On the Relevance of Self-Similarity in Network Traffic Prediction

Majid Ghaderi

School of Computer Science

University of Waterloo, Waterloo, Ont N2L 3G1, Canada

ghaderi@uwaterloo.ca

Abstract

Self-similarity is an important characteristic of traffic in high speed networks which can not be captured by traditional traffic models. Traffic predictors based on non-traditional long-memory models are computationally more complex than traditional predictors based on short-memory models. Even on-line estimation of their parameters for actual traffic traces is not trivial work. Based on the observation that the Hurst parameter of real traffic traces rarely exceeds 0.85, which means that real traffic does not exhibit strong long-range dependence, and the fact that infinite history is not possible in practice, we propose to use a simple non-model-based minimum mean square error predictor.

In this paper, we look at the problem of traffic prediction in the presence of self-similarity. We briefly describe a number of short-memory and long-memory stochastic traffic models and talk about non-model-based predictors, particularly minimum mean square error and its normalized version. Numerical results of our experimental comparison between the so-called fractional predictors and the simple minimum mean square error predictor show that this simple method can achieve accuracy within 5% of the best fractional predictor while it is much simpler than any model-based predictor and is easily used in an on-line fashion.

I. Introduction

One of the key issues in measurement-based network control is to predict traffic in the next control time interval based on the online measurements of traffic characteristics. The goal is to forecast future traffic variations as precisely as possible, based on the measured traffic history.

Traffic prediction requires accurate traffic models which can capture the statistical characteristics of actual traffic. Inaccurate models may overestimate or underestimate network traffic.

Recently, there has been a significant change in the understanding of network traffic. It has been found in numerous studies that data traffic in high-speed networks exhibits self-similarity [1]–[6] that can not be captured by classical models, hence self-similar models have been developed. The problem with self-similar models is that they are computationally complex. Their fitting procedure is very time consuming while their parameters can not be estimated based on the on-line measurements.

In this paper, we look at the problem of traffic prediction in the presence of self-similarity. We briefly describe a number of short-memory and long-memory stochastic traffic models and talk about non-model-based predictors, particularly minimum mean square error and its normalized version. This study aims to compare the so-called fractional predictors with the simple minimum mean square error predictor based on the following criteria:

1) **Accuracy**: the most important criteria in choosing a predictor is the quality of its predictions, since the goal of the predictor is to closely model the future.
2) **Simplicity**: in order to achieve a real-time predictor a certain level of simplicity is necessary. Simplicity has an intrinsic value due to ease of use and implementation.
3) **On-line**: most traffic modeling has been done for off-line data. In reality, e.g., network control, we want to use on-line measurements to forecast the future. We do not know any thing about the underlying traffic structure instead we should estimate the predictor parameters based on the on-line measurements.
4) **Adaptability**: a good predictor should adapt to changing traffic. As time progresses, more samples are available. Therefore more is known about the traffic characteristics. An adaptive traffic predictor should use new information to improve itself and update its parameters.

This study is motivated by the observation that the Hurst parameter, referred to by $0 < H < 1$ as the long-range dependence indicator, of real traffic traces rarely exceeds 0.85 [2], [4], [7], [8], which means that real traffic does not exhibit strong long-range dependence, and the fact that infinite history is not possible in practice. We propose to use a simple non-model-based minimum mean square error predictor instead of complex long-memory predictors for on-line predictions. Similar work has been done for variable-bit-rate video traffic in ATM networks [5], we want to verify it for Ethernet traffic. Briefly speaking, this study aims to verify the relevance of long-range dependency in real applications of traffic prediction.

The rest of this paper is organized as follows. In section II we discuss the self-similarity concept and introduce short-range and long-range dependencies. Section III introduces widely used short-memory and long-memory stochastic models applicable to traffic modeling. Section IV is dedicated to minimum mean square error predictors. Our experimental results of applying these predictors on the Ethernet traffic trace from Bellcore are presented in section V. Finally, section VI is the conclusion of this article.

