Efficient Network-wide Available Bandwidth Estimation through Active Learning and Belief Propagation

Mark Coates
mark.coates@mcgill.ca

McGill University
Department of Electrical and Computer Engineering
Montreal, Quebec, Canada

IPAM Workshop IV: New Mathematical Frontiers in Network Multi-Resolution Analysis

Outline

1. Problem Statement
2. Our approach: Active Learning and Belief Propagation
   - Measurement Model
   - Graphical Model
   - Active Learning Algorithm
3. Experimental Results and Discussion
   - Measurement Model
   - Simulation Results
   - Issues and Future Work

Available Bandwidth

Available Bandwidth of a link

$$A_i(t, t+\tau) = \frac{1}{\tau} \int_t^{t+\tau} C_i(x) - \lambda_i(x)\, dx \qquad (1)$$

Available Bandwidth of a path

$$A(t, t+\tau) = \min_{i=1,\ldots,H} A_i(t, t+\tau) \qquad (2)$$

Goal

- Available bandwidth is unused capacity of a path.
- Estimate using end-to-end measurements.
- "Tight" link determines the available bandwidth.

Rate Scanning Approach

Pathload [Jain & Dovrolis, 2002]

- Sender transmits periodic packet stream of rate R.
- Receiver measures one-way delay $D(k)$ for each packet.
- Calculate one-way delay variations $\Delta(k) = D(k) - D(k-1)$.
- Ideally (stationary, fluid-model cross-traffic) if $R > A$ then $\Delta(k) > 0$ for all k.
- Binary bisection search to determine upper and lower bounds.

Network-wide Measurement

Multiple paths

- High load when measuring multiple paths.
- If paths share links, simultaneous measurement can bias results.
- Sequential rate-scanning is a slow process for multiple paths.

Network-wide monitoring approaches

- Network kriging [Chua et al., 2006]
- Compressed network monitoring [Pointurier et al., 2007]
- Both methods rely on linear relationship between link and path metrics.

Measuring a Single Path

Send trains of probes

- Adjust rate: change packet size and time interval.
- Make observations at receiver side.

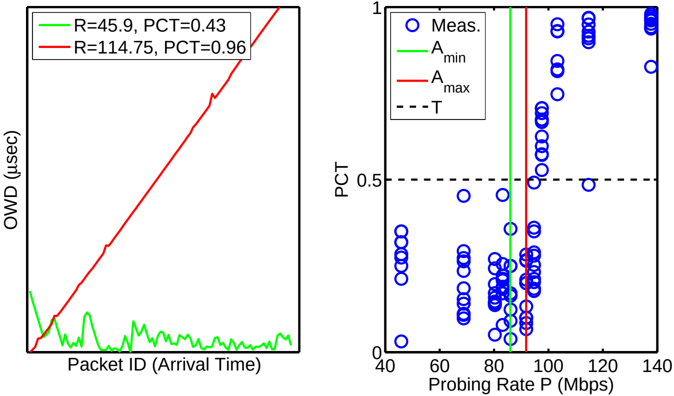
Observations

- One-way delay (owd): arrival time - departure time.
- Receiving rate (R).

Outcome: generate binary value

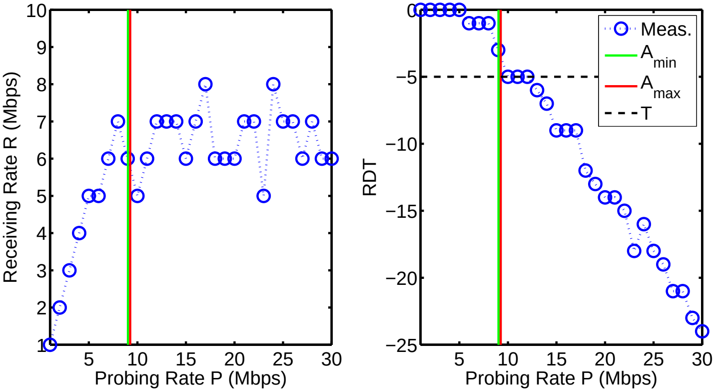
- Compare observations to threshold T.
  - 1: Probing Rate (P) $>$ Available Bandwidth (A).
  - 0: Probing Rate(P) $<$ Available Bandwidth (A).

