On-line estimation of the parameters of long-range dependence

Matthew Roughan and Darryl Veitch
Software Engineering Research Centre, RMIT University,
Level 2, 723 Swanston St, Carlton, Vic 3053, Australia
E-mail: {matt,darryl}@serc.rmit.edu.au
Patrice Abry
CNRS URA 1325 - Laboratoire de Physique, Ecole Normale Supérieure de Lyon
46, allée d'Italie 69 364 LYON Cedex 07 - France
E-mail: pabry@physique.ens-lyon.fr

Abstract— **An on-line version of the Abry-Veitch wavelet based estimator of the Hurst parameter is presented. It has very low memory and computational requirements and scales naturally to arbitrarily high data rates, enabling its use in real-time applications such as congestion control, and avoiding the need to store huge data sets for off-line analysis. An implementation for Ethernet based on standard hardware supporting sampling rates of 1000 data points per second is described, and results of its operation presented. The performance of the estimator as a function of the length of data processed is demonstrated using simulated data.**

I. Introduction

Real-time traffic measurement is necessary to support network management functions such as call admission control, rate adaptation mechanisms, and congestion control. As such activities must take place on the small time scales implied by the high bandwidth of modern telecommunications systems, the extent of such measurements and the complexity of the algorithms which use them are limited by hard processing constraints, a situation which is unlikely to change. Even in the less demanding case of off-line processing, the ever increasing volume of data represented by collection over a given time interval poses huge storage and processing problems. Such limitations are particularly serious when parameters which are crucial to meaningful traffic characterisation have high computational complexity, say of $O(n^2)$ where n is the length of the data.

In the last few years the discovery of the *self-similar* nature of many kinds of packet traffic [9], [11] has inspired a small revolution in the way that high-speed traffic is viewed. Although no single model is accepted as definitive, the *Hurst* parameter H, which describes the degree of self-similarity, holds a central place in the description of such traffic. Its accurate measurement is therefore of considerable importance for the provision of quality of service as well as for capacity planning. Unfortunately, methods for the estimation of this parameter from data have suffered from poor statistical performance, and/or high computational complexity inappropriate for large data sets or real-time use. Recent work based on wavelets however has provided a semi-parametric estimator for H which gives unbiased estimates together with significant computational advantages, notably a run time complexity of only $O(n)$. Details of this estimator are summarized below and can be found in [7], [6] (see also [1], [2], [3]). The aim of the present paper is to show how these computational advantages can be exploited in an on-line setting to allow H to be estimated in real-time simply, rapidly, and with very low memory requirements. The method scales to arbitrary size with respect to both memory and processing requirements, so that it will remain applicable even as data rates increase with time. It can be applied not only in traditional real-time settings but also, by performing estimation at the point of measurement, as a means to radically reduce the volume of data that needs to be stored from measurements destined for further off-line analysis. This work is the subject of an Australian provisional patent application number PP1692.

II. The Abry-Veitch (AV) estimator

In any data measurement situation a basic theoretical framework is required through which to view the data, to select important parameters which describe it, and to propose and evaluate estimators of them. In our case the time varying *rate* $x(t)$ of incoming traffic is the data of interest, and we model it as a stationary stochastic process. Basic features of this process are its mean $\mu_x = E[x]$, variance $\sigma_x^2 = E[(x-\mu_x)^2]$, and correlation function $\gamma_x(k) = E[(x(t+k)-\mu_x)(x(t)-\mu_x)]$. In this context the self-similar properties of traffic manifest themselves in a particular form of $\gamma_x(k)$, namely a decrease with lag k so slow that the sum of all correlations downstream from any given time instant is always appreciable, even if individually the correlations are small. The past therefore exerts a long term influence on the future, exaggerating the impact of traffic variability and rendering statistical estimation problematic. This phenomenon is known as *Long Range Dependence* (LRD), and is commonly defined as $\gamma_x(k) \sim c_\gamma |k|^{-(1-\alpha)}$, $\alpha \in (0,1)$, or equivalently as the power-law divergence at the origin of its spectrum: $f_x(\nu) \sim c_f |\nu|^{-\alpha}, |\nu| \to 0$. The Hurst parameter describes the (asymptotic) self-similarity of the cumulative traffic process corresponding to $x(t)$ which generates the LRD of $x(t)$, itself described by α. It is nonetheless common practice to speak of H in relation to LRD. The two are related as $H = (1+\alpha)/2$.

