Network Traffic Modeling

Mark Crovella
Boston University Computer Science
SIGCOMM 2004

Outline

- 9:00 - 11:00 Performance Evaluation
  - Part 0: Stationary assumption
  - Part 1: Models of short-timescale behavior
  - Part 2: Traffic patterns seen in practice

- 11:00 - 11:15 Break

- 11:15 - 12:30 Network Engineering
  - Models of long-timescale behavior
  - Part 1: Single Link
  - Part 2: Multiple Links
What is a Data Model? [SF03]

- A succinct description of a generative process that gives rise to an output (data) of interest.
  - Sometimes the underlying “process” isn’t emphasized, but it’s never very far away
- What do we want from Data Models?
  - The ability to concisely represent data
  - The ability to interpret data
- Data modeling can be approached two ways:
  - Constructive (White Box)
  - Descriptive (Black Box)

Constructive vs. Descriptive

- System
- Model
- Data

Constructive ➔ [Diagram]

? ➔ ?

? ➔ Descriptive
Tradeoffs

- **Constructive**
  - ✓ Model outputs easily interpretable
  - ✗ Generalization is difficult - too many parameters
  - ✗ Results generally won't match real measurements

- **Descriptive**
  - ✓ Model matches data well
  - ✗ Doesn't explain "why"
  - ✗ Doesn't answer "what if system changes"
  - ✗ Difficult to interpret model parameters

- **N.B.:** These approaches are not mutually exclusive

- We will generally use descriptive models, but we'll
  - Try to avoid pitfalls by seeking interpretable parameters
  - Occasionally use constructive models where possible

Why probabilistic models? [Mat89]

- We will generally use probabilistic data models
- Is this because we believe that the system is truly random? (hopefully not ...)
- We use probabilistic models because a deterministic model would require specifying
  - Things we don't care about
  - Things we don't (or can't) know
Probabilistic Models are Abstractions

In a serious work ... an expression such as "this phenomenon is due to chance" constitutes simply, an elliptic form of speech.

It really means "everything occurs as if this phenomenon were due to chance," or, to be more precise:

"To describe, or interpret or formalize this phenomenon, only probabilistic models have so far given good results."

Georges Matheron, *Estimating and Choosing*

Models and Parameters

- This distinction makes clear an important fact regarding the physical reality of probabilistic models:
  - Parameters of models (mean, variance, etc.) are abstractions
  - They are not objectively measurable quantities
- When we speak about data, we often take verbal shortcuts:
  "The mean of this dataset is 7" \(\Rightarrow\)
  "The mean of the model describing this dataset is 7"
- Keeping these facts in mind will help avoid considerable confusion!
Traffic Models: The Big Picture

- There are two main uses for Traffic Modeling:
  - Performance Analysis
    - Concerned with questions such as delay, throughput, packet loss.
  - Network Engineering and Management:
    - Concerned with questions such as capacity planning, traffic engineering, anomaly detection.

- Some principal differences are that of timescale and stationarity.

Relevant Timescales

Performance effects happen on short timescales
from nanoseconds up to an hour

Network Engineering effects happen on long timescales
from an hour to months

1 usec 1 sec 1 hour 1 day 1 week
Stationarity, informally
[NIS 04]
• “A stationary process has the property that the mean, variance and autocorrelation structure do not change over time.”
• This is a stationary model (not data!)
• Using a stationary model is a choice, whose justification should be examined.

The 1-Hour / Stationarity Connection
• Trends in traffic are primarily a result of varying human behavior over time
• The biggest trend is diurnal
• This trend can usually be ignored up to timescales of about an hour, especially in the “busy hour”
• Stationary models are often justified up to timescales c. 1 hour
Outline

• 9:00 - 11:00 Performance Evaluation
  - Part 0: Stationary assumption
  - Part 1: Models of short-timescale behavior
  - Part 2: Traffic patterns seen in practice

• 11:00 - 11:15 Break

• 11:15 - 12:30 Network Engineering
  - Models of long-timescale behavior
  - Part 1: Single Link
  - Part 2: Multiple Links

