Detection of Hurst Parameter Changes using Data Compression

S. Raj Rajagopalan*
Bellcore

*Joint work with T. Varugis Kurien

Sources with Long Range Dependence

- (Leland Willinger Taqqu Wilson 94) and others showed Ethernet traffic shows *long range dependence*.

- The traffic is modeled as the aggregate of a large number of on-off sources with on periods having long-tailed distributions.

- Network Implication: Queueing behavior for such sources is drastically different from short range dependent models. (e.g. [Erramilli Willinger Narayan 95])

- Extended by others to ATM traffic as well.

Implications of LRD on Network performance

- Buffer sizes.
- Resource allocation strategies.
- Call Admission Policies.
- QoS and reservation strategies.
- Network performance monitoring.

Goal of this work

- **What we have:** The Hurst parameter is the most popular model for the strength of the long-range dependence. (Higher Hurst parameter implies greater long range dependence.)

- **What we want to do:** We want to build an efficient (network) monitor that tells if this has parameter value has changed. In particular, the change detector should be

  - Fast (or computationally lightweight)
  - Sensitive
  - One pass (for network applications)
  - Efficient trade-off between accuracy and speed

LRD and Self-similar processes

- Let $X_0, X_1, X_2, \ldots$ be a covariance-stationary process (stationarity is always assumed).

- Let the autocorrelation function: $r(k) = Cov(X_0, X_k)$ be $r(k) \sim k^{\beta}$ (β constant). (Polynomial decay).

- X is second-order self-similar if the aggregated process $X^{(m)}$ is indistinguishable from X (upto second order).

- $H = 1 - \frac{\beta}{2}$ is called the Hurst parameter.

- Self-similar processes have long range dependence (with the failure of the usual central limit theorem.)

- Leland et. al. found that normal Ethernet traffic has stable H.

A Self-similar process : Fractional Gaussian Noise

$X_0, X_1, X_2, \ldots$ stationary *Gaussian* (μ, σ^2)

Auto-correlation function $r(k) = \frac{1}{2}(|k+1|^{2H} - |k|^{2H} + |k-1|^{2H}) \sim H(2H-1)k^{2H-2}$.

This process is second-order self-similar when $\frac{1}{2} < H < 1$.

Estimating H is a computationally intensive process.

Estimating the Hurst parameter

R/S statistic, periodogram plots, variance-time plots.

For a process $X_1, X_2, ...$

R/S statistic: A ratio of the adjusted range and the a rescaled standard deviation.

Periodogram plot:
$I_X(\omega) = (2\pi n)^{-1} |\sum_j X_j \exp(ij\omega)|^2$.
Slope of periodogram plot near zero on a log-log scale is linearly related to H.

Variance-Time plot: If $X^{(m)}$ formed by block aggregation of X over blocks of size m, then $Var(X^{(m)}) \sim m^{2H-1}$.

Standard way to estimate H for FGN

- An approximate MLE: The Whittle estimator.

- Replace the variance-covariance matrix by an approximation.

- Replace the determinant of the variance-covariance matrix by an approximation.

- Implementation: Calculate the periodogram.

This is very slow.

Data Compression and Statistical Estimation

Intuition: Compression algorithms look for patterns. Sources with correlation must output streams with patterns: A universal predictor may be based on a universal compressor.

- Vitter, Krishnan et al used Data Compression for prediction. (Page Faults).

- Compression-based techniques have been used for Web caching.

Idea: Really random sequences cannot be compressed but if there is correlation in a sequence, an optimal data compressor is likely to compress it *to the maximum extent possible*.

Data Compression: The Move-To-Front Algorithm

- Objects to be compressed: Words over a *finite and known* alphabet.

- Both Sender and Receiver have finite lists with words.

- To send a word W, Sender looks it up in the list. If it is present at position i, Sender transmits i. Receiver looks up the i-th word in his list, and outputs it. Both Sender and Receiver move W to position 1 of their lists.