In [7], [6] a semi-parametric joint estimator of (α, c_f) is described based on the *Discrete Wavelet Transform* (DWT). Wavelet transforms in general can be understood as a more flexible form of a Fourier transform, where $x(t)$ is transformed, not into a frequency domain, but into a time-scale wavelet domain. The sinusoidal functions of Fourier theory are replaced by wavelet basis functions $\psi_{a,t}(u) \equiv \psi_0(\frac{u-t}{a})/\sqrt{a}$, $a \in I\!R^+$, $t \in I\!R$ generated by simple translations and dilations of the the *mother wavelet* ψ_0, a band pass function with limited spread in both time and frequency. The wavelet transform can thus be thought of as a method of simultaneously observing a time series at a full range of different scales a, whilst retaining the time dimension of the original data. Multiresolution analysis theory shows that no information is lost if we sample the continous wavelet coefficients at a sparse set of points in the time-scale plane known as the *dyadic grid*, defined by $(a,t) = (2^j, 2^j k)$, $j,k \in \mathbb{N}$, leading to the DWT with discrete coefficients $d_x(j,k)$ known as *details*. Henceforth we will deal exclusively with the details of the DWT. The *octave* j is simply the base 2 logarithm of scale $a = 2^j$, and k plays the role of time (although a time whose rate varies with j). For finite data of length n, j will vary from $j = 1$, the finest scale in the data, up to some $j_{\max} \approx \log_2(n)$. The number of coefficients available at octave j is denoted by n_j, and approximately halves with each increase of j.

The estimator has excellent computational properties due to the availability of the fast 'pyramid' or filter-bank algorithm for the calculation of the discrete wavelet transform, which has a complexity of only $O(n)$ [4]. The number of wavelet coefficients $d_x(j,k)$ thus generated is also of order n, and subsequent computations required to form the estimate of H from them have only this complexity. The overall complexity therefore remains $O(n)$, which clearly scales satisfactorily with n.

The main feature of the wavelet approach which makes it so effective for the statistical analysis of scaling phenomenon such as LRD is the fact that the wavelet basis functions themselves possess a scaling property, and therefore constitute an optimal 'co-ordinate system' from which to view such phenomena. The main practical outcome is that the LRD in the time domain representation is reduced to residual **short** range correlation in the wavelet coefficient plane $\{j,k\}$, thus removing entirely the special estimation difficulties. Thus for each fixed j, the series $d_x(j,\cdot)$ can be regarded as a stationary process with weak short range dependence, and across scales the different series can be regarded as independent.

We can now outline the estimator as consisting of the following three stages:

1. **Wavelet decomposition** A discrete wavelet transform of the data is performed, generating the details $d_x(j,k)$ over the dyadic grid.
2. **Detail variance estimation** At each fixed octave j the details are squared then averaged across 'time' k to produce an (excellent) estimate of the variance of the wavelet coefficients, called μ_j. It can be shown that the true value of this variance follows a power-law in j with exponent α.
3. **LRD parameters estimation** A log-log plot is made of $y_j = \log_2(\mu_j)$ against j and from it the range of octaves $[j_1, j_2]$ where scaling occurs is determined. The LRD parameters H and c_f are then extracted by performing a weighted linear regression over those scales.

Notes:

- Since the expectations of the details are all identically zero [1], the average of the squares of the details at a given j is an estimate of the variance at that j.
- In forming y_j small corrective terms $g(j)$ are in fact subtracted from $\log_2(\mu_j)$ to account for the fact that $E[\log](\cdot) \neq \log(E[\cdot])$.
- H is related to the slope of the plot, and c_f to a power of the intercept.
- The weights are the known variances σ_j^2 of the y_j under Gaussian assumptions and are **not** functions of the data.
- Confidence intervals for H are derived from the standard variance formulae for weighted linear regression with mutually independent y_j, and the known σ_j^2, and so again are **not** functions of the data.

An example of the regression fit using a simulated data set is given in *Figure 1.* The 95% confidence intervals for each y_j, shown as vertical lines at each octave j, are seen to increase with j.

