios das redes neurais e da lógica fuzzy, resultando em um sistema integrado, onde a aprendizagem e o poder computacional das redes neurais, e a capacidade de representação e raciocínio da lógica fuzzy são combinados. Estes sistemas são chamados de redes neurais nebulosas [8]. Esta abordagem vem sendo usada em modelos de previsão de carga [4] e previsão de vazões [5], [10], mostrando resultados promissores. Neste contexto, um modelo de rede neural nebulosa vem sendo aplicado ao problema de previsão de vazões médias mensais das usinas hidroelétricas brasileiras localizadas no Rio Paranapanema.

O modelo foi implementado em um programa computacional com recursos avançados para as fases de treinamento e previsão com redes neuro-fuzzy. Esta interface, além de ajudar no ajuste do modelo, também auxilia na visualização dos resultados através de gráficos ou tabelas de todo o histórico ou de períodos específicos. O programa foi implementado através do ambiente de desenvolvimento C++ Builder, da Borland.

Neste trabalho, o modelo de rede neural nebulosa juntamente com o algoritmo de aprendizagem será descrito na próxima seção. A Seção 3 apresenta a interface implementada. A Seção 4 descreve a aplicação da metodologia proposta para previsão de vazões médias mensais para o posto de Jurumirim. Finalmente, na Seção 5 são apresentadas as conclusões.

Este trabalho tem o apoio financeiro da Duke Energy International, Geração Paranapanema.

M. Cicogna, M. Kadowaki, são doutorandos na Faculdade de Engenharia Elétrica e de Computação na Universidade Estadual de Campinas e-mails: (makoto, marcelo)@densis.fee.unicamp.br.

R. Ballini é professora no Instituto de Economia na Universidade Estadual de Campinas (e-mail: ballini@eco.unicamp.br).

S. Soares é professor na Faculdade de Engenharia Elétrica e de Computação na Universidade Estadual de Campinas (e-mail: dino@densis.fee.unicamp.br).

## II. REDE NEURAL NEBULOSA

Esta seção apresenta a estrutura da rede neural nebulosa (RNN) desenvolvida por [11]. A estrutura da rede neural nebulosa (RNN) foi desenvolvida com base em duas características essenciais: o mapeamento das regras nebulosas na estrutura é direto e o processamento neural deve equivaler totalmente a um mecanismo de inferência nebulosa. Essas propriedades são condições necessárias para garantir que o sistema proposto tenha uma natureza dual, ou seja, o sistema pode ser visto ora como um sistema baseado em regras nebulosas, ora como uma rede neural nebulosa [8].

A rede é baseada no mecanismo de inferência nebulosa, codificando uma base de regras na forma de “Se um conjunto de condições é satisfeito Então um conjunto de consequentes é inferido”, o qual pode ser definido da seguinte forma:

Entradas:  $x_1 \text{ é } A_1 \text{ e } \dots x_M \text{ é } A_M$

Se  $x_1 \text{ é } A_1^i \text{ e } \dots x_M \text{ é } A_M^i$  Então  $y \text{ é } w_i$

Regras: Se  $x_1 \text{ é } A_1^i \text{ e } \dots x_M \text{ é } A_M^i$  Então  $y \text{ é } w_i$

Se  $x_1 \text{ é } A_1^N \text{ e } \dots x_M \text{ é } A_M^N$  Então  $y \text{ é } w_N$

Saída:  $y = y(x)$

onde  $x_j, j=1, \dots, M$ , é uma variável nebulosa e  $M$  o número de entradas;  $A_j$  e  $A_j^i, i=1, \dots, N$ , são conjuntos nebulosos representados por funções de pertinência sendo  $N$  o número de regras nebulosas;  $y$  é um número real definido no espaço de saída. Os valores de  $w_i(x) = w_i$ , são números reais e representam os pesos sinápticos correspondentes aos consequentes das regras nebulosas, sendo que, para cada vetor de entrada  $x = (x_1, \dots, x_j, \dots, x_M)$ , está associado um único peso sináptico  $w_i$ .