Analysis of MTF Compression

Let $B_k = \left(X_r, ..., X_{r+k}\right)$ denote a block of data.

Define the compression ratio:

$$\rho_{MTF}(B_k) := \frac{\text{size of } B_k \text{ after compression}}{\text{size of } B_k \text{ before compression}}$$

Theorem [Bentley Sleator Tarjan Wei 86]

$\rho_{MTF}(B_k) \to Entropy_{empirical}(B_k) + \epsilon$ as $k \to$ inf .

The Theorem holds for any source!

What is the relation between Compression ratio and H?

Entropy of FGN

Let X be a fractional Gaussian process with Hurst parameter H.

Then $f(X_1, \ldots, X_n) =$

$$\frac{1}{\sqrt{2\pi}^n |\Sigma(n,H)|^{0.5}} exp(-0.5 x^T \Sigma^{-1} x)$$

where $\Sigma(n,H) =$ variance-covariance matrix.

Entropy $= E(-\log f(X_1, \ldots, X_n))$.

Lemma Entropy of $(X_1, \ldots, X_n)$ is $\frac{n}{2} \log \sqrt{2\pi} + \frac{1}{2} \log(|\Sigma(n,H)|) + \frac{1}{2} E(Z^2)$ (where Z is n-variate indept. normal).

Estimating $|\Sigma|$

Estimation of $\log|\Sigma|$ is done via the spectral density, $f(\lambda, H)$.

Lemma(Grenander, Szego)

$$n^{-1} \ \log(|\boldsymbol{\Sigma}(n,H)|) \ \rightarrow (2\pi)^{-1}\int_{-\pi}^{\pi} \log(f(\lambda,H))d\lambda, \quad \text{as } n \rightarrow \infty.$$

The spectral density, $f(\lambda, H)$ has an explicit representation as an infinite sum.

Theorem (Simple form) Entropy of FGN is a strictly decreasing function of H.

How to make the process fast?

MTF is not as fast as it can be.

[Grinberg Rajagopalan Venkatesan Wei 95] proposed using Splay Trees [Sleator Tarjan 85] and using multiple-lists for MTF and MTFS.

Splay Trees: A self-organizing binary search tree that moves the last accessed item to the root.

Theorem [ST85] If the number of items in the splay tree is m, the total access time for *any* n accesses is at most $3n\log(n) + O(m\log(n))$.

So if m is a constant, the cost per access is about $3\log n$. Moreover, the total access time is independent of the source model!

Putting it all together is very simple

- Block a FGN source with unknown H, with a suitable k.

- Compress k-sized blocks using MTF with splay tree and compute the compression ratio.

- Plot the compression ratio in time and look for sudden changes.

Comparison with other estimators

- Traditional methods of empirical estimation of H are very slow or require multiple passes over the data.
- (Abry Veitch 96) proposed a clever new method to estimate H but it cannot directly handle the case of H varying.
- (Erramilli Wang 97) proposed a method based on sums of squares.
- Our method cannot handle more general LRD sources (yet).
- Compression techniques cannot handle deterministic trends well.

What about short-term correlations?

Effect of Short Range Correlations

- What is the effect on the scheme of adding a short range process to the FGN?

- **Preliminary Answer:** An additive constant to the entropy.

- If we add white noise to the FGN, we need to estimate the entropy of the sum of the FGN and the white noise.

Conclusions

- We have a very fast one-pass method of detecting changes in Hurst parameter of fractional Gaussian processes.

- Using multiple buffer sizes in parallel (in the same pass), we can get multiple sensitivity levels.

- Using calibration, we can use this to estimate Hurst parameter for FGN.

Open Problems

- Higher order properties for the change-point estimator.

- Extend the analysis to a wider class of self-similar processes (ARIMA etc).

- Use Splay Trees directly to get even finer analysis/performance. This can be done for Markov processes, by using the strong Markov property. However, no such property exists for long range dependent processes.