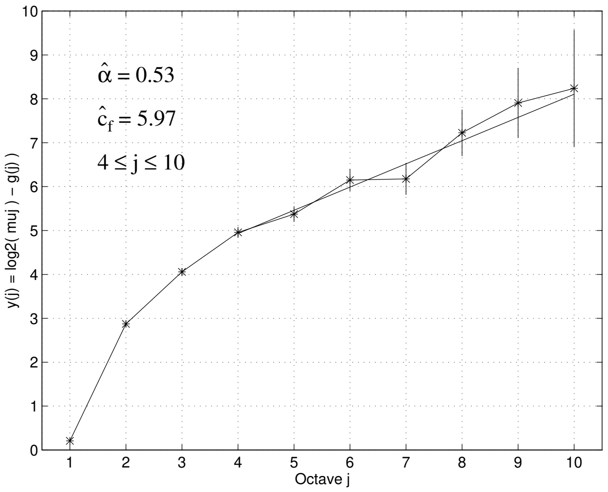


Fig. 1. **Stage 3, the estimation.** An example of the y_j against j logscale diagram and regression line for a LRD process with strong SRD. The series is simulated farima(0,d,2) with $d = 0.25$ ($\alpha = 0.50$) and $\Psi = [-2,-1]$ implying $c_f = 6.38$. Selecting $(j_1, j_2) = (4, 10)$ identifies the relevant scaling range allowing an accurate estimation despite the strong SRD: $\hat{\alpha} = 0.53 \pm 0.07$, $\hat{c}_f = 6.0$ with $4.5 < \hat{c}_f < 7.8$.

III. The on-line estimator

The AV estimator summarized above is gaining acceptance as the method of choice for measuring LRD in traffic [5], [8]. Until now however it has been used as a batch estimator, that is where a data set is collected and analysed off-line. It is ideally suited to on-line use however, making it usable within network elements such as switches as well as network monitoring systems. By on-line estimation we mean a data processing method whereby new fragments of data are processed as they arrive. In what follows we concentrate on the estimation of H, although the second LRD parameter, c_f, is also estimated by the method.

On-line estimation has two main requirements:

1. That an algorithm be devised such that newly acquired data elements can be processed individually and merged with existing processed data, rather than requiring complete re-computation from the base data.
2. That the algorithm be efficient enough to implement the above at the rate that new data arrives.

The first requirement is critical for on-line estimation, whereas the second is an issue of the necessary computing power versus its cost. Because of the steadily increasing bandwidth of networks however, the method must be scalable, so the second requirement is in fact principally an issue of the time and memory complexity of the algorithm.

The AV algorithm can be adapted to satisfy both requirements. The first stage of the estimator, the wavelet decomposition, is easily implemented in an on-line fashion using a real-time pyramidal filter-bank (*Figure 2*). Indeed, such filter-banks were devised with on-line applications in mind. The second stage is trivial and can be performed on-line as follows. Let the current stored sum of squares at octave j calculated from the first n_j values be

$$S_j = \sum_{k=1}^{n_j} d_x(j,k)^2.$$

Assume that the arrival of the new data point $x(n)$ results in the output of a new coefficient $d_x(j, n_j+1)$ at octave j from the filter-bank. The sum is then simply updated as

$$\begin{aligned} n_j &\leftarrow n_j + 1, \\ S_j &\leftarrow S_j + d_x(j, n_j)^2. \end{aligned}$$

When the current variance estimate at octave j is required for the final stage it can be calculated as $\mu_j = S_j/n_j$. The final stage of the estimation algorithm need not be adapted to an on-line version, as there is no need to compute H every time a new data point is acquired. It may be recalculated only as needed, typically at 'human' time-scales several orders of magnitude larger than the data collection rate. In any case the complexity of the final stage is only of $O(\log_2(n))$.

Some explanation is required to explain why the first stage of the on-line estimator is scalable. The on-line filter-bank, illustrated in *Figure 2*, consists of a number of filters of fixed size K connected in series (typically the size of these filters is small, say $K = 6$). Because the output rate of each filter is only half of its input rate, data of length n is effectively summarized and held in the filter-bank in the form of $K \log_2(n)$ 'half-processed' values. These numbers are the only ones which must be stored in memory, not the full set of historical input data $x(t)$. Regarding the run-time complexity, on average each new data point $x(n)$ results in $2(K+1)$ operations, a number independent of n. The maximum possible number of operations scales as $O(\log_2(n))$, however if problems of processor load arise the filter-bank can be naturally implemented in Digital Signal Processing (DSP) hardware.

Section V shows how a quite modest computer is capable of performing the AV estimation algorithm, on-line and in real-time on Ethernet data sampled at 1000 times per second.