II. Self-Similarity

There is evidence that traffic is self-similar and fractal in nature. This can be explained by assuming that network workloads are described by a *power-law* distribution; e.g., file sizes, web object sizes, transfer times, and even user's think times have heavy-tailed distributions which decay according to a power-law distribution [4]. A heavy-tailed distribution has the following form

$$P[X > x] \sim x^{-\alpha}$$

as $x \to \infty$ and $0 < \alpha < 2$.

These heavy-tailed distributions can be explained by *Zipf's law*. According to Zipf's law the degree of popularity is exactly inversely proportional to the rank of popularity. It is believed that every *human information processing* obeys a power-law distribution which leads to the belief that nature has a fractal geometry.

As stated in [2], [4], the self-similarity in high-speed networks such as the Internet can be explained by heavy-tailed distributions and many ON/OFF traffic sources. In other words, many ON/OFF sources with heavy-tailed ON and/or OFF periods result in aggregate self-similar traffic. Human as well as computer sources of traffic, behave as heavy-tailed ON/OFF sources, therefore the resulting traffic in core networks is self-similar.

Intuitively, a process is self-similar if its statistical behavior is independent of the time-scale. This means that averaging over equal periods of time does not change the statistical characteristics of the process.

Let $\{X_t\}, t = 0, 1, 2, \ldots$ be a wide-sense stationary process (covariance stationary) with mean zero and autocorrelation function $\rho_k = \gamma_k/\gamma_0$ at lag k where $\gamma_k = E[X_t X_{t-k}]$ is the autocovariance function. A stochastic process $\{X_t\}$ is stationary if, for each n, the n dimensional joint distribution $\{X_{t_1+\tau}, \ldots, X_{t_n+\tau}\}$ is independent of τ for any set of n times $\{t_1, \ldots, t_n\}$.

For each m, let $\{X_j^{(m)}\}, m = 1, 2, \ldots$ denotes a new time series obtained by averaging the original series $\{X_t\}$ over non-overlapping blocks of size m, i.e.,

$$X_j^{(m)} = \frac{1}{m}(\sum_{l=0}^{m-1} X_{jm+l}) \tag{1}$$

The processes $\{X_j^{(m)}\}$ are also wide sense stationary with mean μ and autocorrelation $\rho_k^{(m)}$. There are different classes of self-similarity:

- *Exact Self-Similar:* the process $\{X_t\}$ is said to be exactly self-similar if $\rho_k^{(m)} = \rho_k$ for all m. In other words, the autocorrelation structure is preserved across different time scales. Fractional Gaussian noise of section III-B is an example of such a process.
- *Asymptotic Self-Similar:* the process $\{X_t\}$ is said to be asymptotically self-similar if $\rho_k^{(m)} \rightarrow \rho_k$, when $m \rightarrow \infty$, e.g., Fractional ARIMA (section III-B) is asymptotically self-similar.
- *Stochastic Self-Similar:* stochastic self-similar processes retain the same statistics over a range of scales, and they satisfy the relation $X_{at} \simeq a^H X_t$ for all $(a > 0)$, where $\simeq$ denotes equality in distribution. This is a very strict form of self-similarity and called *self-similarity with stationary increments*. Process X_t as defined above, is a H-sssi process. Fractional Brownian motion of section III-B is an example of such a process.

A. Short-Range and Long-Range Dependence

Long-range dependence (LRD) can be considered the phenomenon where current observations are significantly correlated to observations farther away in time. This phenomenon is of particular interest to traffic modeling, since it has been discovered that Internet traffic posses long-range dependence. Short-range dependence (SRD) on the other hand, refers to the phenomenon where current observations are not correlated to very old observations. For SRD processes, correlation to previous observations decays to zero very quickly while it remains significant for LRDs even for very old observations.