Metric I: One-Way Delays



$$PCT = \left(\frac{\sum_{i=2}^{N} \mathbf{1}(owd_i - owd_{i-1})}{N} > T \right)$$

Metric II: Receiving Rate



$$RDT = (R - P < T)$$

Measuring Multiple Paths

(Very) Naive approach

Sequentially measure each path independently.

We can do better...

- Paths can have tight links in common.
- Use information from measurements on other paths.

Our approach

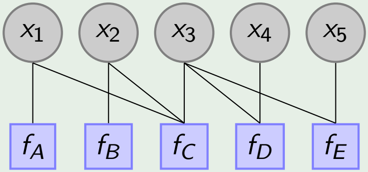
- Factor Graphs
- Belief Propagation
- Active Learning

Factor Graphs

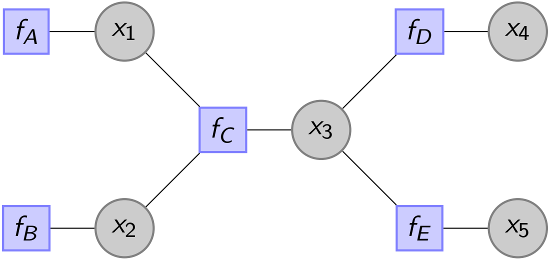
Definitions

- *Variable node* for each variable x_i.
- *Factor node* for each function f_i.
- Edge connecting x_i to f_j if x_i is an argument of f_j.

Example



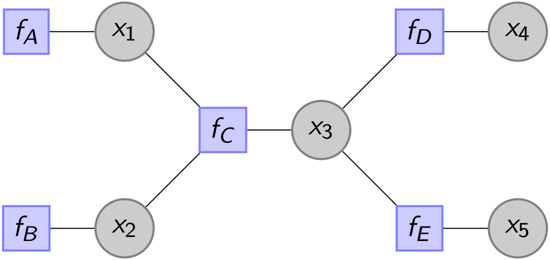
Belief Propagation



Sum-Product Update Rule

The message sent from a node v on an edge e is the product of the local function at v with all messages received at v on edges other than e, summarized for the variable associated with e.

Belief Propagation

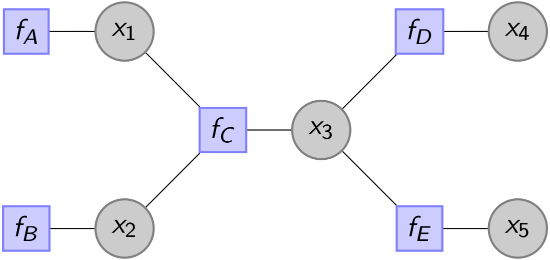


Message computation

$$\mu_{x \to f}(x) = \prod_{h \in n(x) \backslash \{f\}} \mu_{h \to x}(x)$$

$$\mu_{f \to x}(x) = \sum_{\sim \{x\}} \left(f(X) \prod_{y \in n(f) \backslash \{x\}} \mu_{y \to f}(y) \right)$$

Belief Propagation

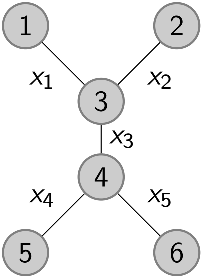


Computing marginal distributions

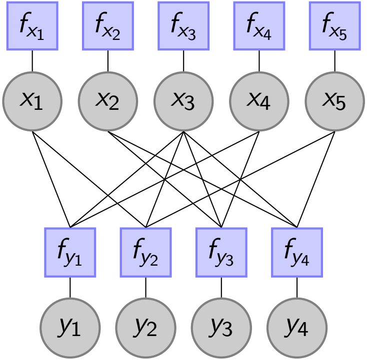
Product of all incoming messages

$$g_3(x_3) \quad = \quad \mu_{f_C \to x_3}(x_3)\mu_{f_D \to x_3}(x_3)\mu_{f_E \to x_3}(x_3)$$

Constructing the Factor Graph



Four paths: 1-3-4-5, 1-3-4-6, 2-3-4-5, 2-3-4-6.