The obvious advantage of computing estimates on-line is that results are immediately available, rather than having to wait through a lengthy cycle of collection and analysis. As mentioned earlier, this is essential for real-time

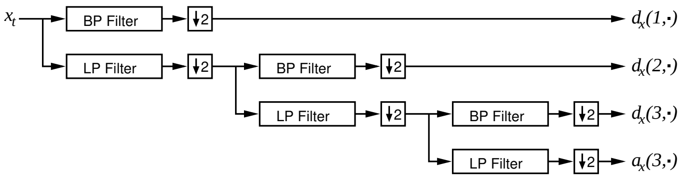


Fig. 2. The filter-bank. At each level in the recursive structure, the Band Pass (BP) output: $d_x(j,\cdot)$, and the Low Pass (LP) output: $a(j,\cdot)$, occur at half the rate of the LP input $a_x(j-1,\cdot)$.

network management purposes, but also offers important advantages for traffic collection and analysis in general. For example, apart from reducing the analysis delay, this approach allows the decision as to whether enough data has been collected to be made as it arrives. It is also advantageous to be able to detect unusual events as they occur, enabling immediate modifications to the collection/analysis effort.

The other central advantage of on-line estimation is the reduction in memory requirements, both in terms of the algorithm itself and of the storage of data sets. Batch analysis requires the collection and analysis of *very* large data sets, and samples larger than any standard computer's memory space are easy to collect. For example, a traditional Ethernet sampled every 1000ms over 1 week represents 604 million sample points, which stored as four byte integers requires approximately 2.4 GB of space. Thus capture of this data may be a problem, as the data cannot all be stored in memory and then saved to disk. Similarly for analysis, the data cannot be held in memory all at the same time resulting in large delays due to disk paging. In contrast, as explained above, on-line measurement does not have substantial memory requirements. Thus a traffic stream can be monitored and measured continuously for weeks at a time, without any delay in the estimation at the end of the process, and without a large memory.

The number of available scales increases with the length n of the data. Ideally the number of available octaves is simply $j_{\max} = \log_2(n)$, however edge effects limit the number in practice. *Figure 3* shows the number of octaves in the data, and the number of octaves actually available, as functions of n.

Note that the on-line algorithm allows all of the scales available in the data to be seen and used, rather than deciding a-priori which scales will be examined.

IV. Performance

The performance of the batch joint AV estimator, and comparisons with other methods of estimating LRD parameters, have been described in detail elsewhere [7], [6] (for H only see also [2], [1], [3]). Briefly, the estimator offers excellent statistical performance: negligible bias, close to optimal variance, and robustness of various kinds including with respect to superimposed deterministic nonstationarities. The aim here is not to repeat these studies but to illustrate the dependence of certain properties on n,

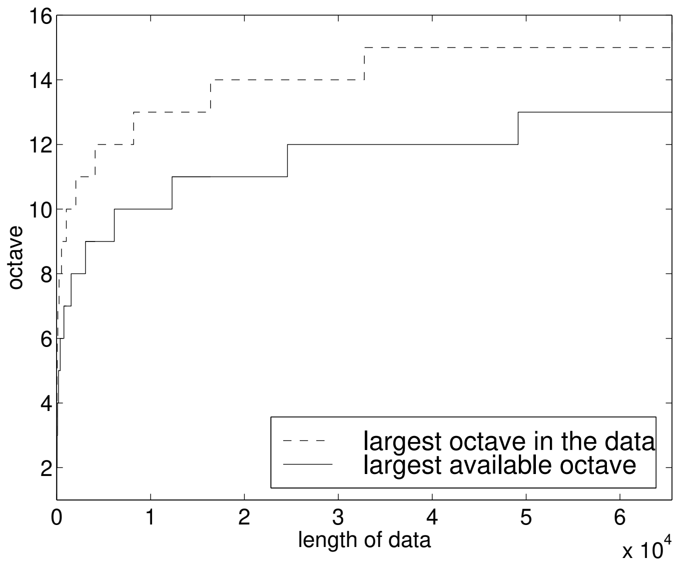


Fig. 3. The number of available scales as a function of n.