Let ν and ν_m denote the variance of X_t and $\{X_j^{(m)}\}$, respectively. For large m equation 1 can be approximated by:

$$\nu_m \approx \nu[2\sum_{k=1}^{m}\rho_k]m^{-1} \tag{2}$$

X_t is said to have a short-range dependence [9], if $\Sigma_k \rho_k < \infty$. Equivalently, ν_m decays to zero in proportion to m^{-1}. According to equation 2 this requires that the autocorrelation function of X_t decays exponentially to zero. That is,

$$\rho_k \sim C^k(-1 < C < 1).$$

The process X_t is said to have a long-range dependence [9], if $\Sigma_k \rho_k \to \infty$. Equivalently, ν_m decays at a slower rate than m^{-1}. For example, processes in which $\rho_k \sim k^{-(2-2H)}$ for large k. H $(0 < H < 1)$ is the so-called *Hurst* parameter, which is an important quantity used to characterize the LRD. An interesting characteristic of the correlation structure of a long-range dependent process is that ρ_k obeys the well-known power-law distribution.

The relation between self-similarity and LRD is that if a process $\{X_t\}$ is self-similar with Hurst parameter H, then its increment process $Y_t = X_{t+s} - X_s$ is LRD with parameter H.

III. Stochastic Traffic Models

Traditional traffic models including Markov and Regression models can only capture short-range dependencies in traffic. We refer to these models as short-memory models. Long-memory models are nontraditional models which are capable of capturing long-range dependencies.

A. *Short-Memory Models*

We briefly discuss two classes of models: (1) Markov models, and (2) Regression models.

In Markov modeling, activities of a system can be modeled by a finite number of states. In general, increasing the number of states results in a more accurate model at the expense of increased computational complexity. The Markovian property is the common characteristic of these models; *the next state of the system depends only on the current state*. Markov-type models often result in a complicated structure and many parameters when used to model a long-range dependent or a mixed process [10].

Regression models define explicitly the next random variable in the sequence by previous ones within a specified time window and a moving average of white noise.

Define the lag operator B as $BX_t = X_{t-1}$, where $B^s X_t = X_{t-s}$. Also assume that Δ denotes the difference operator, i.e., $\Delta X_t = X_t - X_{t-1}$, equivalently $\Delta^d = (1-B)^d$ which can be expressed using the binomial expansion:

$$(1-B)^d = \sum_{k=0}^{\infty} \binom{d}{k} (-1)^k B^k$$

where,

$$\binom{d}{k} = \frac{d!}{k!(d-k)!} = \frac{\Gamma(d+1)}{\Gamma(k+1)\Gamma(d-k+1)}$$

We also define polynomials $\phi(B)$ and $\theta(B)$ as follows:

$$\phi(B) = (1 - \phi_1 B - \cdots - \phi_p B^p)$$

$$\theta(B) = (1 - \theta_1 B - \cdots - \theta_q B^q).$$

1) **Autoregressive Model**:

The autoregressive model of order p, denoted as AR(p), has the form

$$\phi(B) X_t = \varepsilon_t$$

where ε_t is white noise (independent identically distributed random variables with mean 0 and variance σ_ε^2). In this model variable X_t is regressed on previous values of itself,

$$X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + \varepsilon_t.$$

AR models can be used to model stationary time series (time series that have a constant mean) and if all the roots of $\phi(B)$ lie outside the unit circle, then it is invertible (can be written in the form $X_t = \phi^{-1}(B)\varepsilon_t$). Autocorrelation of AR(p) is expressed by

$$\rho_k = A_1 G_1^k + \ldots + A_P G_P^k,$$

where $\frac{1}{G_i}$s are the roots of $\phi(B)$.

2) **Autoregressive Moving Average Model**:

An ARMA(p, q) has the form

$$\phi(B) X_t = \theta(B) \varepsilon_t,$$

or equivalently,

$$X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + \varepsilon_t - \cdots - \theta_q \varepsilon_{t-q}.$$

Note that $\theta(B)\varepsilon_t$ is the moving average part of this model. These models have a great flexibility in modeling time series but still they can not model non-stationary time series. In practice, it is frequently true that adequate representation of actual time series can be obtained with models, in which p and q are not greater than 2 and often less than 2.

