Anais do IX Congresso Brasileiro de Redes Neurais / Inteligência Computacional (IX CBRN)
Ouro Preto 25-28 de Outubro de 2009 ©Sociedade Brasileira de Redes Neurais

PREDIÇÃO DE TRÁFEGO DE REDES UTILIZANDO UM MODELO FUZZY COM FUNÇÕES DE BASE ORTONORMAIS BASEADAS EM PROPRIEDADES MULTIFRACTAIS

FLÁVIO H. T. VIEIRA, FLÁVIO G. C. ROCHA

Escola de Engenharia Elétrica e de Computação (EEEC), Universidade Federal de Goiás.
Av. Universitária, n. 1488 - Qd 86 - Bloco A - 3º piso 74605-010 - Setor Leste Universitário - Goiânia - Goiás
E-mails: flavio@eeec.ufg.br, flavio.geraldo@gmail.com

LEE L. LING, SCHEILA G. GARCEZ

Faculdade de Engenharia Elétrica e de Computação (FEEC), Universidade Estadual de Campinas.
Av. Albert Einstein, 400, Caixa Postal 6101 13083-852, Campinas, SP
Emails: lee@decom.fee.unicamp.br, garcez@dsif.fee.unicamp.br.

Abstract— Efficient traffic prediction algorithms are important tools to provide resource allocation and congestion control in computer networks. Accordingly, the prediction step can be adjusted in order to enable the network to have time to obtain and allocate resources and thus ensure the desired quality of service. In this article, we present an adaptive predictive model Fuzzy-FBO that considers the multifractal behavior of network traffic flows. To this end, we model the traffic flows using orthonormal basis functions obtained through information from the multifractal analysis of these flows. Then, we insert the orthonormal basis functions into an adaptive fuzzy model trained with an algorithm we call ARFA (Adaptive Regressive Fuzzy Grouping). Comparisons with other predictive schemes prove the efficiency of the proposed model.

Keywords— Traffic prediction, Multifractal traffic, Orthonormal basis functions, Fuzzy Logic.

Resumo— Algoritmos eficientes de predição da taxa de tráfego são ferramentas importantes em esquemas para antecipar as ações de alocação de recursos e controle de congestionamento em redes de computadores. Neste sentido, o passo de predição pode ser ajustado de forma a possibilitar que a rede tenha tempo para obtenção e alocação dos recursos necessários e assim garantir a qualidade de serviço desejada. Neste artigo, apresentamos um modelo preditivo adaptativo Fuzzy-FBO que leva em consideração o comportamento multifractal dos fluxos de tráfego de redes. Para tanto, modelamos os fluxos de tráfego através de funções de base ortonormais obtidas através de informações provindas da análise multifractal desses fluxos. Em seguida, inserimos as funções de base ortonormal em um modelo fuzzy adaptativo treinado com um algoritmo que denominamos de ARFA (Agrupamento Regressivo Fuzzy Adaptativo). Comparações com outros esquemas preditivos comprovam a eficiência do modelo proposto.

Palavras-chave— Predição de Tráfego, Tráfego Multifractal, Funções de Base Ortonormais, Lógica Fuzzy.

1 Introdução

Na busca de uma descrição mais completa do tráfego de redes, modelos multifractais têm sido empregados [1] [2]. Processos multifractais apresentam além de dependência de longo prazo, diferentes leis de escala [3]. A dependência de longo prazo, presente em séries de tráfego e que tem importante impacto no desempenho de redes [4], pode ser constatada pelo decaimento lento da função de autocorrelação dessas séries temporais. Assim, utilizaremos neste artigo uma expressão analítica para a função de autocorrelação de fluxos de tráfego multifractal que será aplicada na modelagem *fuzzy* proposta.

Algumas propostas de alocação de recursos para fluxos de tráfego baseadas em predição realizadas por redes neurais e *fuzzy* merecem atenção pois mostram que o controle preditivo de tráfego pode ser bastante eficiente, uma vez que o mesmo tenta reduzir o congestionamento antes que este aconteça e se adapta bem às variações bruscas do tráfego de redes [5][6]. Entre as propostas de controle de congestionamento que não dependem de mecanismos específicos de rede, destacamos o esquema apresentado por Adas et al. [7], que usa o algoritmo adaptativo LMS [8] para a predição de tráfego de redes e estas predições são usadas como taxas exigidas pelos fluxos. Em [9][10], outro trabalho nesse sentido, os autores utilizam o algoritmo RLS que apresenta convergência mais rápida.

