ECEN 689
Special Topics in Data Science for Communications Networks

Nick Duffield
Department of Electrical & Computer Engineering
Texas A&M University

Lecture 16
Delay Tomography and Topology Inference
Delay Inference

• Packets incur latency in traversing links
  – propagation delay on links
  – queueing and processing delays
• Delay is cumulative along paths
• Network with link delays \{X_i\} and path delays \{D_p\}
• Formally:
  – \(D_p = \sum_i A_{pi} X_i\) where \(A = [A_{pi}]\) is incidence matrix of links over paths
Delay Inference

• \( D_p = \sum_i A_{pi} X_i \)

• Interpretations:
  1. \( X_i \) represent common distribution of delays on all packets traversing \( i \)
     • sample values for different packets differ, e.g. on different paths
  2. \( X_i \) are common delay values experienced on any path \( p \) traversing
     • multicast packet
  3. \( X_i \) represent mean values
     • Invert linear system \( D = A.X \) to recover \( X \) from \( D \)?
Delay Model for Multicast Trees

- **Tree model**
  - known tree $G=(V,L) =$ (nodes, links)
  - source multicast probes from root
  - receiver nodes at leaves $R$

- **Delay model**
  - Probe delay $X_k$ on link $k$
    - Treat loss as infinite delay
  - Independent delays
    - Between links, probes
    - Assume finite 4th moments of $X_k$
      - for non-lost packets
Internal vs. External Characteristics

- Cumulative delay from root to node $k$
  \[ D_k = \sum_{i \text{ ancestor of } k} X_i \]

- Aim
  - characterize link delays $X_k$ in terms of cumulative delays $D_k$ to leaves

- Data
  - Multicast $n$ probes from source
  - Data $D = \{ D^i(j): j \in \mathbb{R}, i = 1, \ldots, n \}$

- Infer
  - Link and cumulative delay variance from data $D$
Cumulative Delay Variance on Tree

- Subtree spanning root 0, and receivers 2,3
  - node 1 = 2 \land 3: closest ancestor to nodes 2,3

- End-to-end delay covariance
  \[ \text{Cov}(D_2, D_3) = \text{E}[D_2 D_3] - \text{E}[D_2] \text{E}[D_3] \]
  - Captures common variability of end-to-end delays

- Theorem
  - \[ \text{Cov}(D_2, D_3) = \text{Var}(D_1) \]

- Covariance of end-to-end delays on path 0 \rightarrow 2 and 0 \rightarrow 3 = delay variance on shared path 0 \rightarrow 1
Cumulative Delay Variance

• Proof of Theorem

\[
\text{Cov}(D_2, D_3) = \text{Cov}(D_1 + (D_2 - D_1), D_1 + (D_3 - D_1))
\]

\[
= \text{Cov}(D_1, D_1) + \text{Cov}(D_1, D_3 - D_1) + \text{Cov}(D_2 - D_1, D_1) + \text{Cov}(D_2 - D_1, D_3 - D_1)
\]

\[
= \text{Cov}(D_1, D_1)
\]

\[
= \text{Var}(D_1)
\]
Characterizing Link Delay Variance

• \(X_4=\text{delay on link (1,4)}\)

• Delay cumulative along path
  – \(D_4 = D_1 + X_4\)

• Expand delay variance
  – \(\text{Var}(D_4) = \text{Var}(D_1) + \text{Var}(X_4) + 2\text{Cov}(D_1,X_4)\)

• Re-express link delay variance
  – In terms of end-to-end delay covariance

\[
\text{Var}(X_4) = \text{Var}(D_4) - \text{Var}(D_1) \\
= \text{Cov}(D_5,D_6) - \text{Cov}(D_2,D_3)
\]
Estimation from Data

- Unbiased Estimator of \( \text{Var}(X_1) \)
  - from delays at leaves 2 and 3
    \[ v(2,3) = (n-1)^{-1}(\sum_i D_2(i)D_3(i))^{-1} \sum_i D_2(i) \sum_i D_3(i) \]