3) **Autoregressive Integrated Moving Average Model**:

ARIMA(p, d, q) [11] is an extension to ARMA(p, q). It is obtained by allowing the polynomial $\phi(B)$ to have d roots equal to unity. The rest of the roots lie outside the unit circle. ARIMA(p, d, q) has the form

$$\phi(B)\Delta^d X(t) = \theta(B)\varepsilon_t.$$

ARIMA is used to model non-stationary processes. Note that

$$\Delta^d X_t = (1 - B)^d X_t = \phi^{-1}(B)\theta(B)\varepsilon_t,$$

and accordingly,

$$X_t = (1 + B + B^2 + \cdots)^d \phi^{-1}(B)\theta(B)\varepsilon_t.$$

In this expansion X_t is regressed to sum (integration) of infinite noise variables. In some cases it is possible that the original series X_t is not stationary but its increments $X_t - X_{t-d} = (1 - B)^d X_t$ exhibit stationary characteristics. This is the philosophy behind the inclusion of difference operator Δ in this model.

B. Long-Memory Models

In this section we review some long-memory models which are widely used in theory and practice.

1) **Fractional Brownian Motion (fBm)**:

Brownian motion [9] is a stochastic process, denoted by $\mathrm{Bm}_t, t \geq 0$. It is characterized by the property that increments $\mathrm{Bm}_{t_0+t} - \mathrm{Bm}_{t_0}$ are normally distributed with mean 0 and variance $\sigma^2 t$. The fractional Brownian motion fBm_t is a self-similar process with

$1/2 < H < 1$. Fractional Brownian motion differs from the Brownian motion by having increments with variance $\sigma^2 t^{2H}$.

2) **Fractional Gaussian Noise (fGn)**:

Although fBm is useful for theoretical analysis, its increment process (for finite increment τ)

$$\mathrm{fGn}_t = \mathrm{fBm}_{t\tau} - \mathrm{fBm}_{(t-1)\tau},$$

known as fractional Gaussian noise, is often more useful in practice. While fBm is not stationary, fGn is stationary. The autocorrelation function of this process has the form

$$\rho_k = 1/2[(k+1)^{2H} - 2k^{2H} + (k-1)^{2H}]$$

and as $k \to \infty$

$$\rho_k = H(2H-1)k^{(2H-2)}.$$

3) **Fractional ARIMA Model (FARIMA)**:

Fractional ARIMA proposed by Hosking [12] in 1980 is the natural extension of the ARIMA process when we allow real values for parameter d. X_t is a stationary invertible FARIMA(p, d, q) process if:

$$\phi(B)\Delta^d X_t = \theta(B)\varepsilon_t$$

where d is a real number $(-1/2 < d < 1/2)$, and where $\phi(B)$ and $\theta(B)$ are stationary AR and invertible MA polynomials. The relation $H = d + \frac{1}{2}$ holds between d and H. Thus, X_t is a long-memory process if $(0 < d < 1/2)$ and a short-memory process if $d = 0$. This model has been extensively used in network traffic modeling [7].

FARIMA$(0, d, 0)$ is the fundamental form of this process

$$\Delta^d X_t = \varepsilon_t \ .$$

For this basic form

$$\rho_k = \frac{(-d)!(k+d-1)!}{(d-1)!(k-d)!}$$

and as $k \to \infty$

$$\rho_k = \frac{(-d)!}{(d-1)!}k^{2d-1}.$$

4) **Generalized ARMA Model (GARMA)**:

GARMA models [8] are the generalization of all the regression models. They can be used to model both short-range and long-range dependencies in a time series. In addition, they can be used to model the cyclical patterns of a time series with fewer parameters than ARMA models. The GARMA(p,q) model of a process X_t is defined as

$$\phi(B)(1-2\eta B+B^2)^d X_t = \theta(B)\varepsilon_t$$

where $(-1/2 < d < 1/2)$ and $(-1 < \eta < 1)$. The term $(1-2\eta B+B^2)^d$ is the Gegenbauer polynomial which can be expanded using the power series expansion.