Part 1:
Traffic Models for Performance Evaluation

• Goal: Develop models useful for
  - Queueing analysis
    • eg, $G/G/1$ queues
  - Other analysis
    • eg, traffic shaping
  - Simulation
    • eg, router or network simulation
Modeling Traffic as a Stochastic Process

• A good (descriptive) model of network traffic is a **stochastic process**
  - We are generally talking about number of bytes (or packets or flows) per unit time
• A (discrete time) stochastic process is a collection of random variables \( \{X_i, i=1,2,...\} \).
• What is needed? Models for
  - Random variables
  - Stochastic processes

Distribution Function

*Given a random variable \( X \), we can fully characterize it by its probability distribution function (pdf):*
  *i.e., \( f(x) = p_X(x) \)*

*Estimated using a histogram*
Histograms and CDFs

- A histogram is often a poor estimate of the pdf $f(x)$ because it involves binning the data.
- The CDF $F(x) = P[X_i \leq x]$ will have a point for each distinct data value; can be much more accurate.

Modeling a Distribution

- We can form a compact summary of a pdf $f(x)$ if we find that it is well described by a standard distribution – e.g.,
  - Gaussian (Normal)
  - Exponential
  - Poisson
  - Pareto
  - Etc
- Statistical methods exist for:
  - asking whether a dataset is well described by a particular distribution
  - Estimating the relevant parameters
  - e.g., see [DS86]
Distributional Tails

- A particularly important part of a distribution is the (upper) tail
- \( P[X > x] \)
- Large values dominate statistics and performance
- "Shape" of tail critically important

Light Tails, Heavy Tails

- Light - Exponential or faster decline
- \( f_1(x) = 2 \exp(-2(x-1)) \)
- Heavy - Slower than any exponential
- \( f_2(x) = x^{-2} \)
Examining Tails

- Best done using log-log complementary CDFs
- Plot log(1-F(x)) vs log(x)

Heavy Tails Arrive

pre-1985: Scattered measurements note high variability in computer systems workloads
1985 - 1992: Detailed measurements note “long” distributional tails
  - File sizes
  - Process lifetimes
1993 - 1998: Attention focuses specifically on (approximately) polynomial tail shape: “heavy tails”
post-1998: Heavy tails used in standard models
Power Tails, Mathematically

A special case of heavy tails is power tails

We say that a random variable $X$ is power tailed if:

$$P[X > x] \sim x^{-\alpha} \quad 0 < \alpha \leq 2$$

where $a \sim b$ means $\lim_{x \to \infty} \frac{a(x)}{b(x)} = 1$.

Focusing on polynomial shape allows

Parsimonious description
Capture of variability in $\alpha$ parameter

A Fundamental Shift in Viewpoint

• Traditional modeling methods have focused on distributions with “light” tails
  - Tails that decline exponentially fast (or faster)
  - Arbitrarily large observations are vanishingly rare
• Heavy tailed models behave quite differently
  - Arbitrarily large observations have non-negligible probability
  - Large observations, although rare, can dominate a system’s performance characteristics
Heavy Tails are Surprisingly Common
[Cro00]

- Sizes of data objects in computer systems
  - Files stored on Web servers
  - Data objects/flow lengths traveling through the Internet
  - Files stored in general-purpose Unix filesystems
  - I/O traces of filesystem, disk, and tape activity
- Process/Job lifetimes
- Node degree in certain graphs
  - Inter-domain and router structure of the Internet
  - Connectivity of WWW pages
- Zipf’s Law

Evidence: Web File Sizes
[BBBC99]
Evidence: Process Lifetimes
[HBD97]

The Bad News
[CL97]

- Workload metrics following heavy tailed distributions are extremely variable
- For example, for power tails:
  - When $\alpha \leq 2$, distribution has infinite variance
  - When $\alpha \leq 1$, distribution has infinite mean
- In practice, empirical moments are slow to converge - or nonconvergent
- To characterize system performance, either:
  - Attention must shift to distribution itself, or
  - Attention must be paid to timescale of analysis
Heavy Tails in Practice

Power tails with $\alpha=0.8$

Large observations dominate statistics (e.g., sample mean)

