Real-Time Measurement of Long-Range Dependence in ATM Networks

Matthew Roughan\textsuperscript{1}, Darryl Veitch\textsuperscript{1}, Jennifer Yates\textsuperscript{2} Martin Ahsberg\textsuperscript{3}, Hans Elgelid\textsuperscript{3}, Maurice Castro\textsuperscript{3}, Michael Dwyer\textsuperscript{1}, Patrice Abry\textsuperscript{4}

1 - “Emulab”, Dept. of Electrical & Electronic Engineering, University of Melbourne
2 - AT&T Research Laboratories, Florham Park, NJ, USA
3 - SERC, RMIT University, Australia
4 - Laboratoire de Physique, Ecole Normale Supérieure de Lyon, France
Real-Time Measurement, Why?

Measurement

- long term network dimensioning
- medium term capacity planning
- short/medium term capacity management
- informs network control strategies
- input to congestion control – detection, response
- input to call admission control – conservative/efficient
- real-time traffic optimisation – network or application

Real-time measurement

- **essential** for congestion control, CAC
- **essential** for flow control, adaptive applications
- need for economic and ubiquitous **off-line** analysis
  - reduces memory needs and processing times
  - results immediately available
  - can adapt measurement parameters/approach
  - **Disadvantage**: can’t return to change analysis
What to Measure?

It is now recognised that packet traffic is fractal.

This means it is characterised by:

- invariant relationships between scales – self-similarity
- lack of characteristic time scale (in a range(s))
- radical burstiness
- strong dependence on past – no exponential drop off
- non-intuitive statistical properties
- difficult statistical properties – long memory, non-stationary

Implications for networks:

- lower utilisation at a given QoS (say loss rate)
- Buffer insensitivity: larger buffer won’t save us – heavy tailed buffer sizes
- impacts scale dependent – simple averages misleading
- can be generated by heavy tailed file sizes
- but multiplexing gain still possible

So: new parameters to measure, in real-time, and in a difficult environment.
A Fractal Traffic Parameter
Long-Range Dependence (LRD)

Consider a second order stationary process \( X(t) \), representing say bytes per time interval, with

- mean: \( m_X = \mathbb{E}[X(t)] \),
- variance: \( \sigma_X^2 = \mathbb{E}[(X(t) - m_X)^2] \),
- covariance: \( R_X(k) = \mathbb{E}[(X(t) - m_X)(X(t + k) - m_X)] \)

Usual (simple) definition of LRD:

\[
R_X(k) \sim c_r |k|^{-(1-\alpha)}, \quad k \to \infty \quad \alpha \in (0, 1)
\]

Describes the slow decay of covariances found in packet data above 100ms.

Equivalently, power-law divergence of spectrum:

\[
\Gamma_X(\nu) \sim c_f |\nu|^{-\alpha}, \quad |\nu| \to 0
\]

Key parameter is the scaling exponent \( \alpha \)
The Wavelet Based Estimator of LRD

Wavelet bases have inherent scale-invariance - ideal tools for scaling processes.

Use the Discrete Wavelet Transform (DWT), to transform to a time-scale domain.
At each \( \log_2 \) of scale \( j \) have a detail process:
\[
\{ d_X(j, k), \ k = 1, 2, \ldots n_j \}
\]

Statistical benefits:

- detail processes quasi-decorrelated
- no LRD in the wavelet domain!
- so classical statistics, \( 1/n \) convergence.

For LRD expect energy at \( j \) of:
\[
\mathbb{E}[d_X(j, \cdot)^2] = 2^{j\alpha_c} \mathcal{C},
\]

Unbiased estimate is
\[
\mu_j = \frac{1}{n_j} \sum_{k=1}^{n_j} |d_X(j, k)|^2
\]
The Logscale Diagram

Definition: The Logscale Diagram (LD) is the graph of $y_j = \log_2(\mu_j)$ vs $j$, with confidence intervals about the $y_j$.

Estimate of $\alpha$: slope in LD via weighted regression.

Properties:

- for LRD: **lower cutoff** and alignment to largest scales
- small $j$ is small scale, more data, smaller CI’s
- look for **alignment**, but w.r.t confidence intervals!
- unbiased, near optimal variance (see Veitch, Abry, 97,98)
- fast $O(n)$ algorithm, can be performed on-line

LD for fARIMA(0,d,2) $(H, c_f) = (0.75, 6.38)$.
The On-line Algorithm

Off-line algorithm has three stages

1. Wavelet decomposit\textsuperscript{n}: \( X(t) \mapsto d_X(j,k) \), \([O(n), O(n)]\)
2. Variance estimat\textsuperscript{n}: \( d_X(j,k) \mapsto \mu_j, n_j \) \([O(n), O(\log_2 n)]\)
3. Slope calc\textsuperscript{n}: \( \mu_j, n_j \mapsto \tilde{\alpha}, (\tilde{c}_f) \) \([O(\log_2 n), O(\log_2 n)]\)

Off-line complexity: \([O(n), O(n)]\)

Step 1 can be done On-line by a filter bank:

1. On average \([2(K + 1), O(\log_2 n)]\) steps/coefficients
2. Step is trivial: \( n_j \leftarrow n_j + 1, S_j \leftarrow S_j + d_X(j,n_j)^2 \)
3. Don’t convert to on-line, just estimate as needed

On-line complexity: \([O(n), O(\log_2 n)]\)

Note: Can get whole Logscale Diagram, not just \((\tilde{\alpha}, \tilde{c}_f)\)
Measurement of TCP/IP over ATM

Original real-time system on 10Mb/s Ethernet

Consider ATM because

- a common point to point transport technology
- high speed available OC3 (155Mb/s), OC12 (622) ..
- although more difficult (not broadcast), an inexpensive PC based system possible

Based on existing CORAL project OCX3mon

- previously used on the vBNS
- drivers available for FORE Systems ATM NIC
- modified FreeBSD driver for our purposes
- use first-cell of packet mode – full TCP/IP headers
- packet rates available from first-cell timestamps
- byte rates inferred from packet sizes
System Architecture

Consists of 3 layers:

- **Lower Layer**: cell capture → time-series generation
- **Analysis Layer**: time-series analysis modules (in C)
- **GUI**: control, stream/parameter selection, display

Features:

- layers communicate over TCP/IP – allows for remote monitoring and shared processor load
- passive monitoring provided by optical splitter
Time Series Generation

Desired series selected from GUI

- can select on VP/VC
- could select on IP addresses, port numbers
- set time-series type and resolution ('sampling interval'), here packets or bytes per millisecond

Cells to time-series, algorithm

- accurate (1/70 cell-time) timestamps
- first-cells and timestamps written to buffer blocks
- track which timestamps falling in each sampling interval to define packet rate
- buffer blocks periodically flushed to avoid backlog (time-series analysis needs regular values!)
- no performance bottlenecks on 200Mhz CPU at 1ms resolution
An Example of Artificial LRD Traffic

Note:
Theoretical values is $H = (1 + \alpha)/2 = 0.75$. Traffic Generation follows an On/Off method (with infinite variance inter-arrivals generating LRD), and is not perfect!
Can store the entire Logscale Diagram, examine the scaling behaviour, watch the scales arrive.
Conclusion

- fractal parameters can be measured in real-time using a wavelet based estimator with excellent properties
- method is intrinsically scalable, with low memory requirements
- inexpensive hardware (< $5000)
- performance excellent at 1ms per data point, on a 200Mhz CPU
- system allows flexible, remote monitoring