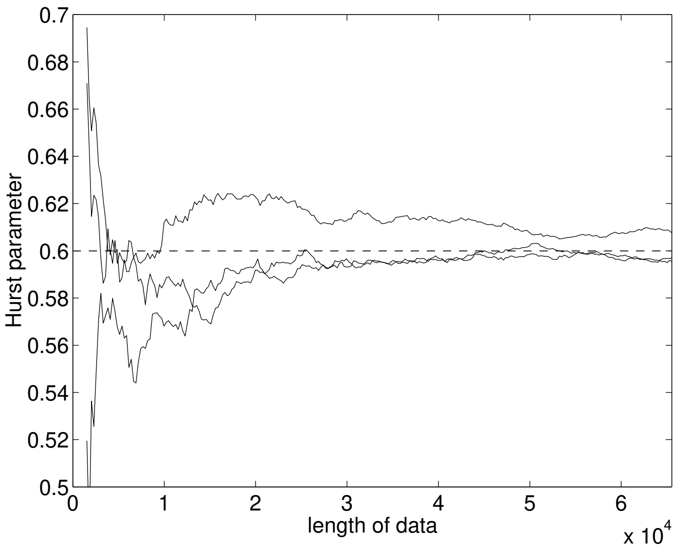


Fig. 4. Three example sample paths. The dashed line shows the true Hurst parameter while the solid lines show three examples of the on-line Hurst parameter estimates for generated fGn.

as the new feature of the on-line version is that the length of the data is constantly increasing.

The series used in this section were all realisations of the fractional Gaussian noise (fGn) process, precalculated using a standard spectral synthesis technique. In each case values from the series were piped to the on-line estimator one at a time, in order to simulate the arrival of raw measurements in real-time. Thus the estimator used here is identical to that used with the working on-line system described in the next section. The interval chosen between the execution of stage 3 of the estimator, the actual estimation of H, was every 2^8 data points. There is a warm up period at the beginning of the measurement run to wait for the octaves required for the analysis to become available (see *Figure 3*).

For each of the values $H = 0.6$, 0.7, 0.8 and 0.9, 100 independent realisations of length $n = 2^{16}$ were generated. *Figure 4* shows three randomly chosen examples from the set with $H = 0.6$. The graph illustrates typical behaviour of the estimator in time. Here as in the other series, we use prior knowledge of the fGn process to choose the lower end of scaling range to be $j_1 = 3$, and the upper end $j_2 = j_{\max}$ to be be the largest octave available. A point of interest is that there is no immediate jump in accuracy when a new octave (scale) becomes available for use in the estimation. This is because when this occurs there are still relatively few data points at the new octave, and so the weighted regression gives little weight to them.

In *Figure 5*, for each H, averages over the set of 100 realisations are plotted. The fact that the averaged estimates tend to the correct values illustrates the lack of bias of the estimator. The speed of convergence to the correct value with increasing n is shown by the shrinking standard deviation of the 100 estimates shown to either side. These sample standard deviations constitute empirical estimates of the standard deviation of the estimator.

To investigate this convergence further, a log-log plot

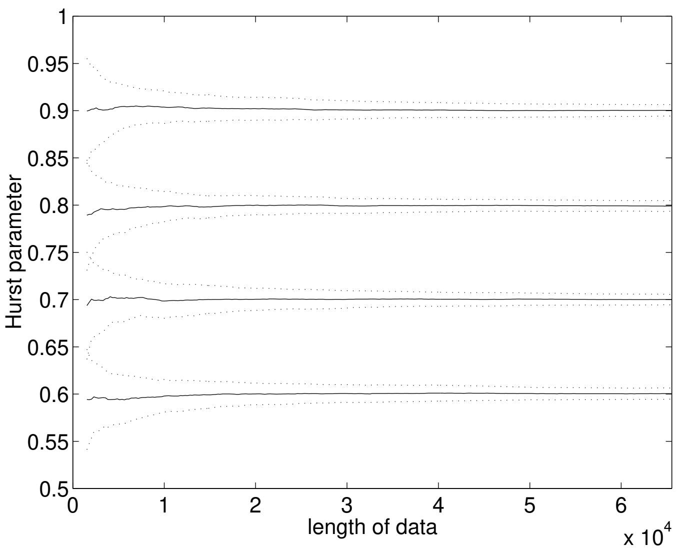


Fig. 5. The average (solid lines) of the estimates of the Hurst parameter, and the standard deviation around the average (dotted lines) for each set of 100 data sequences.