Characterizing a Stochastic Process

- Building on models for random variables, we can model stochastic processes
- Complete description of a stochastic process requires specifying all joint distributions, i.e.,
  
  $p_{X_1}$, $p_{X_2}$, $p_{X_3}$, ...
  $p_{X_1,X_2}$, $p_{X_1,X_3}$, $p_{X_2,X_3}$, ...
  $p_{X_1,X_2,X_3}$, ...
A Reasonable Approach

• Fully characterizing a stochastic process can be impossible
  - Potentially infinite set of properties to capture
  - Some properties can be very hard to estimate

• A reasonable approach in the case of stationary network traffic models is to concentrate on two particular properties:

  marginal distribution and autocorrelation

Marginals and Autocorrelation

Characterizing a traffic process in terms of these two properties gives you
  - a good approximate understanding of the process,
  - without involving a lot of work,
  - or requiring complicated models,
  - or requiring estimation of too many parameters.

... Hopefully!
**Marginals**

Given a stochastic process $X = \{X_i\}$, we are interested in the distribution of any $X_i$:

i.e., $f(x) = p_{X_i}(x)$

Since we assume $X$ is stationary, it doesn't matter which $X_i$ we pick.

Estimated using methods for random variables, i.e., histograms, etc.

**Autocorrelation**

- Once we have characterized the marginals, we know a lot about the process.
- In fact, if the process consisted of i.i.d. RVs, we would be done.
- However, most traffic has the property that its measurements are not independent.
- Lack of independence usually results in autocorrelation
- Autocorrelation is the tendency for two measurements to both be greater than, or less than, the mean at the same time.

![Histogram](image)
Autocorrelation Function (ACF) (assumes stationarity):

\[ R(k) = \text{Cov}(X_n, X_{n+k}) = E[X_n X_{n+k}] - E[X_0]^2 \]
Performance Evaluation Part 2:
Traffic Patterns Seen in Practice

- Long Range Dependence in Byte Traffic
  - Self-similarity
  - Estimation

- Reference Traffic Models
  - Sessions
  - Flows
  - Bytes and Packets

ACF of samples of i.i.d. random variables
How Does Autocorrelation Arise?

Traffic is the superposition of flows

Why "Flows?": Sources appear to be ON/OFF

- P1: ON - OFF
- P2: \[\text{ON} \quad \text{OFF} \]
- P3: ON - OFF
- \[\vdots\]
Superposition of ON/OFF sources $\implies$ Autocorrelation

Long Range Dependence

$[\text{Ber94}]$

$R[k] \sim k^{-a}$ \quad $0 < a < 1$

$H=1-a/2$

$R[k] \sim a^k$ \quad $a > 1$
Correlation and Scaling

- Long range dependence affects how variability scales with timescale
- Take a traffic timeseries \( X_n \), sum it over blocks of size \( m \)
  - This is equivalent to observing the original process on a longer timescale
- How do the mean and std dev change?
  - Mean will always grow in proportion to \( m \)
  - For an i.i.d. process, the std dev will grow in proportion to \( \sqrt{m} \)
  - So an i.i.d. process is “smoother” at longer timescales

Self-similarity: unusual scaling of variability

[LTWW93]

- Exact self-similarity of a zero-mean, stationary process \( X_n \)

\[
X_i \overset{d}{=} m^{-H} \sum_{i=(t-1)m+1}^{tm} X_i \\
\text{for all } m \in \mathbb{N}, t > 0
\]

- \( H \): Hurst parameter \( 1/2 < H < 1 \)
- \( H = 1/2 \) for i.i.d. \( X_n \)
- LRD leads to (at least) asymptotic s.s.
Self Similarity in Practice
Byte / Packet Traffic [PKC96]

The Great Wave (Hokusai)
How Does Self-Similarity Arise?

Flows $\Rightarrow$ Autocorrelation $\Rightarrow$ Self-similarity

Distribution of flow lengths has power law tail $\Rightarrow$
  Autocorrelation declines like a power law

Power Tailed ON/OFF sources $\Rightarrow$
Self-Similarity

P1:
P2:
P3:
Measuring Scaling Properties
[TTW95]

- In principle, one can simply aggregate $X_n$ over varying sizes of $m$, and plot resulting variance as a function of $m$.
- Linear behavior on a log-log plot gives an estimate of $H$ (or $a$).
- Slope > -1 indicates LRD.