Algoritmos de controle adaptativos são mais adequados para aplicações multimídia em tempo real nas redes atuais devido ao processamento 'on-line' das informações [11]. Tendo em vista isso, apresentamos uma nova modelagem *fuzzy* com funções de base ortonormais (FBO) que através do algoritmo proposto ARFA (Agrupamento Regressivo Fuzzy Adaptativo) cria adaptativamente agrupamentos *fuzzy* à medida que dados de tráfego de entrada são disponibilizados, sendo capaz de prever eficientemente o tráfego de redes.

2 Funções de Base Ortonormais

Nesta seção, abordaremos alguns conceitos envolvendo funções de base ortonormais e sua relação com a modelagem *fuzzy*.

A base de Laguerre é usada em vários contextos de identificação e controle de sistemas não-lineares [12] [13]. Neste trabalho, adotamos a base de Laguerre especialmente porque esta é completamente parametrizada por um único pólo, o pólo de Laguerre p. O conjunto de funções de transferência associadas a esta base é dada pela seguinte equação:

$$\Phi_{mag,j}(q^{-1}) = \sqrt{1-p^2}\,\frac{q^{-1}(q^{-1}-p)^{j-1}}{(1-pq^{-1})^j}, \quad j=1,\dots,n \quad (1)$$

onde $p \in \{P : -1 < p < 1\}$ é o pólo das funções de Laguerre (base de Laguerre) e onde q^{-j} é o operador de deslocamento. Pode-se notar que fazendo $p=0$ em (1), resulta na base $\Phi_{mag,j}(q^{-1}) = q^{-j}$. Portanto, a base $\Phi_{mag,j}(q^{-1}) = q^{-j}$ é um caso especial da base de Laguerre.

A saída de um modelo entrada-saída pode ser escrita como:

$$y(k) = H(l_1(k),\dots,l_n(k)) \quad (2)$$

onde $l_j(k) = \Phi_{mag,j}(q^{-1})u(k)$ é a *j*-ésima função de Laguerre no instante de tempo k, n é o número de funções de base utilizadas, $u(k)$ é o sinal de entrada e H é um operador não-linear. Note que a operação não-linear correspondente a H pode ser realizada através da modelagem *fuzzy*.

As funções de Laguerre $l_j(k)$ são recursivas e podem ser obtidas por equações de estado da seguinte forma [12]:

$$\mathbf{l}(k+1) = A\mathbf{l}(k) + b\mathbf{u}(k) \quad (3)$$

$$y(k) = H(\mathbf{l}(k)) \quad (4)$$

onde $\mathbf{l}(k) = [l_1(k)\dots l_n(k)]^T$. A matriz A e o vetor b dependem da ordem do modelo n e do valor do pólo p como segue [14]:

$$A = \begin{pmatrix} p & 0 & 0 & \dots & 0 \\ 1-p^2 & p & 0 & \dots & 0 \\ (-p)(1-p^2) & 1-p^2 & p & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (-p)^{n-2}(1-p^2) & (-p)^{n-3}(1-p^2) & (-p)^{n-4}(1-p^2) & \dots & p \end{pmatrix} \quad (5)$$

$$b = \sqrt{(1-p^2)}[1 \quad -p \quad (-p)^2 \quad \dots \quad (-p)^{n-1}]^T \quad (6)$$

O modelo não-linear representado pelas equações (3) e (4) consiste de um mapeamento linear entre a entrada $u(k)$ e as funções de Laguerre $l_j(k)$, mais o mapeamento entre $l_j(k)$ e a saída do sistema $y(k)$. Obtém-se dessa forma, um modelo mais preciso do que o modelo NLMA (*Non-Linear Moving Average*) com o mesmo número de funções [14] [15].

Dado um número de funções de Laguerre n (ordem do modelo), a estimação de um valor adequado para o pólo p da base ortonormal resulta em uma melhor representação do sistema a ser modelado. Objetivamos introduzir informações a respeito do comportamento multifractal dos processos de tráfego no cálculo do pólo p. Assim, introduziremos, brevemente, o comportamento multifractal de processos estocásticos na próxima seção.