- Such estimators not unique
  - e.g. from delays at leaves 4 and 5
    \[ v(4,5) = (n-1)^{-1}(\sum_i D_4(i)D_5(i))^{-1} \sum_i D_4(i) \sum_i D_5(i) \]

- Convex family
  - of unbiased estimators
    - in form \( \sum_{i,k} \mu(i,k) v(i,k) \),
    - with \( \mu(i,k) > 0, \sum_{i,k} \mu(i,k) = 1 \)

- Theorem
  - each such estimator is consistent
    - converges to true value as \#probes \( \to \infty \)
Minimum Variance Estimation

• General Estimator $\Sigma_{ik} \mu(i,k) v(i,k)$
  – adapt $\mu(i,k)$ to data:
    • to minimize variance of estimated variance
    • to de-emphasize contribution of high variance paths
    • increase convergence rate

• Example: estimate $\text{Var}(D_1)$
  – combine $v(2,3), v(3,4), v(4,2)$
  – de-emphasize $\mu(2,3), \mu(2,4)$

• Minimum Variance Estimator
  – select $\mu = C^{-1}.1/1.C^{-1}.1$
    • $C$ = (estimated) covariance matrix of $v(i,j)$
    • Finite 4th moment property used here

• Contrast Uniform Estimator
  – $\mu(i,k) = \text{constant}$
Estimator Convergence

Model simulation, 8-leaf binary topology
- faster convergence for minimum variance estimator
Path Delays: Filtering Propagation Delay

- **Setting:** single multicast tree
  - Generalizable to “forest” of trees as before

- **Path delays**
  - Total = Propagation + Queueing
  - Focus on queuing delay
  - Normalize n path delay measurement \( \{ D_i(k): i=1,\ldots,n \} \) to node k
  - \( D_i(k) \rightarrow D_i(k) - \min_i D_i(k) \)
    - Subtract off minimum observed delay
    - Assume some packet incurs no queuing delay
Link Delay Model

• Link delay model
  – Quantize delay to finite set of value \( Q = \{0, q, 2q, \ldots, Bq, \infty\} \)
  – Bins \([0,q/2),[q/2,3q/2),\ldots,[(B-1/2)q,(B+1/2)q),[(B+1/2)q,\infty)\} \)

• Discretized distribution of delay \( X_i \) on link \( i \)
  – \( \Pr[X_i = d] = \alpha(i,d) \), \( d \in Q \)

• Assume delays independent between packets and links
Complete Data Likelihood Function and EM

- **Complete data**
  - Queueing delays at all nodes: \( X = \{X(k): k \in V\} \)
  - Summarize as \( n(k,d) = \# \text{packets experience delay } d \text{ on link } k \)

- **Observed data**
  - Path delays at receivers: \( D = \{D(k): k \in R\} \)

- **Complete Data log-Likelihood**

\[
\log L^c_{\alpha}(X) = \sum_{k \in V} \sum_{d \in Q} n(k,d) \log \alpha(k,d)
\]

- **MLE**: \( \alpha(k,d) = \frac{n(k,d)}{\sum_{d \in Q} n(k,d)} \)

- **EM algorithm**
  - Replace \( n(k,d) \) with \( E_{\alpha^*}[ n(k,d) \mid D ] \)

- Computation significantly more complex than loss case

- Generalizes to multiple tree case
Topology Inference
Topology Inference

• Problem
  – given
    • multicast probe source
    • receiver traces (loss, delay, …)
  – identify (logical) topology

• Motivation
  – topology may not be supplied in advance
  – grouping receivers for multicast flow control
General Approach to Topology Inference

• Given model class
  – tree with independent loss or delay

• Find classification function of nodes $k$ which is
  – increasing along path from root
  – can be estimated from measurements at $R(k) = \text{leaves descended from } k$

• Examples
  – $1 - A_k = \text{Prob[probe lost on path from root 0 to } k]$  
  – mean of delay $Y_k$ from root to node $k$
  – variance of delay $Y_k$ from root to node $k$

• Build tree by recursively grouping nodes $\{r_1, r_2, \ldots, r_m\}$
  – to maximize classification function on putative parent
Example: Loss-based Topology Inference