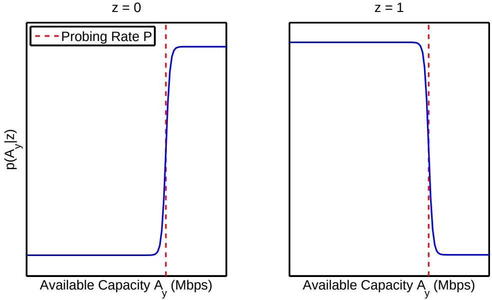


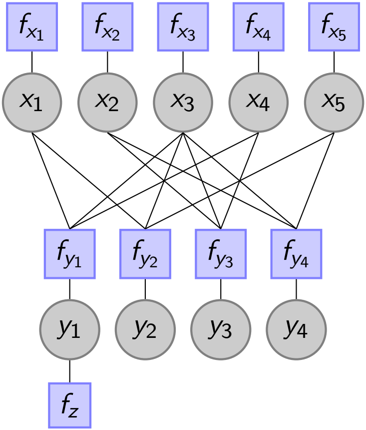
1. Variable nodes: links (x) and paths (y).
2. Factor nodes: prior distributions (f_x) and min functions (f_y).
3. Edges: $y_1 = \min\{x_1, x_3, x_4\}$.

Adding the measurement node

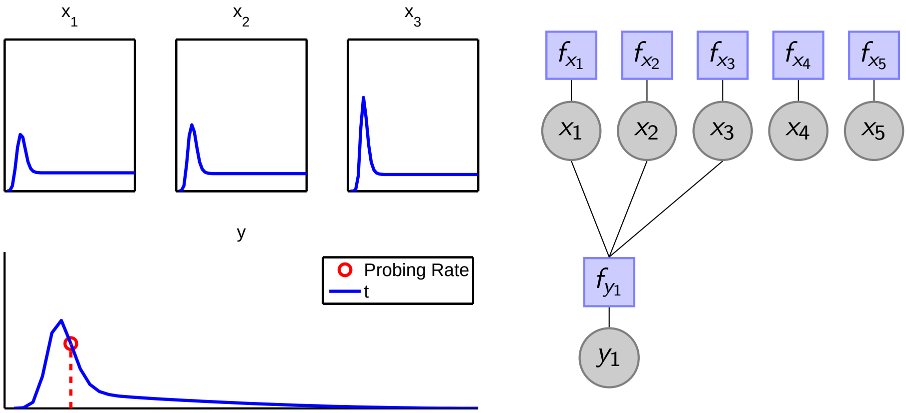
Measurements

- Connect factor node $f_z = \ell(z|A_y)$ to variable node being measured.
- Two possible outcomes: $z = 0 (P < A)$ or $z = 1 (P > A)$.

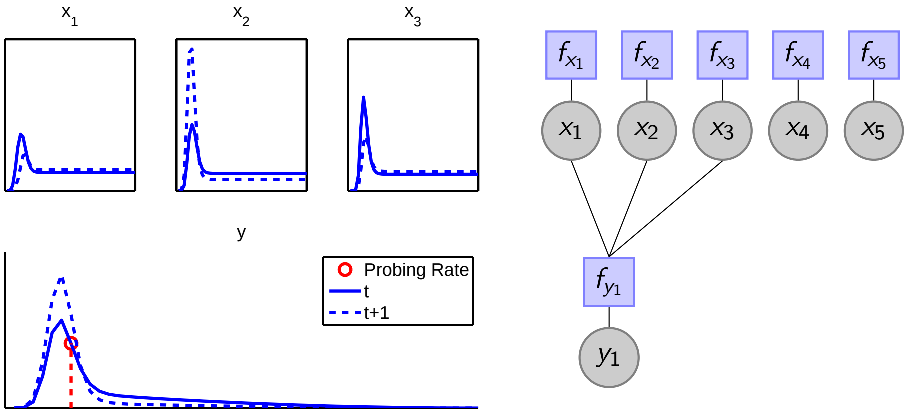




Updating the distributions: an example



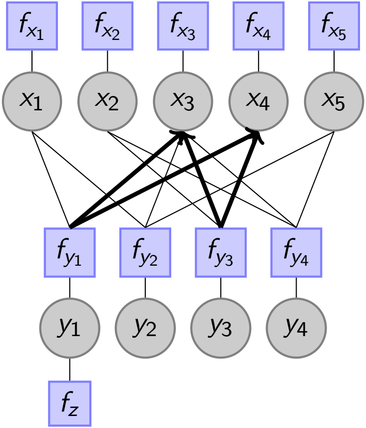
Updating the distributions: an example



Dealing with Cycles

Modified Update Rule

- Initially send unit messages on all edges.
- Nodes calculate new messages for all outgoing links.
- Messages are sent if different "enough" from previous one.