3 Análise Multifractal

3.1 Tráfego Multifractal de Redes

O tráfego de redes ao ser considerado multifractal significa que possui uma estrutura de forte dependência inerente entre as amostras, com incidência de rajadas em várias escalas [16][17]. Estas características podem degradar o desempenho de rede em relação a fluxos de tráfego Gaussianos e de curta-dependência [18].

A partir das propriedades da modelagem multifractal, pode-se obter a função de autocorrelação de um processo multifractal de forma analítica.

Seja o processo multifractal $X(n)$ discreto no tempo com parâmetros α, ρ e γ. A função de autocorrelação deste processo para os instantes de tempo n e k, é dada pela seguinte equação [19][20]:

$$E[X(n),X(n+k)] = e^{2\rho+\gamma^2}\left(\frac{\alpha(\alpha+1)^{N-1}}{(\alpha+1/2)^N}k^{-\log_2\left(\frac{\alpha+1}{\alpha+1/2}\right)}\right). \quad (7)$$

onde $N = \log_2(N_a)$ e N_a é a quantidade de amostras do processo.

3.2 Pólo do Modelo Obtido a Partir da Função de Autocorrelação para Processos Multifractais

Nesta seção, introduzimos uma expressão analítica para o pólo p utilizado no cálculo das funções de base ortonormais.

Proposição 1: O pólo p utilizado para o cálculo das funções de base ortonormais para o modelo fuzzy-FBO pode ser dado por:

$$p = -\frac{1}{2^{\log_2\left(\frac{\alpha+1}{\alpha+1/2}\right)}}. \quad (8)$$

Prova: Ver Apêndice A

Note que uma vez que o pólo p é dado em função do parâmetro multifractal α, podemos estimá-lo adaptativamente.

4 Modelo Fuzzy-FBO

A estimação do valor do pólo permite que estendamos a interpolação *fuzzy* de modelos locais, que é a idéia central do modelo TSK [21], a um contexto de funções de base ortonormais. Nossa proposta se baseia em uma versão em espaço de estados do modelo TSK, ou seja, cada regra R^i do modelo *fuzzy* representa um modelo de espaço de estados diferente:

$$R^i: \ Se \ \ l_1(k) \ \ e' \ \ L_1^i, \ \ \ldots \ \ e \ \ l_n(k) \ \ e' \ \ L_n^i$$

$$Então \begin{cases} \mathbf{l}_i(k+1) = A_i \mathbf{l}_i(k) + b_i \vec{x}(k) \\ y_i(k) = H_i(\mathbf{l}_i(k)) \end{cases} \quad (9)$$

onde a matriz A_i e o vetor b_i dependem do pólo $p_i(k)$ e $H_i(\mathbf{l}_i(k))$ é o mapeamento que relaciona a saída $y_i(k)$ do modelo local i a seu correspondente estado de Laguerre (funções de base ortonormais) $\mathbf{l}_i(k) = [l_1(k) \ \ l_2(k) \ \ \ldots \ \ l_n(k)]$, sendo $\vec{x}(k) = [x_1(k) \ \ x_2(k) \ \ \ldots \ \ x_n(k)]$ o vetor de entrada e L_j^i a função de pertinência *fuzzy* para a regra i associada a j-ésima variável de premissa. As variáveis de premissa são os estados de Laguerre do modelo fuzzy-FBO resultante. A saída do modelo fuzzy-FBO é dada por:

$$y(k) = \frac{\sum_{i=1}^{C} y_i(k) w_i(\mathbf{l}_i(k))}{\sum_{i=1}^{C} w_i(\mathbf{l}_i(k))}, \quad (10)$$

onde C é o número de regras (modelos locais) e os pesos $w_i(\mathbf{l}_i(k))$ da regra i são dados por:

$$w_i(\mathbf{l}_i(k)) = \prod_{j=1}^{n} L_j^i(l_j(k)). \quad (11)$$

O modelo fuzzy-FBO pode ser representado conforme as equações (3) e (4) e H dado de acordo com as equações (10) e (11).

4.1 Algoritmo de Treinamento Adaptativo para o Modelo fuzzy-FBO

Nessa seção, propomos um algoritmo de agrupamento regressivo fuzzy adaptativo (ARFA) para o modelo preditivo fuzzy-FBO .