C. Fractional Predictors

Let $\{X_t\}$ be the invertible FARIMA(p, d, q) process,

$$\phi(B)\Delta^d X_t = \theta(B)\varepsilon_t \ .$$

Regarding invertibility, we can write

$$\varepsilon_t = \sum_{j=0}^{\infty} \pi_j X_{t-j}$$

where

$$\sum_{j=0}^{\infty} \pi_j B^j = \phi(B)\theta^{-1}(B)(1-B)^d \ .$$

From the theorems on linear prediction [9], a one-step predictor of a FARIMA process is

$$\hat{X}_{t+1} = -\sum_{j=1}^{\infty} \pi_j X_{t-j+1} \ .$$

GARMA is very similar to FARIMA, therefore we extended this method to apply it as GARMA predictor. The only difference is the computation of π_j coefficients which are given by the following equation

$$\sum_{j=0}^{\infty} \pi_j B^j = \phi(B)\theta^{-1}(B)(1-2\eta B+B^2)^d.$$

For fGn we preferred to use the simple mean square error predictor which will be introduced in section IV-A. Of course there is a direct predictor for fGn [13] but it is too complicated and our simulations did not show any improvement for this particular predictor. Note that instead of computing the autocorrelation matrix from observed values, we use the respective autocorrelation function of fGn process as stated in the previous section. Therefore, fGn predictor is the fastest one in our simulations.

IV. Mean Square Error Predictors

Let $\{X_t\}$ denotes a linear stochastic process and suppose that the next value of $\{X_t\}$ can be expressed as a linear combination of current and previous observations. That is

$$X_{t+1} = w_m X_t + \cdots + w_1 X_{t-m+1} + \varepsilon_t$$

where m is the order of regression. Equivalently in matrix form

$$X_{t+1} = \mathbf{W}\mathbf{X}' + \varepsilon_t.$$

As you can see, this is the case for all the regression models and particularly for the FARIMA and GARMA models. In practical applications such as network control that need on-line traffic prediction, we do not have any prior knowledge about the underlying structure of the traffic but it is possible to estimate the weighting constants w_i.

Let $\hat{\mathbf{W}}$ denote the estimated weight vector, then

$$\hat{X}_{t+1} = \hat{\mathbf{W}}\mathbf{X}' + \varepsilon_t$$

where $\hat{X}_{t+1}$ is the predicted value of X_{t+1}.

In the following subsections we will introduce two solutions for this estimation problem. The first solution is based on minimum mean square error (MMSE) which requires matrix inversion and autocorrelation computation while the second solution which is based on recursive linear regression can eliminate these time-consuming computations in the expense of decreased accuracy.

A. Minimum Mean Square Error Predictor

One simple solution to the estimation problem of section IV is *minimum mean square error (MMSE)*, in which a weight vector is derived by minimizing the expected value of squared errors:

$$e_t = X_{t+1} - \hat{X}_{t+1}$$

and

$$E[e_t^2] = E[(X_{t+1} - \hat{X}_{t+1})^2]\ .$$

This is a minimization problem and can be solved by using its derivative equation which leads to the following solution [14]

$$\hat{\mathbf{W}} = \mathbf{\Gamma}\mathbf{G}^{-1}$$

where $\mathbf{G}$ is the autocorrelation matrix and $\mathbf{\Gamma}$ is an autocorrelation vector starting at lag m.

$$\mathbf{G} = \begin{bmatrix} \rho_0 & \rho_1 & \dots & \rho_{m-1} \\ \rho_1 & \rho_0 & \dots & \rho_{m-2} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{m-1} & \rho_{m-2} & \dots & \rho_0 \end{bmatrix}$$

and

$$\mathbf{\Gamma} = [\ \rho_m \quad \dots \quad \rho_1\ ].$$

Autocorrelations ρ_k can be computed by the following relation

$$\rho_k = \frac{1}{m} \sum_{t=k+1}^{m} X_t X_{t-k},$$

where m is the order of MMSE predictor.

MMSE predictor has the benefit that there is no need to know the underlying structure of traffic, therefore it can be used for on-line prediction purposes. Another benefit of using MMSE is the simplicity of implementation. There are only some matrix manipulations which can readily be implemented in hardware and software at a very high speed [15]. There are even some approximation approaches for computing the weight vector $\hat{\mathbf{W}}$ which eliminate matrix inversion and autocorrelation computations [16]. In section IV-B we will discuss a recursive method for calculating $\hat{\mathbf{W}}$.