**WARNING:** this method is very sensitive to violation of assumptions (stationarity)!

Better: Wavelet-based estimation
[AV98]
Three levels of On/Off structure in Traffic
Reference Models for Traffic

- Measured traffic is the superposition of all of these patterns
- Session arrivals
  - Generally well modeled as Poisson
  - Internal structure application dependent
- Flows
  - Lengths are generally well modeled as heavy-tailed
    - "elephants and mice"
  - Arrivals are application, user, & network dependent
- Packets / Bytes
  - Counts are generally well modeled as LRD
  - Marginal distribution is largely network dependent

Simulation models for traffic

[BC98]

- Rather than trying to reproduce traffic properties at the packet level, much better to employ a source model
  - Move "up a level" and shift from purely descriptive to a mixed descriptive/structural model
- Generate sessions consisting of heavy-tailed flows (byte counts)
  - Simulate transport protocol (e.g., in ns)
- Results in self-similar traffic that is generated in a natural manner and that responds realistically to network conditions
Traffic Shaping Effects

Traffic patterns on a link are strongly affected by two network factors:

- Amount of *multiplexing* on the link
  - Essentially – how many flows are sharing the link at a typical moment?
- Where flows are *bottlenecked*
  - Is each flow's bottleneck on, or off the link?
  - Do all bottlenecks have similar rate?

**Low Multiplexed Traffic**

- Marginals: highly variable
- Autocorrelation: low
Highly Multiplexed Traffic

Highly Multiplexed, Bottlenecked Traffic

- Marginals: tending to Gaussian
- Autocorrelation: high
Highly Multiplexed, Mixed-Bottlenecks

Internet Traffic Archive [DMPS]

Alpha and Beta Traffic [SRB01]

High variability in connection rates (RTTs)

Low rate = beta

High rate = alpha

fractional Gaussian noise

stable Levy noise
Very Small Timescales
[FGW98, RW99]

- Below the timescales of a round-trip-time (apprx 100 ms) packet arrival processes can have very complex properties
- Result of interactions of TCP control loops in multiple flows and network conditions
- Has been characterized as “multifractal”

Outline

- 9:00 - 11:00 Performance Evaluation
  - Part 0: Stationary assumption
  - Part 1: Models of short-timescale behavior
  - Part 2: Traffic patterns seen in practice

- 11:00 - 11:15 Break

- 11:15 - 12:30 Network Engineering
  - Models of long-timescale behavior
  - Part 1: Single Link
  - Part 2: Multiple Links
Outline

• 9:00 - 11:00  Performance Evaluation
  - Part 0: Stationary assumption
  - Part 1: Models of short-timescale behavior
  - Part 2: Traffic patterns seen in practice

• 11:00 - 11:15  Break

• 11:15 - 12:30  Network Engineering
  - Models of long-timescale behavior
  - Part 1: Single Link
  - Part 2: Multiple Links

Network Engineering

• Moving from stationary to nonstationary models
• Goal: Traffic models that are useful for
  - capacity planning
  - traffic engineering
  - anomaly / attack detection
• Two main variants
  - Looking at traffic on a single link at a time
  - Looking at traffic on multiple or all links in a network simultaneously
Part 1: Single Link Analysis

- Now, mean is changing
  - Introduces a source of predictability into network traffic
  - Usually the largest single source of traffic variation
- Variability around mean is essentially described as before
  - Gaussian + nonGaussian, generally LRD
  - But we also tend to encounter anomalies
    - Variability due to unexpected events or conditions

A Typical Trace

Three notable features:
1. periodicity - predictability of the mean
2. noisiness - random variation about the mean
3. “spikes” - nonGaussian deviations, anomalies
How should we model this traffic?