- Given set of receivers $R$, but unknown tree
  - each $j \in R$: set $\gamma_j = \text{Prob}[\text{probe received at node } j]$

- For each subset $S \subseteq R$
  - set $\gamma_S = \text{Prob}[\text{probe received at some node } j \in S]$
  - let $A_S$ be unique solution $> \gamma_S$ to $(1 - \gamma_S/A_S) = \prod_{j \in S}(1 - \gamma_j/A_S)$

- Select maximal $S \subset R$ that minimizes $A_S$
  - join $\{S\}$ to set of vertices (initially just $R$)
  - join $\{(S,j), j \in S\}$ to set of links (initially empty)
  - set $R = (R\setminus S) \cup \{S\}$

- Iterate till $\#R = 1$

- Theorem: algorithm reconstructs tree when $0 < \gamma_k < 1$
  - using estimates $\gamma'_k$: $\text{Pr}[\text{misclassification}] \to 0$ as $\#\text{probes} \to \infty$
Correlation and Unicast Tomography
Illustrative Example: 2 Leaf Tree

• Packet Transmission Model
  – traverse link $k$ independently with prob. $\alpha_k$

• Method
  – source “stripes” packets to receivers
    1 = packet received, 0 = not received
  – initially assume single packet loss rates uniform across stripe
  – joint end-to-end transmission probabilities
    • $p(ij) = \text{Prob}[ i \text{ at receiver } L, j \text{ at receiver } R]$

• Aim
  – estimate the $\alpha_k$ from the $p(ij)$
A Thought Experiment

• Suppose
  – perfect loss correlation on link C
  – in given stripe:
    • left and right packets, or neither, reach C

• Inference
  – Left packet reaches L ⇒ right packet reaches C
    \[ \alpha_R = \text{Prob}[\text{right reaches R | right reaches C}] \]
    \[ = \text{Prob}[\text{right reaches R | left reaches L}] \quad \text{(perfect correlation)} \]
    \[ = \frac{p(11)}{p(11)+p(10)} \]

• Similarly:
  \[ \alpha_L = \frac{p(11)}{p(11)+p(01)}, \quad \alpha_C = \frac{(p(11)+p(10))(p(11)+p(01))}{p(11)} \]

• Consequence
  – Could estimate \( \alpha \) from measured end-to-end probabilities \( p \)

• Statistically: like estimation from multicast probes
Imperfect Correlations

• Loss correlations are imperfect
  – $\beta = \text{Prob}\left[\text{left reaches } C \mid \text{right reaches } C\right] \leq 1$

• If perfect correlation:
  – $p(11) = \alpha_C \alpha_L \alpha_R$

• If imperfect correlation:
  – $p(11) = \alpha_C \alpha_L \alpha_R \beta$

• Under “standard estimator”
  – substitute probabilities $p$ by observed frequencies
    • as if $\beta = 1$, i.e., incorrect assumption of perfect correlations
      – $\alpha_C(\text{est}) = \alpha_C \beta$ \hspace{1cm} $\alpha_L(\text{est}) = \alpha_L / \beta$ \hspace{1cm} $\alpha_R(\text{est}) = \alpha_R / \beta$

• Standard estimator
  – underestimates transmission rate on common link
  – overestimates transmission rate on leaf links
Extend Method?

• Account for and infer conditional probability $\beta$
• Problem: non-identifiability
  – too many parameters: $\alpha_C, \alpha_R, \alpha_L, \beta$
  – too few independent end-to-end probabilities $p(10), p(01), p(11)$
• Reduce effective number of independent parameters
  – link $\alpha_C$ and $\beta$ through queueing model
  – Bayesian inference from prior distribution for ($\alpha$, $\beta$)
  – yields distributional estimate for ($\alpha$, $\beta$)
• ML estimation + EM algorithm
  – generate approximating sequence ($\alpha^n, \beta^n$)
  – raises new issues
    • estimator bias, sensitivity to initial ($\alpha^0, \beta^0$)
Standard Estimator Revisited

• **Standard estimator most accurate when** $\beta \approx 1$
  – an uncontrolled assumption
  – expect better observance if:
    • inter-packet time < duration of congestion events
      – back-to-back packets