Para a implementação do mecanismo de inferência nebulosa foi utilizado o método proposto por [9], onde a saída  $y$  é determinada a partir de três estágios de processamento, como segue:

1. *Comparação*: Para cada regra  $i, i = 1, \dots, N$ , e para cada antecedente  $j = 1, \dots, M$ , calcula-se a medida de possibilidade  $P_j^i$  entre os conjuntos nebulosos e, da seguinte forma:

$$P_j^i(x) = S_k \{ T(A_j(x), A_j^i(x)) \} \quad (1)$$

onde  $S$  e  $T$  são operadores lógicos, correspondentes a uma  $s$ -norma e uma  $t$ -norma, respectivamente, e  $x$  é o vetor de entrada.

2. *Agregação dos Antecedentes*: Para cada regra  $i, i = 1, \dots, N$ , agregam-se os valores de cada um dos antecedentes através de uma  $t$ -norma. O resultado desta agregação é chamado de nível de ativação, definido por:

$$H^i(x) = T_j \{ P_j^i(x) \} \quad (2)$$

3. *Agregação das Regras*: A saída  $y$  do sistema de inferência, ou seja, a etapa de defuzzificação dos dados, é realizada através de uma média ponderada entre os conse-

qüentes e os níveis de ativação calculados em (2) para cada regra, dada por:

$$y(x) = \frac{\sum_{i=1}^N H^i(x) w_i}{\sum_{i=1}^N H^i(x)} \quad (3)$$

A RNN é constituída por unidades de processamento, chamadas de neurônios nebulosos. Matematicamente, este neurônio pode ser representado da seguinte forma:

$$y(x) = \varphi(\Psi(x)) = \varphi(\phi(\psi_1(x_1), \dots, \psi_M(x_M)))$$

onde  $\phi$  é o operador sináptico,  $\psi$  é o operador de agregação e  $\varphi$  é a função de ativação.

O modelo RNN apresenta uma arquitetura não recorrente com 5 camadas, como mostra a Figura 1. Pode-se verificar que o mecanismo de inferência descrito através dos procedimentos de comparação, agregação dos antecedentes e agregação das regras é isomórfico à estrutura da rede.

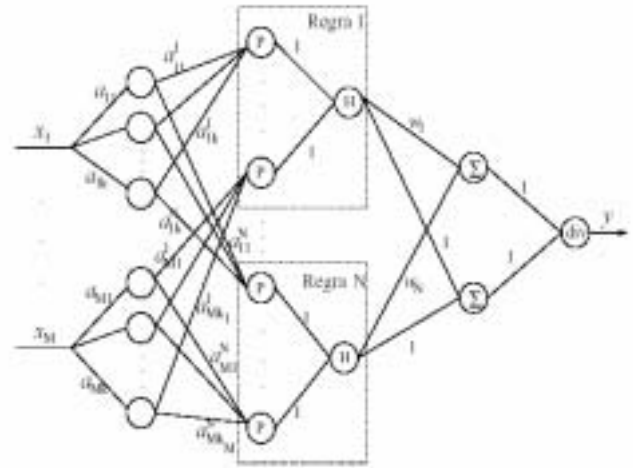


FIGURA 1 - Rede Neural Nebulosa.

A primeira camada é dividida em  $M$  grupos de neurônios, cada um associado a uma variável nebulosa de entrada. Cada neurônio nesta camada representa um intervalo de discretização, correspondente ao espaço de entrada. Ou seja, há tantos neurônios nesta camada quantos forem os intervalos de discretização. Cada grupo de neurônios é responsável por transformar as entradas não nebulosas para que as camadas seguintes possam tratar a informação adequadamente. Estes grupos geram os valores referentes à função de pertinência do conjunto nebuloso. Assim, um neurônio de entrada recebe um sinal simples, decodifica e o transmite para a segunda camada. O sinal é transmitido pelo  $k$ -ésimo neurônio localizado no  $j$ -ésimo grupo.