B. Normalized Minimum Mean Square Error Predictor

The normalized MMSE method is an adaptive and recursive solution to compute weight vector for MMSE. We choose this name as the authors of [17] although this method has been originally called *normalized recursive linear regression* in [16]. NMMSE does not require prior knowledge of the correlation structure of the time series. Therefore, it can be used as an on-line algorithm for forecasting network traffic.

The recursive linear estimator for weight vector $\mathbf{W}$ is as follows

$$\hat{\mathbf{W}}_{t+1} = \hat{\mathbf{W}}_t + \mu \frac{\mathbf{X}}{||\mathbf{X}||^2} e_t,$$

where μ is the adaptation constant and determines the convergence speed. NMMSE is convergent in the mean square error sense if the adaptation constant μ satisfies the following condition [17]

$$0 < \mu < 2.$$

V. Numerical Results

So far, in this paper we have talked about several prediction methods. Briefly speaking, MMSE based methods can be used as on-line predictors while they are very simple and efficient for implementation. Regarding our criteria for comparing traffic predictors, we have not seen anything about the accuracy of these methods in practical traffic forecasting. In this section we investigate this issue.

To do this accuracy comparison, we implemented three fractional predictors, fGn, FARIMA, and GARMA, and three non-fractional predictors, MMSE, NMMSE, and Naive predictor. Naive predictor is a very simple predictor as follow

$$X_{t+1} = X_t,$$

which is an AR(1) predictor.

In this experiment we used an Ethernet traffic trace (`pAug89.TL`[1]) from Bellcore which is collected by Leland et al. [2]. This trace has information on the time-stamp and the packet size of traffic. The data they collected is cumulative. To get a time series, we need a uniform time scale. We extracted the traffic data at 0.01 millisecond intervals. We dedicated the first 2000 samples of this trace to estimate parameters of the fractional models (actually we used the reported parameters in [18] for FARIMA and [8] for GARMA predictors) and then we implemented and used predictors to forecast 20,000 samples into the future.

[1]Accessible at `http://ita.ee.lbl.gov`

TABLE I

PREDICTOR PARAMETERS

Model	Model Parameters
FARIMA$(1, d, 1)$	$\theta_1 = -0.37$, $\phi_1 = -0.17$, $d = 0.3$
GARMA(0,0)	$\eta = 1$, $d = 0.3$
fGn	$H = 0.8$, $\mu = 3.90$, $\sigma^2 = 7.36$

Finally, we used the reverse of *Signal to Noise Ratio* [19] as the accuracy measure to compare these predictors

$$\mathrm{SNR}^{-1} = \frac{\sum e^2}{\sum X^2},$$

the smaller the SNR^{-1}, the more accurate the predictor.

The first experiment investigates the accuracy of each model with respect to the history size which is used in prediction. This is a performance measure because there is a direct relation between the size of history and the amount of computation required by predictor.

Simulation results show that for this traffic trace the best history size is 100 samples for both FARIMA and GARMA predictors. Theoretically, as the size of the history increases the accuracy of long-memory models must increase but in practice, increasing the history size beyond a limit has a marginal effect on the accuracy while it increases the computation time. In practice you can not calculate infinite series of coefficients π_j because of the finite precision of floating point operations and numerical errors introduced in this computation. An important observation is that optimal history size for both MMSE and NMMSE is about 20 samples which is much smaller than the optimal history size of FARIMA and GARMA. For fGn, as the size of the history increases, the accuracy also increases, as expected. fGn is a pure long-range dependent process, which means that it can not capture short-range dependencies. Therefore, as the history size increases its forecasts become more accurate. This predictor does not suffer from the impreciseness introduced by numerical errors because of the simplicity of the predictor. Again, accuracy improvement by longer history size is very marginal.