- In general, we are looking for a way of representing traffic that makes it easy to separate what is important from what is not.
  - Essentially a descriptive approach
- “what is important” varies from problem to problem
- Separation is usually treated linearly
  - i.e., traffic is viewed as the sum of various components, some of which are important and some not.
  - can be viewed as a linear filtering problem

Problems

- Depending on the problem, we might be interested in separating or isolating:
  - Long term trend
  - Daily or Weekly pattern of variation
  - Unusual spikes
  - Usual noise
Methods

- Signal processing and statistical techniques are common
  - ARMA, ARIMA timeseries models
  - Frequency Domain Analysis (Fourier)
  - Time-Frequency Analysis (Wavelet)
  - Principal Component Analysis
- Won’t describe methods in detail
  - All essentially decompose measurements linearly (sum of components)
  - Differ in how components are chosen

Comparing Methods Broadly

- Frequency domain (Fourier)
  - Traffic is viewed as the sum of periodic functions with infinite extent in time
  - Good if what you care about is something periodic, and present everywhere
- Time-Frequency Analysis (Wavelets)
  - Traffic is viewed as the sum of finite-extent patterns at varying locations and scales
  - Good if what you care about is multiscale, but not necessarily periodic or present everywhere
- Principal Component Analysis
  - (Multivariate) traffic is viewed as sum of arbitrary functions, chosen to capture maximum variability in minimum dimension
  - Good if what you care about is minimizing number of components
Examples

- Capacity planning for single links
  - Fourier and Wavelet analysis
- Anomaly detection for single links
  - Wavelet analysis
- Anomaly detection for multiple links
  - Principal Component Analysis

Capacity Planning

[PTZD03]

- Goal: ability to predict traffic volume 6 months in future
- Mainly interested in the long-term trend
- Need to extract and model the trend
  - i.e., separate short-term variability from long term
  - Essentially a multiscale problem
Periodicity: Fourier Analysis

Strong periodicities at 12, 24, and 168 hours

Denoising with Wavelets

Essentially, a smoothing operation

Scale 4 corresponds to smoothing over 24 hours

Scale 6 (4 days) is a reasonable level for long term prediction – avoids most intra-week variability
Capacity Planning via Forecasting

Timeseries models based on linear trend in mean, linear trend in std dev

Anomaly detection

- Goal: models that are useful in detecting anomalies
- What is an anomaly?
  - Unusually large service demand
  - Network equipment failures / misconfigurations
  - Customer traffic shifts
  - Flash crowds
  - Worm propagation
  - Network Abuse (DOS attacks, port scans)
Anomalies as traffic outliers

- Most of these anomalies can be observed as unusual variations in traffic
- What is “unusual?”
- Single-link view:
  - Capture variation of the mean
  - Capture typical variation around the mean
  - What’s left is unusual
- Clearly, this is somewhat heuristic
- Leads to detection, but not necessarily identification of anomaly

Separating Signal and Noise via Wavelets [BKPR02]
Anomaly Detection

- Misconfiguration of client backup software
- Interested in the noise rather than the trend
- Detected via visual inspection

Part 2: Multiple Links

- How to analyze traffic from multiple links?
  - Clearly, could treat as a collection of single links, and proceed as before
  - But, want more: to detect and exploit trends and patterns across multiple links
- Observation: multiple links share common underlying patterns
  - Diurnal variation should be similar across links
  - Many anomalies will span multiple links
- Problem is one of pattern extraction in high dimension
  - Dimension is number of links
Whole-Network Traffic is a Multidimensional Timeseries

Some have visible structure, some less so...

Example Link Traces from a Single Network
High Dimensionality: A General Strategy

- Look for a low-dimensional representation preserving the most important features of data
- Often, a high-dimensional structure may explainable in terms of a small number of independent variables
- Commonly used tool: Principal Component Analysis (PCA)

Principal Component Analysis

For any given dataset, PCA finds a new coordinate system that maps maximum variability in the data to a minimum number of coordinates

New axes are called Principal Axes or Components
Predictability in Space, not Time

Example Problem: Anomaly Detection
[LCD04a]

- Given set of all link measurements across a network, identify time points where measurements are anomalous
- PCA based detection
  - Normal variability captured in certain components
  - Anomalous variability lies outside those components

Traditional Frequency-Domain Analysis
PCA on Link Traffic

• Start with data in matrix form (X)