Definindo-se  $y$  e  $f$  como funções identidade, a saída é dada por  $y = f(x)$ . Para um intervalo  $[x_{jk}, x_{jk+1}]$ , a função decodificada  $j(\cdot)$  para este neurônio, através do  $k$ -ésimo intervalo, é dada por:

$$a_{jk} = \varphi(x_j) = \begin{cases} 1, & \text{se } x_j \in (x_{jk}, x_{jk+1}] \\ 0, & \text{caso contrário} \end{cases}$$

A segunda camada contém  $N$  grupos (número de regras pré-definido), cada qual contendo  $M$  neurônios (número de antecedentes de cada regra). Esta camada repre-

senta o primeiro estágio de inferência chamada comparação. O  $j$ -ésimo neurônio do  $i$ -ésimo grupo representa, através dos pesos sinápticos, a função de pertinência do conjunto nebuloso. Portanto, o  $i$ -ésimo grupo é composto por neurônios que representam as funções de pertinência de todos os conjuntos nebulosos correspondentes aos antecedentes da  $i$ -ésima regra. Além disso, cada neurônio da  $i$ -ésima regra calcula a comparação entre os conjuntos nebulosos  $e$  e  $s$ . Assim, o  $k$ -ésimo neurônio do  $j$ -ésimo grupo da primeira camada, cuja saída é dada por  $y$ , se conecta com o  $i$ -ésimo grupo da segunda camada através de pesos sinápticos  $w_{ij}$ . Considerando-se que  $y = t$ -norma,  $f = s$ -norma e  $j$  é a função identidade, a saída do  $j$ -ésimo neurônio é a medida de possibilidade dada pela equação (1).

A terceira camada contém  $N$  neurônios, cada um com  $M$  entradas. Para cada grupo  $i$  da segunda camada, um neurônio na terceira camada realiza agregação dos antecedentes (segundo estágio de inferência). Todas as conexões possuem peso sináptico unitário.

A saída desta camada é dada pela equação (2), assumindo que  $y =$  função identidade,  $f = t$ -norma e  $j =$  função identidade.

A quarta camada possui dois neurônios, ambos com  $N$  entradas. Ou seja, cada neurônio  $i$  da terceira camada está conectado com os dois neurônios da quarta camada. O operador de agregação associado com estes neurônios é a soma algébrica.

Um dos neurônios conecta-se com o  $i$ -ésimo neurônio da camada anterior através das sinapses, com pesos  $w_{ij}$ . Os operadores  $y$  e  $f$  são, respectivamente, o operador algébrico e a soma algébrica, e  $j$  é a função identidade. As saídas são constituídas pelo numerador da equação (3).

O outro neurônio também é conectado com todos os neurônios da camada anterior com pesos sinápticos unitários. Considerando  $y =$  função identidade,  $f =$  soma algébrica e  $j =$  função identidade, a saída equivale ao denominador da inferência nebulosa definida em (3).

Finalmente, a última camada consiste de um único neurônio para calcular o quociente dos sinais de entrada, ou seja, para calcular  $y(x)$  (equação (3)). Portanto, as conexões possuem pesos unitários e os operadores  $y$ ,  $f$  e  $j$  são função identidade, operador de divisão e a função identidade, respectivamente.

Cada uma das camadas da RNN possui uma associação com os diversos estágios da inferência nebulosa, descritos de (1) a (3). Além disso, os pesos sinápticos representam tanto o conhecimento da rede neural como também representam as funções de pertinência dos antecedentes e consequentes das regras nebulosas. Assim, a RNN codifica um conjunto de  $N$  regras em sua estrutura, processa as informações através de neurocomputação.

#### A. Algoritmo de Aprendizado

A estratégia de aprendizagem da RNN se divide em duas fases. A primeira fase corresponde à aprendizagem

não supervisionada, ou seja, a aprendizagem se processa sem que as saídas sejam fornecidas, envolvendo somente uma parte da arquitetura, mais especificamente a segunda camada. Nesta fase, a rede aprende as funções de pertinência dos antecedentes das regras nebulosas enquanto ajusta os pesos sinápticos associados a esta camada. O método utilizado durante esta etapa é do tipo competitivo, baseado nas redes do tipo Kohonen. A segunda fase adota um método supervisionado, baseado no método do gradiente para aprendizagem dos consequentes das regras, e envolve apenas o ajuste dos pesos da quarta camada.

Assim, a RNN deve aprender os pesos sinápticos relacionados aos neurônios da segunda camada, bem como os pesos sinápticos da quarta camada. O algoritmo de aprendizagem consiste na apresentação dos pares de entrada/saída à rede, sendo o  $t$ -ésimo par dado por  $(x(t), y(t))$ , onde  $x(t)$  é o vetor de entrada com  $M$  coordenadas e  $y(t)$  é a saída desejada.