is provided in *Figure 6* of the mean squared error of the estimates as a function of n. The mean squared error corresponds closely to an empirical estimation of the variance of the estimator, as we know the bias to be negligible. The fact that an approximately straight line is seen suggests that the variance of the estimator decreases as a power law. It is also noteworthy that the mean squared errors seem to have very little dependence on the Hurst parameter. Both of these facts are in agreement with the theoretical results of [7] which state that there is no H (nor c_f, nor μ_x, nor σ_x^2) dependence in the variance of the estimation of H, and that the asymptotic variance is

$$\mathrm{var}(\hat{H}(n)) \sim \frac{2^{j_1-3}}{\ln^2 2} n^{-1},$$

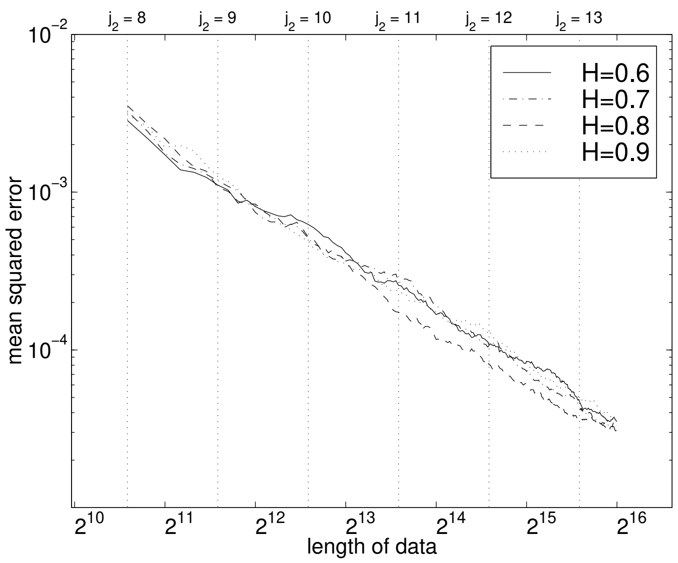


Fig. 6. The mean squared error of the estimated Hurst parameter for each of the four sets of 100 sample paths.

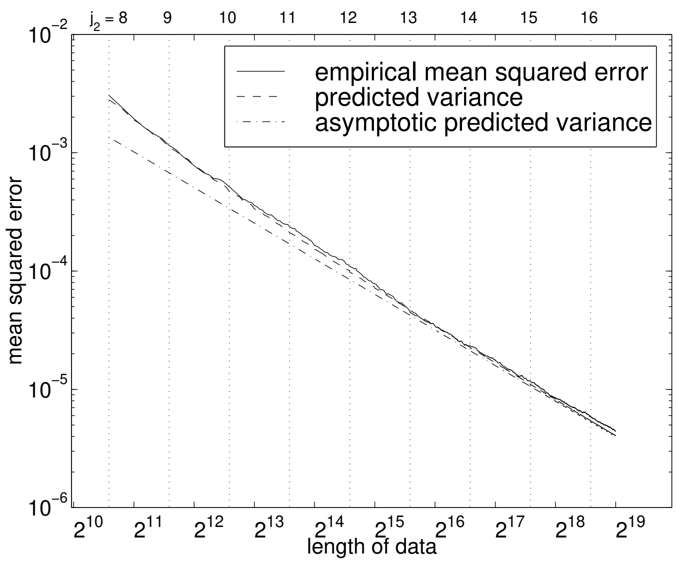


Fig. 7. The dependence of the mean squared error of the estimates on n is seen to agree with the theoretical result, being a $1/n$ decrease.

where again j_1 is the smallest scale used in the estimation. The $1/n$ decrease is the normal rate characteristic of the variance of estimators in a **short** range dependent context (for example independent random variables), vindicating the claim that the AV estimator obtains short range dependent statistics from long-range dependent data. The hypotheses used to obtain the theoretical results are never exactly satisfied in practice however, not even for a 'model' LRD process such as the fGn. We therefore repeated the test for $H = 0.6$, shown in *Figure 7*, this time over 500 realisations each of length 2^{19}, in order to determine the actual decay rate of the mean squared error. In the plot the mean square errors (empirical variances) are compared against the asymptotic theoretical prediction given above, and the full theoretical variance prediction of [7]. Performing a linear regression on the mean square errors in the plot leads to a slope of -1.04, which agrees with the predicted rate of n^{-1}, the minor deviation from -1 being easily accounted for by the asymptotic nature of the $1/n$ dependence together with statistical fluctuations of the empirical variance.