Our Algorithm

Belief Propagation + Active Learning

1. Create factor graph using known topology.
2. Determine which path to probe.
3. Determine probing rate P.
4. Add measurement node to factor graph.
5. Run belief propagation until convergence.
6. Calculate new links marginal and update f_x.
7. Calculate paths marginal.
8. Repeat until stopping criterion is met.

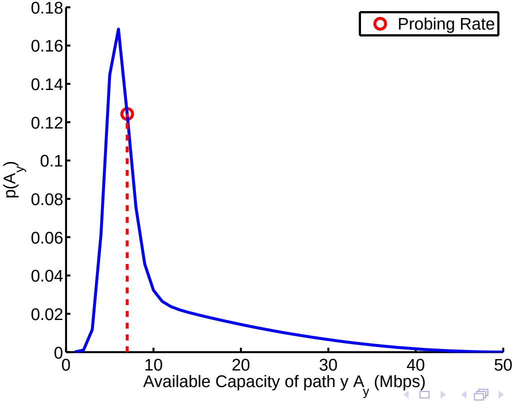
Active Learning Approach

Active Learning

- Use previous data to guide selection of new samples.
- Noiseless setting: $O(\log n)$ samples to achieve same performance as passive learning with $O(n)$ samples.
- $|\eta(x) - 1/2| \geq c|x - \theta^*|^{\kappa-1}$
- Bounded noise ($\kappa = 1$): Excess risk decays exponentially in number of samples.
- Decay rate depends on noise margin c.
- Unbounded noise ($\kappa > 1$): less significant improvement
- Lower bounds for $\kappa = 2$ are n^{-1} (active) versus $n^{-2/3}$ (passive)

Active Probing Rate Selection

- Use calculated marginal: probe at the median.
- This is (intuitively) the most informative measurement and algorithm achieves rates close to the lower bounds



Path Selection

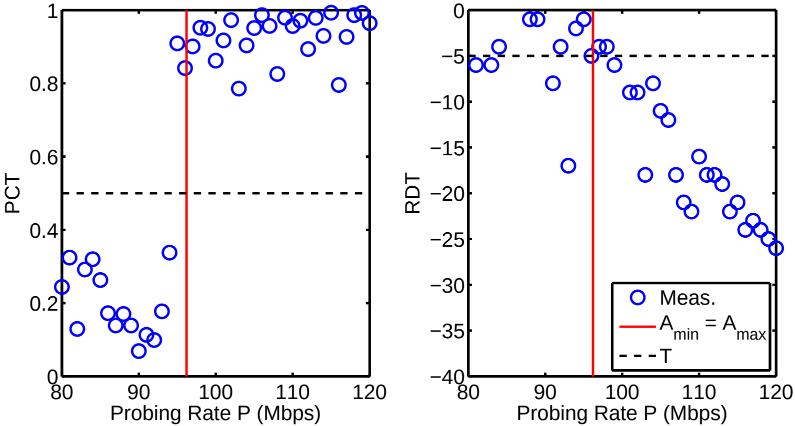
Passive Selection

- Series : probe the same path until stopping criterion is met.
- Round Robin (RR): probe the next path in the list.

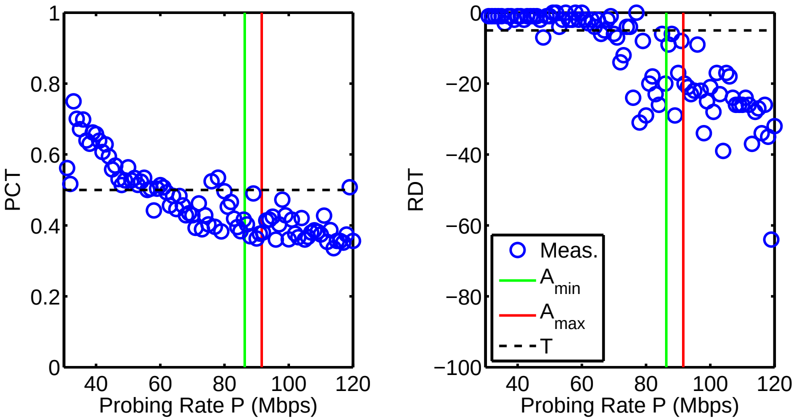
Active Path Selection: use available information

- Weighted Entropy (WE): coin toss w.p. proportional to path entropy.
- Active Path (ALP):
  1. Simulate measurement for every path.
  2. Run belief propagation for all possible outcomes.
  3. Choose path with best expected value (e.g. lowest sum of path entropy).