Considera-se que os conjuntos nebulosos,  $e$  e  $s$ , são representados por funções de pertinência do mesmo tipo, simétricas e completamente definidas pelos valores modais e de dispersão. Por exemplo, funções do tipo Gaussianas ou funções triangulares.

Determinando-se o valor modal e de dispersão de cada função de pertinência e definidos os intervalos de discretização, é possível determinar os pesos sinápticos na forma:

Os valores modal e de dispersão de cada função devem ser aprendidos durante o treinamento, segundo o método de aprendizado competitivo, para determinar os pesos sinápticos da segunda camada.

A função erro que determina o índice de desempenho do modelo, necessária à aplicação do método baseado no gradiente descendente, é definida por:

$$E(x(t)) = \frac{1}{2} (y_d(t) - y(t))^2 \quad (4)$$

onde  $y(t)$  é a saída da rede em relação à entrada  $x(t)$ , dada pela equação (3).

O objetivo do processo de treinamento é ajustar os parâmetros (valor modal, dispersão e pesos) da rede, minimizando  $E(x(t))$ . Desta forma, o ajuste dos pesos sinápticos é proporcional à direção oposta ao gradiente e é dado da seguinte forma:

$$w_i(t+1) = w_i(t) - \eta \frac{\partial E(x(t))}{\partial w_i(t)} \quad (5)$$

onde  $\eta$  é a taxa de aprendizagem.

O algoritmo de aprendizagem capacita a RNN a adquirir novos conhecimentos codificando novas regras nebulosas em sua estrutura, alterando a arquitetura da rede. Portanto, o número de regras nebulosas codificadas pela rede, correspondente ao número de grupos de neurônios na segunda camada, não é constante, ou seja,  $N=N(t)$ .

A estrutura da rede é alterada basicamente devido a dois motivos: o desempenho desejado não é satisfeito pela rede ou todos os grupos neurais da segunda camada são

tais que o nível de ativação é igual a zero. Nestes casos, a rede acrescenta novos elementos à sua estrutura de maneira a codificar uma nova regra nebulosa. Portanto, os grupos de neurônios da segunda camada competem entre si sempre que a rede recebe uma nova entrada  $x(t)$ . O grupo do neurônio vencedor guia os procedimentos de ajuste dos pesos sinápticos.

### III. A INTERFACE GRÁFICA

Neste item apresenta-se alguns exemplos de um sistema computacional de apoio à decisão para o planejamento energético de sistemas hidrotérmicos de potência, enfocando a interface gráfica de seu módulo de previsão de vazões, denominadas HydroLab e HydroPrev, respectivamente.

O sistema de suporte à decisão fornece ao módulo de previsão acesso aos dados de 86 usinas brasileiras, aquelas participantes do Sistema Interligado Nacional – SIN. Através desta base de dados pode-se realizar previsões para todos os postos de vazões das usinas pertencentes ao SIN.

A interface gráfica é orientada a estudos, permitindo que o usuário crie estudos de treinamento de redes e estes elementos são gerenciados pelo sistema HydroLab. Dessa forma, é possível visualizar várias vezes os estudos criados no passado, realizar comparações e descartar estudos. Um exemplo desta ferramenta pode ser visualizada na Figura 2.

O programa HydroPrev corresponde a uma categoria de modelos dentro do programa *HydroLab* e se divide em dois módulos:

1. *Rede Neuro-Fuzzy*: módulo para o treinamento de redes neuro-fuzzy. Para facilitar este processo e diminuir o provável número de estudos manipulados pelo usuário, um estudo de treinamento pode configurar até doze redes, uma para cada mês do histórico de vazões.
2. *Previsão RNF*: módulo para a realização de previsões utilizando redes neuro-fuzzy treinadas no item anterior. Com este módulo pode-se realizar previsões sobre o histórico de vazões para medir a acuidade do modelo ajustado, ou determinar vazões futuras.



FIGURA 2 - Um estudo de treinamento de Redes Neuro-Fuzzy.