No algoritmo ARFA, levamos em conta a distribuição espacial dos dados considerando o erro de regressão e a distância entre os dados de entrada e os *clusters*. Seja a função custo do algoritmo ARFA definida como:

$$J = \sum_{i=1}^{C} \sum_{k=1}^{N} u_{ik}^2 (r_{ik} d_{ik})^2 \quad (12)$$

sujeito a

$$\sum_{i=1}^{C} u_{ik} = 1, \quad para \quad 1 \le k \le N \quad (13)$$

onde u_{ik} é o grau de ativação da i-ésima regra para o k-ésimo padrão de treinamento, C é o número de regras *fuzzy* e N é o número total de dados de treinamento. Na equação (12), r_{ik} é o erro entre a k-ésima saída desejada $y(k)$ do sistema modelado e a saída da i-ésima regra $f_i(\vec{x}(k); \vec{a}^i(k))$ com a k-ésima entrada, isto é,

$$r_{ik} = y(k) - f_i(\vec{x}(k); \vec{a}^i(k)), \quad (14)$$

com $i = 1,2,...,C$ e $k = 1,2,...,N$. Na mesma equação (12), d_{ik} é a distância entre o vetor de entrada $\vec{x}(k)$ no instante discreto k e o centro do i-ésimo cluster β_i, ou seja,

$$d_{ik} = \vec{x}(k) - \beta_i. \quad (15)$$

Seja $X \in P^{N \times (n+1)}$ uma matriz onde seus elementos são os valores de $\vec{x}(k)$ em sua j-ésima coluna $j = 1, \ldots, n+1$ (a primeira coluna de X é toda composta por 1), $Y \in P^N$ um vetor onde o k-ésimo elemento é o valor de $y(k)$ e $Q_i \in R^{N \times N}$ uma matriz diagonal onde a k-ésima diagonal é dada pelo termo $q(k) = u_{ik}^2 d_{ik}^2$.

Como proposta, tomou-se como solução para a estimação dos parâmetros do modelo, a aplicação de um algoritmo recursivo representado pelas seguintes equações [22]:

$$\begin{aligned} \vec{a}^i(k+1) &= \vec{a}^i(k) + (P_i(k+1) x(k+1) q(k+1)) \times \\ &\times (y(k+1) - x(k+1)^T \vec{a}^i(k)) \end{aligned} \quad (16)$$

e,

$$P(k+1) = P(k) - \frac{q(k+1) P_i(k) x(k+1) x(k+1)^T P_i(k)}{1 + q(k+1) x(k+1)^T P_i(k) x(k+1)}, \quad (17)$$

onde $x(k+1)$ é a $(k+1)$-ésima linha da matriz X e $q(k+1)$ é o $(k+1)$-ésimo elemento da matriz diagonal $Q_i(k+1)$.

Para que se minimize a função custo (12), o grau de ativação u_{ik} da i-ésima regra deve ser dado por:

$$u_{ik} = \frac{1/2(r_{ik}^2 d_{ik}^2)}{\sum_{i=1}^{C} 1/2(r_{ik}^2 d_{ik}^2)}. \quad (18)$$

Da mesma forma, encontra-se a seguinte equação para o centro do i-ésimo cluster (β_i):

$$\beta_i = \frac{\sum_{z=1}^{N} r_{iz}^2 u_{iz}^2 \vec{x}(z)}{\sum_{z=1}^{N} r_{iz}^2 u_{iz}^2}. \quad (19)$$

5 Avaliação de Desempenho de Predição do Modelo Fuzzy-FBO

Utilizamos nas simulações, traços de tráfego TCP/IP (*Transmission Control Protocol/ Internet Protocol*) obtidos da Digital Equipment Corporation (DEC)[1], traços de tráfego Ethernet obtidos da Bellcore[2] e traços capturados entre os anos de 2000 e 2002 na rede Petrobrás através de um analisador de dados DA350 da Acterna^{TM} , com uma resolução de 32 microsegundos [23].