• **Stripe extension**
  – use more complex stripe pattern
  – event selection
    • infer only from events which favor $\beta \approx 1$
      – for an appropriate notion of conditional transmission rate

• **Path testing**
  – determine how closely $\beta \approx 1$
    • along end-to-end paths in the Internet
Extended Striping and Coalescence

• Extended striping
  – send multiple times to same receiver

• Conditional probabilities
  – $\beta(1|2\ldots n)$ = conditional prob. for packet 1 to reach node C, given packets 2,3,…,n all reach C

• Coalescence
  – idea:
    • packet transmission more likely given longer bursts
  – formalization:
    $\beta(1|2\ldots n) \leq \beta(1|2\ldots n+1)$
  – example:
    • transmission in the M/M/1/K queue is coalescent
Inference, Coalescence and Bias

• Utilizing coalescence
  – condition on event \{n\} = reception of packets 2,3,…,n
  – if correlations were perfect:
    \[ \alpha_L = \frac{p(1\{n\})}{p(1\{n\}) + p(0\{n\})} \]

• Reduction in estimator bias
  – packet 1 more likely to have reached C for larger n
    • analog of standard estimator becomes more accurate
    – \[ \alpha_{L\text{ (est)}} = \frac{\alpha_L}{\beta(1|2…n)} \]
      • coalescence + larger n \Rightarrow larger \beta \Rightarrow smaller bias
Inference on General Networks

• Estimation of loss rate $A_k$
  – along path from source to $k$
• Stripe template $S(i,j)$, e.g.,
  – stripe 1 packet to receiver $i$,
  – then $n$ packets to receiver $j$
• Estimate $A_k$
  – by $A_k(i,j)$ from stripe $S(i,j)$
    • receivers $(i,j)$ descended through different children of $k$
  – lightweight
    • pick one such pair $(i,j)$ for each node $k$
  – exhaustive
    • average $A_k(i,j)$ over all suitable pairs $(i,j)$
• Recover link transmission rates
  – $\alpha_k = A(k)/A(\text{parent}(k))$
Path Testing

• Using standard estimator
  – relative error of estimated transmission rate $\alpha_1$ is $\beta$
    • if $\beta \approx 1$:
      – standard transmission rate estimator is accurate
  – relative error of estimated loss rate $1-\alpha_1$ is $\alpha_1 (1-\beta)/(1-\alpha_1)$
    • standard loss rate estimator is accurate (assume low loss: $\alpha_1 \approx 1$)
      if $(1- \beta) < (1-\alpha_1)$ i.e. conditional loss < marginal loss

• Operational criterion for expected inference accuracy
  – stripe packets to single endpoint
    • if conditional loss < marginal loss on path
    • then typically conditional loss < marginal loss on lossiest links
  – path characteristic (assume low loss: $\alpha \approx 1$)
    • $\text{conditional\_loss/marginal\_loss}$
      – relative error in inferred loss due to imperfect correlations
  – test for coalescence along complete paths
Network Experiments
Measured Internet Path Properties

• **Aim**
  – is stripe based inference expected to be effective?

• **Internet paths**
  – experiments across 28 paths

• **Transmission Probabilities**
  – conditional > marginal

• **Coalescence**
  – $\beta(1|23) > \beta(1|2)$ in almost all cases (also for longer stripes)

• **Path characteristic**
  – relative error = conditional_loss/marginal_loss
    • median: 0.12 for 2 packet stripes, 0.10 for 3 packet stripes

• **Conclusion**
  – expect inference accurate for lossier links on these paths
Network Simulations

- **ns simulation**
  - 39 node topology, TCP/UDP background traffic
  - 1 µsec inter-packet time, 16ms inter-stripe time
  - infer over various subtrees,
    - e.g.
Conditional Probabilities and Stripe Length

- Conditional probabilities
  - link transmission
  - increase with stripe length
  - expect better inference accuracy

- Largest effect:
  - $2 \rightarrow 3$ packet stripe
  - Marginal improvement for $3 \rightarrow 4$ packet stripe

Scatter plot of $(\text{marginal, conditional})$