V. Real-time estimation for Ethernet

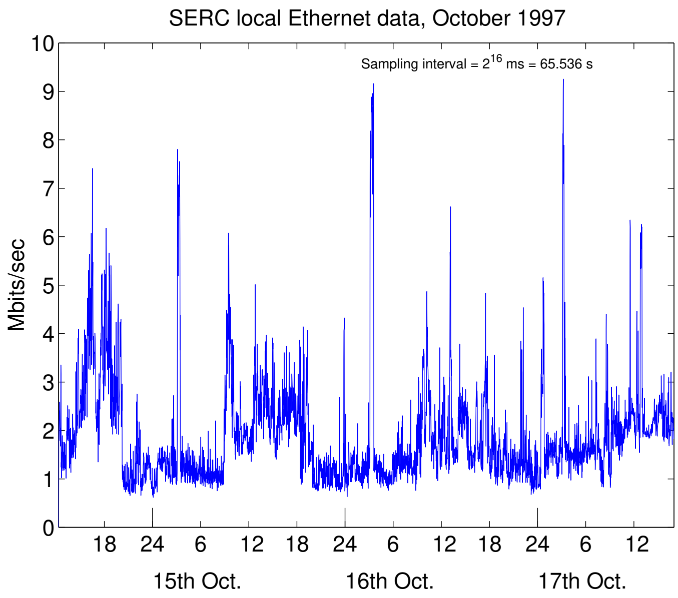
We have explained in section III how the on-line estimator is scalable, and in section IV illustrated a working version using simulated on-line data. In this section we use the estimator to analyse Ethernet data on the local area network at the Software Engineering Research Centre (SERC) without the use of high performance hardware, proving that it is efficient enough to be used on real systems at reasonable cost.

An Ethernet was chosen for two reasons. First it was the first type of data network where self-similar traffic was shown to exist [9]. Second, it is relatively easy to extract traffic from an Ethernet because of the broadcast nature of the medium. We used the Berkely Packet Filters (BPF) which are part of the kernel of `FreeBSD` to capture and time stamp packets. `FreeBSD` is a free implementation of the the `Unix` operating system suitable for Intel PCs - the packet capture and the estimation algorithm were all run on a 133MHz Pentium computer. Though the timestamping of the BPF is not as accurate as that obtained in the original Bellcore study [9], that same study showed that timestamping accuracy of the order of 1ms is quite sufficient to demonstrate self-similarity.

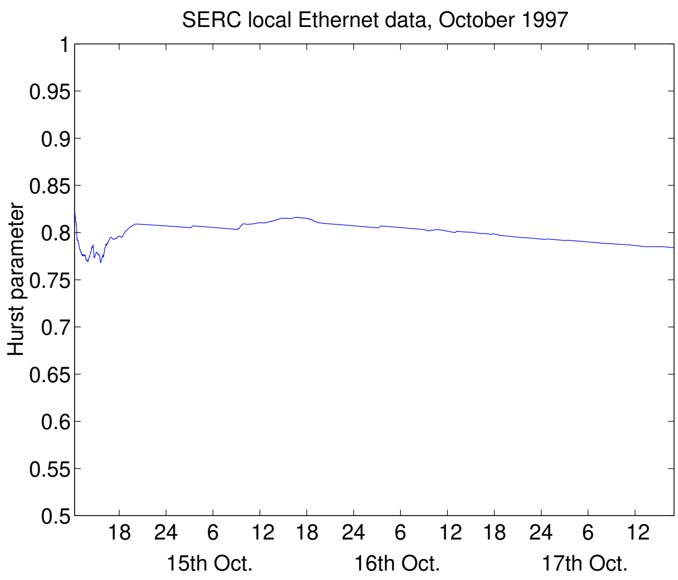
The output from the BPF passes through a simple prefiltering program which generates a sequence of data values corresponding to the number of bytes transferred over the Ethernet during each sampling interval. This sequence is the raw data series $x(t)$ to be analysed by the on-line estimator. A sampling interval of 10ms was generally used, though experiments with intervals as fine as 1ms posed no computational problems for the 133 MHz processor. At 10ms the series is approximately Gaussian, and thus satisfies an important hypothesis upon which confidence intervals are calculated. Octaves $(j_1, j_2) = (8, j_{\max})$ were used in the estimation, based on visual inspection of the logscale diagrams (log-log plots).