In the second experiment, we investigated the effect of cumulative errors for optimal history size. We used the optimal history size from the first experiment and predicted 20,000 samples in future, step by step. In other words, in each time interval we used one-step predictors to predict

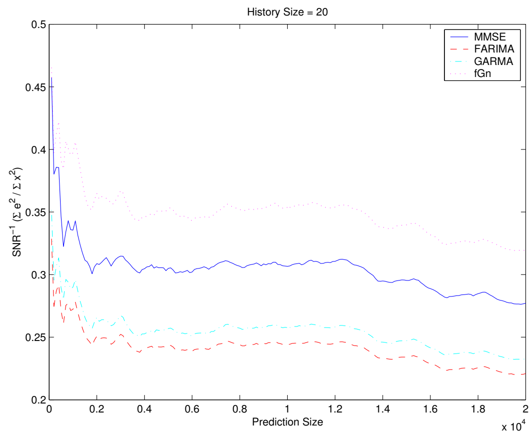

Fig. 1. All Predictors Accuracy Comparison

the next value and so on. This experiment confirms that SNR^{-1} goes to zero as we go farther into the future. This is a very important feature and shows the rate of adaptability of predictors to the traffic.

Figure 1 depicts the accuracy of all the predictors, when the history size is 20. As you can see, MMSE is somewhere between fGn and GARMA predictor which is very close to FARIMA predictor. As we said earlier, fGn is a pure long-memory process. Therefore, its accuracy is clearly weak for history size 20. The interesting observation is the difference between accuracy of complex off-line predictor FARIMA and simple on-line predictor MMSE:

$$\mathrm{SNR}^{-1}_{(\mathrm{FARIMA})} - \mathrm{SNR}^{-1}_{(\mathrm{MMSE})} \leq \%5$$

Note that we does not mean that long-memory models are useless. Indeed, our comparison is between traffic predictors not traffic models. In MMSE approach there is not any underlying model at all. It means that you can not use MMSE to generate synthesis traffic. There are several applications both in theory and in practice that you need an exact model for the network traffic. In these situations models like FARIMA are the best choice, but if you want a simple on-line predictor with an acceptable performance and accuracy, at least in our experiment, MMSE predictor is the most suitable predictor.

In figures 2 and 3 we have compared the accuracy of MMSE predictor with Naive and normalized MMSE predictors. Although it has been claimed that NMMSE can achieve a good accuracy for VBR video traffic prediction [19], its accuracy is not good for our experiment with

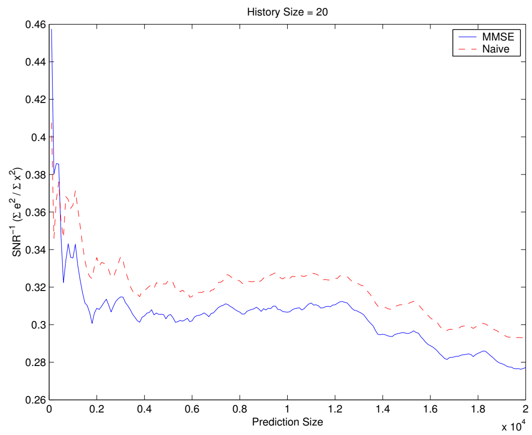

Fig. 2. MMSE vs. Naive

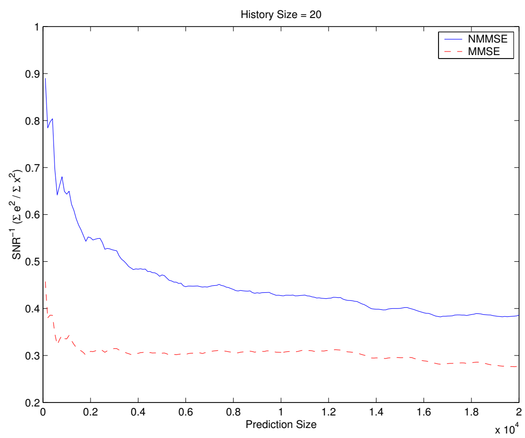

Fig. 3. MMSE vs. NMMSE

Ethernet traffic. As we mentioned earlier in section IV-B, advantage of NMMSE over MMSE is that it eliminates matrix inversion and autocorrelation computations, therefore, it is the fastest predictor among the investigated predictors.