Na presente seção, apresentamos avaliações comparativas entre o desempenho do preditor pro-

[1]http://ita.ee.lbl.gov/html/contrib/DEC-PKT.html
[2]http://ita.ee.lbl.gov/html/contrib/BC.html

posto e o desempenho de outros três diferentes preditores, quando aplicados a traços de tráfego TCP/IP e Ethernet. Os outros preditores levados em consideração foram: o LMS (*Least Mean Square*) [8] [10], o RLS (*Recursive Least Square* [10] e o Fuzzy FCRM (*Fuzzy Clustering Regression Model*) [3] [24]. Utilizamos inicialmente o conceito de erro quadrático médio (EQM) na análise de desempenho de predição. Avaliamos o preditor proposto utilizando-se duas medidas relativas de erro, conhecidos como erros quadráticos médios normalizados (EQMN) [25].

Definição 2: Seja σ_x^2 a variância do processo X, dada por $\sigma_x^2 = E\left[(\mu - x)^2\right]$ onde μ é a média do processo, define-se o erro quadrático médio normalizado do tipo 1 como:

$$EQMN1 = \frac{EQM}{\sigma_x^2} = \frac{E\left[(\hat{x} - x)^2\right]}{E\left[(\mu - x)^2\right]}. \quad (20)$$

Definição 3: Seja $\hat{x}_{pa}$ o valor predito da amostra do processo X, cujo valor é o mesmo da amostra imediatamente anterior do processo. Define-se o erro quadrático médio normalizado do tipo 2 como:

$$EQMN2 = \frac{E\left[(\hat{x} - x)^2\right]}{E\left[(\hat{x}_{pa} - x)^2\right]}. \quad (21)$$

De acordo com as definições anteriores, um preditor que apresente um valor de EQMN1 igual ou inferior a 1 possuirá desempenho igual ou superior a um preditor que apenas estime o valor futuro como sendo igual a média do processo. Para um EQMN2 próximo de 1, teremos que o preditor analisado apresentará desempenho próximo ao de um preditor que estime o valor futuro como sendo igual ao valor da amostra imediatamente anterior.

Tabela 1. Comparação entre EQMN1

Traço de Tráfego	Intervalo	Fuzzy-FBO Adapt.	Fuzzy Adapt.	RLS	LMS	Fuzzy FCRM
Dec-pkt-1	1-2048	0,6564	0,7366	0,8513	0,9304	0,6987
Dec-pkt-2	1-2048	0,5758	0,6704	0,7022	0,7614	0,5836
Bc-Octint	801-1701	0,4102	0,4654	0,4114	0,4817	0,3107
BC-Octext	1000-2000	0,4144	0,4355	0,4298	0,5010	0,4654

Tabela 2. EQMN2: Aumento do n.° de regras=n.° de coeficientes

N° de regras	Fuzzy FBO Adapt.	Fuzzy Adapt.	RLS	LMS
2	0,6121	0,7208	0,9553	1,0779
3	0,5762	0,7114	0,8820	0,9779
4	0,5633	0,7043	0,8739	0,9531
5	0,5846	0,7282	0,8731	0,9378

Um dos objetivos do treinamento adaptativo é o ajuste do algoritmo ao ambiente dinâmico do tráfego de redes. A fim de realizarmos uma comparação com treinamento do tipo ‘on batch’, inserimos também na Tabela 1 os valores de EQMN1 obtidos com o preditor *fuzzy* TSK treinado com o algoritmo não-adaptativo FCRM [24]. O preditor *fuzzy* FCRM se utiliza de todas as amostras da série de tráfego para o cálculo de seus parâmetros. Uma vez determinados estes parâmetros, aplica-se o modelo obtido na predição a um passo da série em questão. Pode-se observar que o preditor *fuzzy*-FBO obteve em geral menor EQMN1 para as séries analisadas. Portanto, os resultados comprovam que se pode ter com o conhecimento de poucas amostras do passado (neste teste, utilizou-se 2 amostras), um erro tão baixo quanto ao processamento com todas as amostras da série temporal. O preditor *fuzzy*-FBO adaptativo proposto captura mais adequadamente as características do processo de tráfego por não supor de antemão que a ‘estrutura’ do processo seja invariante, como é feito por alguns modelos com treinamento ‘offline’. Além disso, ao se comparar o preditor fuzzy-FBO com uma versão sua sem a utilização de funções de base ortonormal (Fuzzy Adapt. na Tabela 1), nota-se também um menor erro de predição para o preditor proposto.