Após a configuração de um estudo de treinamento, o módulo *HydroPrev* disponibiliza ao usuário uma coleção de ferramentas gráficas para a análise da qualidade do treinamento. Estas ferramentas permitem concluir, rapidamente, se as redes neuro-fuzzy adquiriram a capacidade de reproduzir o comportamento das séries de vazões em estudo.

Na Figura 3, pode-se notar a comparação entre as séries de vazões real e prevista para um período que engloba a cheia de 1982/1983 registrada para o posto de Jurumirim. O gráfico localizado na parte inferior da mesma figura mostra a trajetória de erros do treinamento.

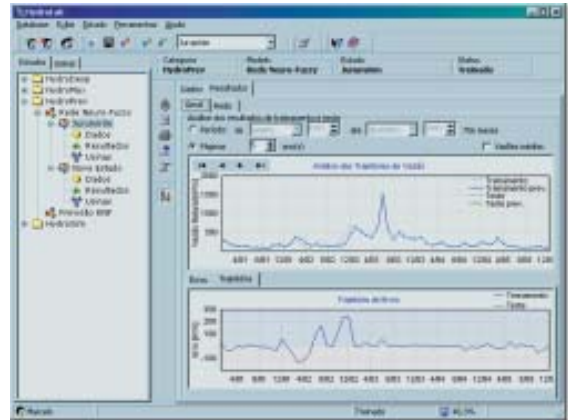


FIGURA 3 - Ferramentas gráficas para análise dos resultados.

O sistema possui ferramentas para a integração com programas como editores de texto e planilha eletrônicas. Estas ferramentas permitem a troca de dados entre o programa *HydroLab* e os demais softwares, conforme exemplo abaixo de um gráfico de vazões retirado do estudo de Jurumirim.

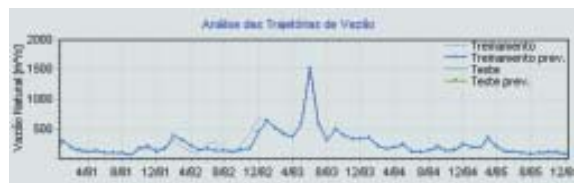


FIGURA 4 - Exemplo de exportação de dados.

Os sistemas computacionais apresentados neste trabalho colocam-se como alternativas concretas para o suprimento de necessidades reais do setor elétrico brasileiro, no que tangem a unificação de modelos matemáticos e de base de dados.

### IV. RESULTADOS

Como forma de ilustrar melhor o funcionamento da rede e de seu algoritmo de aprendizagem, realiza-se o ajuste para os dados de vazões médias mensais do posto de Jurumirim, para o mês de Julho. Para os padrões de treinamento é considerado o período de 1931 a 1995.

Os dados de vazões foram padronizados para o intervalo  $[0, 1]$ . Esta padronização é realizada para simplificar a entrada na rede pois, desta forma, pode-se considerar que a primeira camada da RNN contém  $M$  grupos, sendo  $M$  igual ao número de entradas, e cada grupo contém 1 neurônio.

Consideramos que o vetor de entrada  $x(t)$  contém 6 componentes representados pelos dados de vazões  $t-1$ ,  $t-2$ ,  $t-3$ ,  $t-11$ ,  $t-12$  e  $t-13$ . A seleção dessas entradas foi devido ao fato de fornecer ao modelo a tendência da série dada pelos três meses imediatamente anteriores ( $t-1$ ,  $t-2$ ,  $t-3$ ) e também informações sobre a sazonalidade representada pelos três meses  $t-11$ ,  $t-12$ ,  $t-13$ .

Para as funções de pertinência das regras nebulosas codificadas na RNN foram selecionadas as funções Gaussianas, definidas como segue:

$$F_j^i(x_j) = \begin{cases} 0, & \text{se } |x_j - c_j^i| > \rho^i \\ \exp(-|x_j - c_j^i| / \rho^i), & \text{caso contrário} \end{cases}$$

A taxa de aprendizagem para atualização dos pesos foi fixada em  $h = 0.5$  e o parâmetro  $e = 1.0$  foi utilizado para atualização do fator de aprendizagem. Os neurônios na terceira camada possuem como operador de agregação a *t-norma* produto.