PCA

• Computed via Singular Value Decomposition
• A factorization of the X matrix:
  \[ X = U \Sigma V^T \]
• U: same shape as X; \( \Sigma \): diagonal; V: square
• The most significant patterns present in the columns of X are stored in the first columns of U
• Call these patterns "eigenlinks"
• Amount of variability captured in each eigenlink (component) is on diagonal of \( \Sigma \)
  - "singular values"
PCA on Link Traffic

\[ \mathbf{X} : \text{Link Traffic matrix} \]

\[ \mathbf{U} : \text{Eigenlink matrix (Columns are Orthonormal)} \]

\[ \mathbf{V} : \text{Principal matrix (Orthogonal)} \]

Low Intrinsic Dimensionality of Link Data
Sprint (49 links); Abilene (41 links)

A plot of the singular values reveals how much energy is captured by each PC.

Sharp elbow indicates that most of the energy captured by 4-5 components, for all datasets.
PCA is a linear decomposition

- Eigenlinks (U) are linear combinations of Link traffic measurements (X)
  \[ U = X\Sigma^{-1} \]
- Link msmts are linear combs. of eigenlinks
  \[ X = U\Sigma V^T \]
- Linear, like Fourier or Wavelet Analysis
  - Difference: components (eigenlinks) are chosen to capture maximum variability in minimum number of components
Decomposing all Links at Once

- **Goal**: decompose a set of link measurements $X$ into normal and anomalous parts: $X = X' + X''$
- $X' = U'\Sigma V^T$ where $U'$ is first few columns of $U$
- $X''$ is what remains
  - residual

**PCA based anomaly detection**

Sprint data

- $L_2$ norm of entire traffic vector $X_t$
- $L_2$ norm of residual vector $X_t''$
Variants
[LC04b]
- The X matrix may be derived from many possible multivariate traffic timeseries, e.g.:
  - IP (5-tuple) flows per Origin-Destination flow
  - Bytes per OD flow
  - Packets per OD flow
- All of these exhibit low effective dimension and are suitable for PCA based anomaly detection
- A wide range of anomalies can be detected this way
  - DoS Attacks, Port Scans, Network Abuse
  - Customer Traffic Shifts
  - High Rate (Alpha) IP Flows

Whew! Looking Back
- Performance evaluation models:
  - Stationary stochastic processes
  - Characterized by marginals and autocorrelation
  - Real traffic has surprising properties:
    - Heavy-tailed transfer sizes
    - Long-range dependence
- Network engineering models
  - Nonstationary (at least, usually...)
  - Mean + Noise + Spikes
  - Choice of model / decomposition based on problem at hand
A Question of Timescale

Performance Evaluation
- Marginals
- Watch out for heavy tails
- Correlation (in time)
- Watch out for LRD / Self Similarity

Network Engineering
- Choose your decomposition
- Single-link: Exploit temporal correlation
- Multi-link: Can also exploit spatial correlation

Traffic Analysis as a Data Modeling Problem

- Mainly descriptive
  - Minimize weaknesses: Seek models with interpretable parameters (e.g., mean, variance, Hurst parameter)
- Sometimes mix descriptive and constructive
  - Source-modeled traffic generation
    - Constructive: based on knowledge of applications
    - Descriptive: based on empirical distributions
- Seek decompositions that expose properties of interest
  - Fourier vs. Wavelets vs. PCA vs. ??
Traffic Modeling Observations

- When we simplify “heavy tails” to become “power tails” we gain the simplicity of using the single alpha parameter
  - Increases interpretability
  - Smaller alpha -> more variable
- When we trace self similarity to heavy-tailed transfer sizes, we are using a mixed descriptive/constructive approach
  - We are building a simplified system model and a simplified data model and working toward the middle
  - Allows us to understand not just s-s, but range of traffic types including alpha/beta, etc

Thanks!

- Questions?

- Please fill out an evaluation form!
  - feedback is welcome

- Slides and Bib are available on the Web
  www.cs.bu.edu/faculty/crovella/traffic-tutorial.pdf
  www.cs.bu.edu/faculty/crovella/traffic-bib.pdf