Em teoria, à medida que se aumenta o número de funções de base para o preditor *fuzzy*-FBO, se obtém um EQMN de predição menor para determinada série de tráfego. No entanto, o que se observou para todos os preditores testados é que os EQMN1 e EQMN2 diminuem até um certo número de coeficientes, depois disso, nem sempre é possível obter um treinamento eficiente para os modelos preditores. A Tabela 2 corrobora esta afirmação, onde para efeito de simplificação, adotamos o número de regras como sendo igual ao número de coeficientes (igual ao número de amostras passadas) dos modelos para predição da série dec-pkt-1. Com 5 regras, o preditor *fuzzy*-FBO começa a apresentar uma deterioração dos EQMN1 e EQMN2. Para os algoritmos RLS e LMS, o mesmo ocorre com um número de coeficientes igual a 7. Note, entretanto que, mesmo esses dois algoritmos estando em sua melhor configuração não propiciaram EQMN menor do que o modelo *fuzzy*-FBO com 2 regras e 2 coeficientes.

Conclusão

As características dos fluxos de tráfego nas redes atuais como dependência de longo prazo e rajadas em múltiplas escalas tornam a modelagem e predição de tráfego tarefas difíceis e desafiadoras. Neste trabalho, foi proposto um modelo fuzzy-FBO cujo algoritmo de treinamento adaptativo permite que a predição em tempo real do tráfego de redes seja realizada com um número reduzido de regras nebulosas.

A fim de se obter funções de base ortonormais para o modelo *fuzzy* em questão através do cálculo do pólo do sistema, utilizamos uma expressão analítica para a função de autocorrelação de processos multifractais que é adequada para descrever várias características do tráfego de redes [19]. Em seguida, introduzimos um procedimento para o cálculo do pólo dominante, o qual é utilizado como pólo de Laguerre para o modelo *fuzzy*-FBO. Comprovamos que há uma melhoria de desempenho de predição do modelo, ou seja, predições mais precisas e robustas são obtidas com o acréscimo das funções de base ortonormais. As comparações realizadas com outros preditores adaptativos, demonstraram na maioria dos casos, um desempenho superior de predição do modelo *fuzzy*-FBO proposto para diferentes horizontes de predição e número de regras consideradas. Constatou-se esse fato por meio do erro quadrático médio normalizado e do teste de hipótese T aplicado a séries de erros obtidos com os preditores comparados.

Referências Bibliográficas

[1] S. Molnar and G. Terdik. "A general fractal model of internet traffic". In IEEE LCN 2001, Tampa, Florida, Novembro 2001.

[2] M. S. Crouse e R. G.Baraniuk V. J. Ribeiro, R. H. Riedi. "Multiscale queueing analysis of long-range dependent traffic". Proc. IEEE INFOCOM, vol.2, pp. 1026-1035, Março 2000.

[3] I. W. C. Lee and A. O. Fapojuwo. "Stochastic processes for computer network traffic modelling". Computer Communications, vol. 29, pp.1-23, Março 2005.

[4] M. Grossglauser e J.-C. Bolot. "On the relevance of long-range dependence in network traffic". IEEE/ACM Transactions on Networking, vol. 7, no.5, pp.629-640, Outubro 1999.

[5] Y. C. Ouyang, C.-W. Yang e W. S. Lian. "Neural networks based variable bit rate traffic prediction for traffic control using multiple leaky bucket". Journal of High Speed Networks. vol. 15, no.2, pp.11-122, 2006.

[6] V. A. Aquino e J. A. Barria. "Multiresolution FIR neural-network-based learning algorithm applied to network traffic prediction". IEEE Transactions on Systems, Man and Cybernetics-C, vol. 36, no.2, pp.208-220, Março 2006.

[7] A. M. Adas. "Using adaptive linear prediction to support real-time VBR video under RCBR network service model". IEEE/ACM Trans. Net., vol. 6, no.5, pp.635-645, Outubro 1998.

[8] B. U. Toreyin, M. Trocan e A. E. Cetin. "LMS based adaptive prediction for scalable video coding". In ICASSP, Toulouse, França, 14 a 19 de Maio 2006.

[9] S. Chong, S.-Q.Li e J. Ghosh. "Predictive dynamic bandwidth allocation for efficient transport of real-time VBR video over ATM". IEEE JSAC, vol. 13, no.1, pp.12-23, Jan. 1995.

[10] S. S. Haykin. Modern Filters. Macmillan Publishing Company, New York, 1989.