OWDs vs. Receiving Rate



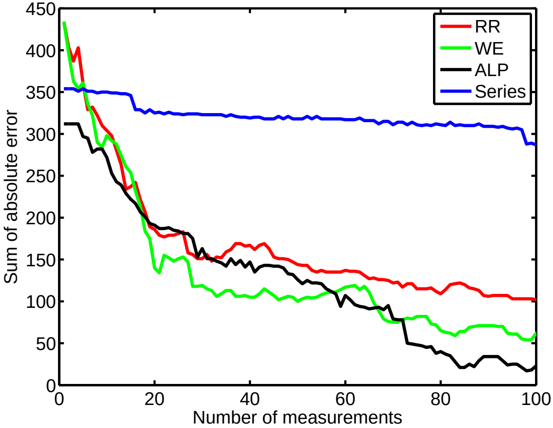
OWDs vs. Receiving Rate



Simulating the algorithm: RocketFuel topology

Setting

- 43 paths y_i.
- 12 links x_j.
- Shortest Path Routing.
- $A_{x_j} \sim \mathbf{U}(1, 50)$.



Current and Future Challenges

Issues

- Choosing best metrics.
- Message construction (likelihood model) for measurement metrics.
- Available bandwidth variations with time (appropriate scales, tracking?).
- Ground truth.

Ongoing and Future Work

- Determine lower bound for active learning heuristics (oracle).
- Algorithm deployed on PlanetLab: results coming soon.

System-theoretic Interpretation [Liebeherr et al. 2007]

Linear Time Invariant (LTI) Systems

- Input $x(t)$, Output $y(t)$, System response $h(t)$.
- Output can be computed from input and system response:

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)\,d\tau =: x * h(t) \qquad (3)$$

Network Calculus

- Networks can be viewed as linear systems in min-plus algebra
  - Addition $(+) \mapsto$ Minimum (inf)
  - Multiplication $(\times) \mapsto$ Addition $(+)$
- Network service is described by a service curve S.

Min-plus Linear Systems

- Input arrivals $B(t)$, Service curve $S(t)$, Departures $Z(t)$
- If input is infinite burst δ, then output is $S(t)$
- Departures can be calculated from arrivals and service curve (min-plus convolution)

$$Z(t) = \inf_{\tau} \{B(\tau) + S(t-\tau)\} =: B * S(t) \tag{4}$$

Non-linear Systems

- Many networks are not min-plus linear; $Z(t) \neq B * S(t)$.
- But they can be described by a lower-service curve $\underline{S}$.
- For all t, $Z(t) \geq B * \underline{S}(t)$.
- Lower bound on the service is often sufficient.

Legendre Transform

Legendre Transform

- Analogous role to Fourier transform in classical LTI systems.
- Maps to rate domain

$$\begin{aligned}\mathcal{L}_f(r) &= \sup_{\tau}\{r\tau - f(\tau)\} \\ \mathcal{L}_{f*g} &= \mathcal{L}_f + \mathcal{L}_g\end{aligned}$$

Properties

- For all functions $\mathcal{L}(\mathcal{L}(f)) \leq f$.
- If f is convex, then $\mathcal{L}(\mathcal{L}(f)) = f$.
- If f and g convex, then $f \geq g \Leftrightarrow \mathcal{L}(f) \leq \mathcal{L}(g)$.

System Theoretic Interpretation of Rate Scanning

- Max. backlog $G_{max}(t) = \sup_t \{B(t) - Z(t)\}$
- If $B(t) = rt$ then:

$$
\begin{aligned}
G_{max}(r) &= \sup_t \{rt - \inf_\tau \{r\tau + S(t-\tau)\}\} \\
&= \sup_t \{\sup_\tau \{r(t-\tau) - S(t-\tau)\}\} \\
&= \sup_t \{r(t) - S(t)\} \\
&= \mathcal{L}_S(r)
\end{aligned}
$$

- Inverse transform: If S is convex, then:

$$S(t) = \mathcal{L}(\mathcal{L}_S)(t) = \mathcal{L}_{G_{max}}(t) = \sup_r \{rt - G_{max}(r)\}$$

System Theoretic Rate Scanning Algorithm

- Transmit a packet train at rate r
- Compute $G_{max}(r)$ and $S(t) = \mathcal{L}_{B_{max}}(t)$.
- If estimate improves, increase r and repeat.