A inicialização da dispersão das funções de pertinência foi adotada como sendo um valor aleatório pertencente ao intervalo  $[0, 1]$ . O número de regras nebulosas iniciais foi  $N(0) = 1$ . Os grupos de neurônios foram adicionados à estrutura da rede para um desempenho  $d = 0,1$  para os meses de janeiro, fevereiro e março. Para os demais meses novos grupos foram adicionados à estrutura da rede para um desempenho  $d = 0,09$ . A tabela 1 mostra o número de regras ajustadas para cada mês.

O modelo foi ajustado sob o período de 1931 a 1988 e, a validação do modelo foi feita no período de 1989 a 1993. Para a previsão foi considerado o período de 1994 a 1998. As figuras 2 e 3 mostram o desempenho do modelo na fase de treinamento, teste e previsão.

**TABELA 1**  
**Número de Regras Ajustadas**

Mês	Nº de Regras
Janeiro	32
Fevereiro	37
Março	33
Abril	36
Mai	20
Junho	10
Julho	25
Agosto	26
Setembro	23
Outubro	49
Novembro	33
Dezembro	32

## V. CONCLUSÕES

Este trabalho apresentou uma classe de rede neural nebulosa para previsão de vazões naturais médias mensais. A estrutura do modelo utilizado é adaptativa e um algoritmo de treinamento baseado no paradigma de aprendizado competitivo foi utilizado, onde grupos de neurônios são adicionados à estrutura da rede sempre que novos conhecimentos são necessários ou quando o desempenho não for satisfatório. Além disso, este trabalho apresentou uma interface a qual auxilia no ajuste e visualização dos resultados. O ajuste foi feito para os dados da usina hidroelétrica de Jurumirim.

## VI. REFERÊNCIAS BIBLIOGRÁFICAS

Periódicos:

- [1] A. I. MecLeod, "Diagnostic Checking of Periodic Autoregression", *Journal of Time Series Analysis*, vol. 15, no. 2 pp. 221-223, 1994.
- [2] A. S. Weigend and D. E. Rumelhart and B. A., "Predicting the Future: A connectionist approach," *International Journal Neural System.*, vol 1, 1990.
- [3] G. Lachtermacher and J. D. Fuller. "Backpropagation in Time-series Forecasting", *Journal of Forecasting*, vol 14, no. 4, pp381-393, 1995.
- [4] A. G. Bakirtzis and J. B. Theocharis and S. J. Kiartzis and K. J. Satsios, "Short Term Load Forecasting Using Fuzzy Neural Networks", *IEEE Transactions on Power Systems*, vol 10, no. 3, pp. 1518-1524, 1995.
- [5] R. Ballini and M. Figueiredo and S. Soares and M. Andrade and F. Gomide, "A Seasonal Streamflow Forecasting Model Using Neurofuzzy Network", *Information, Uncertainty and Fusion*, B. Bouchon- Meunier and R. R. Yager and L. Zadeh, Eds., Kluwer Academic Publishers, pp. 257-276, 2000.

Livros:

- [6] J. Kacprzyk and M. Fedrizzi, "Fuzzy regression Analysis. Studies in Fuzziness", Physica-Verlag, Heidelberg, 1992.
- [7] Box, G. E. P. e Jenkins, G. M. "Time Series Analysis, Forecasting and Control", Holden Day, San Francisco, 1976.
- [8] W. Pedrycz and F. Gomide, "An Introduction to Fuzzy Sets: Analysis and Design," MIT Press, Cambridge, MA, 1998.
- [9] Yager, R. e Filev, D., "Essentials of Fuzzy Modeling and Control", Wiley Interscience, New York, 1994.

Artigos em Anais de Conferências (Publicados):

- [10] R. Ballini and S. Soares and M. G. Andrade, "Seasonal Streamflow Forecasting Via a Neural Fuzzy System," *Proceedings of International Federation of Automatic Control*, vol k, pp. 81-86. 1999.
- [11] Figueiredo, M. e Gomide, F., "Fuzzy System Design Using Neurofuzzy Networks", *Proceedings of the 7th Information Processing and Management of Uncertainty on Knowledge - Based Systems - IPMU'98*, vol. 2, pp. 1416 - 1422, 1998.