[11] H.-H Liu and P.-L. Hsu. "Design and simulation of adaptive fuzzy control on the traffic network". In International Joint Conference SICE-ICASE, pp. 4961-4966, Outubro 2006.

[12] G. H. C. Oliveira, R. J. G. B.Campello e W.C.Amaral. "Fuzzy models within orthonormal basis function framework". In IEEE International Fuzzy Systems Conference Proceedings, Seoul, Korea, 22-25 Agosto 1999.

[13] G. A. Dumont e Y. Fu. "Non-linear adaptive control via Laguerre expansion of Voltera kernels". Int. J. Adaptive Control and Signal Processing, vol. 7, pp.367-382, 1993.

[14] B. Ninness e F. Gustafsson. Orthonormal bases for system identification. In Proc. of 3rd European Control Conference, pp. 13-18, Roma, Itália, Setembro 1995.

[15] B. Wahlberg e L. Ljung. "Hard frequency-domain model error bounds from least-squares like identification techniques". IEEE Trans. on Automatic Control, vol. 37, no.7, pp.900-912, Julho 1992.

[16] A. Feldmann, A. C. Gilbert e W. Willinger. "Data networks as cascades: Investigating the multifractal nature of Internet WAN traffic". pp. 25-38. ACM/SIGCOMM'98, Vancouver, 1998.

[17] R. H. Riedi, M. S. Crouse, V. J. Ribeiro e R. G. Baraniuk. "A multifractal wavelet model with application to network traffic". IEEE Trans. on Information Theory, vol. 45, no.3, pp. 992–1018, Abril 1999.

[18] K.Park e W. Willinger. Self-similar Network Traffic and Performance Evaluation. John Wiley and Sons, New York, 2000.

[19] T. D. Dang., S. Molnar e I. Maricza. "Capturing the complete characteristics of multifractal network traffic". In Globecom 2002, Taipei, Taiwan, Novembro 2002.

[20] T. D. Dang; S. Molnar e I. Maricza. "Queuing performance estimation for general multifractal traffic". Int. J. Commun. Syst., vol. 16, no.2, pp.117-136, 2003.

[21] T. Takagi e M. Sugeno. "Fuzzy identification of systems and its applications to modeling and control". IEEE Trans. Syst., Man, Cybern., vol. 15, pp.116-132, Jan. 1985.

[22] P. Young. *Recursive Estimation and Time Series Analysis: An Introduction*. Communications and Control Engineering Series. Springer-Verlag, New York, 1984.

[23] Lee Luan Ling et al. A computational tool and optimization methods for multimedia traffic characterization and effective bandwidth estimation on modern communication networks. Relatório Técnico. Projeto Ericson UNI-20, Laboratório de Reconhecimento de Padrões e Redes de Comunicações (LRPRC) -Unicamp, Março 2002.

[24] F. H. T. Vieira e L.L. Lee. "Fuzzy modeling and prediction with confidence bound estimation for traffic rate allocation in high-speed networks". *Simpósio Brasileiro de Redes Neurais*, São Luís, 29 de Setembro a 01 de outubro 2004.

[25] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, New York, 3rd edition, 1991.

Apêndice A

Seja $X(t)$ um processo cuja função de autocorrelação é representada por $r_x(k) = E[X(t+k)X(t)]$ e Γ_{j+1} $(j = 1,\ldots,n)$, o coeficiente de reflexão para encontrar um modelo AR (AutoRegressivo) de ordem n [10]. Os valores dos coeficientes $a_{n+1}(j)$ do modelo AR podem ser calculados a partir de $a_n(j)$ de tal forma que [10]:

$$r_x(k) + \sum_{l=1}^{p} a_p(l) r_x(k-l) = 0; \quad k = 1,2,\ldots,p \tag{22}$$

Assim, o pólo de um modelo com $(j = 1)$ é dado por [10]:

$$\Gamma_{j+1} = -\frac{\gamma_j}{\varepsilon_j} \tag{23}$$

onde ε_{j+1} é o erro de modelagem.

Inserindo a equação de autocorrelação (7) em (23), e sabendo que $p = \Gamma_1$, temos:

$$p = -\frac{1}{2^{\log_2(\frac{\alpha+1}{\alpha+1/2})}} \tag{24}$$