Figure 8(a) shows Ethernet data (in Mbps) averaged over approximately one minute intervals (2^{16}ms). *Figure 8*(b) shows the on-line estimation for the same data, based on the finer 10ms sampling. The intent of these figures is not to demonstrate self-similarity in Ethernet traffic as this has already be accomplished. Rather the results are intended to show that the algorithm is efficient and robust enough to apply to real data, in real time.

In higher speed environments such as ATM networks the sampling rate will have to increase in order to monitor the traffic on an adequate range of time-scales, resulting in an increase in processing requirements. It is possible that as network speed increases a point will be reached where those requirements will exceed the capacity of the proces-



(a) Ethernet byte data averaged over minute intervals.



(b) The corresponding on-line estimation session.

Fig. 8.

sor chips available at the time. However in such a case the time critical on-line aspects could be implemented on Digital Signal Processing (DSP) hardware, to which they are ideally suited. Such a solution would be able to cope with any data rates currently envisaged with room to spare.

VI. Conclusion

We have shown that the Abry-Veitch estimator for the measurement of the parameters of long-range dependence, including the Hurst parameter, can be successfully adapted to an on-line context, in real-time. This enables their measurement and use in real-time situations such as network congestion control. Furthermore the immediate analysis of data at the point of measurement avoids the need to store huge data sets for off-line analysis. The scalability of the method was demonstrated both with respect to memory requirements, which are very modest, and processing complexity. The algorithm's performance under ideal conditions was demonstrated by applying it to simulated on-line data, and found to be excellent and in agreement with theoretical results. Estimation output was presented from a working system where on-line, real-time measurements are made of Ethernet traffic using a 133 Mhz processor. Thus the method not only scales but is efficient enough to deal with high data rates on inexpensive hardware.

There is much scope for future work, notably:
1. Methods for discarding old data - windowing and smoothing (Kalman filtering).
2. Effects of non-stationarities of various kinds.
3. Automatic choice of scaling octaves (j_1, j_2).

Other problems with LRD parameter estimation under real traffic conditions are discussed in [10]. Note that the robustness features of the wavelet based analysis can be used effectively to eliminate many of these, including those due to load variations in the network [2].

Acknowledgements

The support of Ericsson Australia is gratefully acknowledged.

References

[1] P.Abry, P.Gonçalvès and P.Flandrin, Wavelets, Spectrum estimation, $1/f$ processes. Wavelets and Statistics, *Lectures Note in Statistics*, Vol.105, pp.15–30, (1995).
[2] P.Abry, D.Veitch, Wavelet Analysis of Long Range Dependent Traffic. *IEEE Trans. on Info. Theory*, Vol.44, no.1 pp.2-15, January (1998).
[3] P.Abry, D.Veitch and P.Flandrin, Long-Range Dependence: revisiting Aggregation with Wavelets. to appear in *Journal of Time Series Analysis*, (1998).
[4] I.Daubechies, Ten Lectures on Wavelets. SIAM, (1992).
[5] A.Feldmann, A.C.Gilbert, W.Willinger, T.G.Kurtz, Looking behind and beyond self-similarity: On scaling phenomena in measured WAN traffic. Preprint, 1997.
[6] D.Veitch, P.Abry, Estimation Conjointe en Ondelette des Paramètres du Phénomène de Dépendance Longue, *16ieme Colloque GRETSI* Grenoble, France, pp.1451–1454, September (1997).
[7] Darryl Veitch and Patrice Abry, "A wavelet based joint estimator of the parameters of long-range dependence," Submitted to *Special Issue of the IEEE Trans. on Info. Theory* , Multiscale Statistical Signal Analysis and its Applications, 1998.
[8] Judith L. Jerkins and Jonathan L. Wang, "A measurement analysis of ATM cell-level aggregate traffic," in *IEEE Global Telecomunications Conference, GLOBECOM'97*, 1997, pp. ???–???
[9] Will E. Leland, Murad S. Taqqu, Walter Willinger, and Daniel V. Wilson, "On the self-similar nature of ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 1–15, Feb 1994.
[10] S. Molnár, A. Vidács, and A.A. Nilson, "Bottlenecks on the way towards fractal characterization of network traffic: Estimation and interpretation of the hurst parameter," in *International Conference on the Performance and Management of Complex Communications Networks, PMCCN'97*, Tsukuba, Japan, Nov 1997.
[11] V.Paxson, S.Floyd, wide-area traffic: the failure of Poisson modelling, *Proceedings of SIGCOMM '94* (1994).