VI. Conclusion

In this paper, we investigated the self-similarity in network traffic. We discussed the concept of short-range and long-range dependencies intuitively and formally. Then stochastic models which are used for traffic modeling, including both short-memory and long-memory models, introduced. While there is certainly much more in the area of stochastic traffic modeling than what we have presented in this paper, we have focused on providing a comprehensive overview of self-similarity and related traffic models.

Based on the observation that the Hurst parameter, H, for real traffic traces rarely exceeds 0.85, we proposed to use more simple non-model-based prediction methods instead of using complex long-memory models. Then we investigated MMSE and NMMSE as on-line traffic predictors. Finally, experimental results of applying these predictors to Ethernet traffic showed the suitability of our proposal for on-line forecasting. Based on our results, accuracy of MMSE predictor is within a 5% of accuracy of FARIMA predictor which is the best fitted fractional model for this traffic trace. An interesting question here is whether we can apply MMSE predictor for Internet traffic.

REFERENCES

[1] W. E. Leland, M. Taque, W. Willinger, and D. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, vol. 2, no. 1, pp. 1–15, Feb. 1994.

[2] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level," *IEEE/ACM Trans. Networking*, vol. 5, no. 1, pp. 71–86, Feb. 1997.

[3] J. Beran *et al.*, "Long-range dependence in variable-bit-rate video traffic," *IEEE Trans. Commun.*, vol. 43, no. 2, pp. 1566–1579, Feb. 1995.

[4] M. E. Crovella and A. Bestavros, "Self-similarity in world wide web traffic: Evidence and possible causes," *IEEE/ACM Trans. Networking*, vol. 5, no. 6, pp. 835–846, Dec. 1997.

[5] M. Grossglauser and J.-C. Bolot, "On the relevance of long-range dependence in network traffic," *IEEE/ACM Trans. Networking*, vol. 7, no. 5, pp. 629–640, Oct. 1999.

[6] S. A. M. Ostring and H. Sirisena, "The influence of long-range dependence on traffic prediction," in *Proc. IEEE ICC'01*, vol. 4, Helsinki, Finland, June 2001, pp. 1000–1005.

[7] Y. Shu, Z. Jin, J. Wang, and O. W. Yang, "Prediction-based admission control using FARIMA models," in *Proc. IEEE ICC'00*, vol. 3, New Orleans, USA, June 2000, pp. 1325–1329.

[8] R. Ramachandran and V. R. Bhethanabotla, "Generalized autoregressive moving average modeling of the Bellcore data," in *Proc. IEEE LCN'00*, Tampa, USA, Nov. 2000, pp. 654–661.

[9] P. J. Brockwell and R. A. Davis, *Time Series: Theory and Methods*, 2nd ed. Springer-Verlag, 1991.

[10] A. T. Andersen and B. F. Nielson, "An application of superpositions of two state markovian sources to the modeling of self-similar behavior," in *Proc. IEEE INFOCOM'97*, vol. 1, Kobe, Japan, Apr. 1997, pp. 196–204.

[11] G. E. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*, 2nd ed. Holden-Day, 1976.

[12] J. R. M. Hosking, "Fractional differencing," *Biometrika*, vol. 83, no. 1, pp. 165–176, Apr. 1981.

[13] G. Gripenberg and I. Norros, "On the prediction of fractional brownian motion," *Journal of Applied Probability*, vol. 33, pp. 400–410, 1996.

[14] P. Whittle, *Prediction and Regulation by Linear Least-Square Methods*, 2nd ed. University of Minnesota Press, 1983.

[15] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. McGraw-Hill, 1994.

[16] A. E. Albert and L. A. Gardner, *Stochastic Approximation and Nonlinear Regression*. MIT Press, 1967.

[17] S. Haykin, *Adaptive Filter Theory*, 2nd ed. Prentice Hall, 1991.

[18] Y. Shu *et al.*, "Traffic prediction using FARIMA models," in *Proc. IEEE ICC'99*, vol. 2, Vancouver, Canada, June 1999, pp. 891–895.

[19] A. M. Adas, "Using adaptive linear prediction to support real-time VBR video under RCBR network service model," *IEEE/ACM Trans. Networking*, vol. 6, no. 5, pp. 635–644, Oct. 1998.