Measurement and Modelling I:
End-to-End Bandwidth Measurements

6.829 Lecture 11

October 10, 2002

Jacob Strauss

End-to-End Measurements

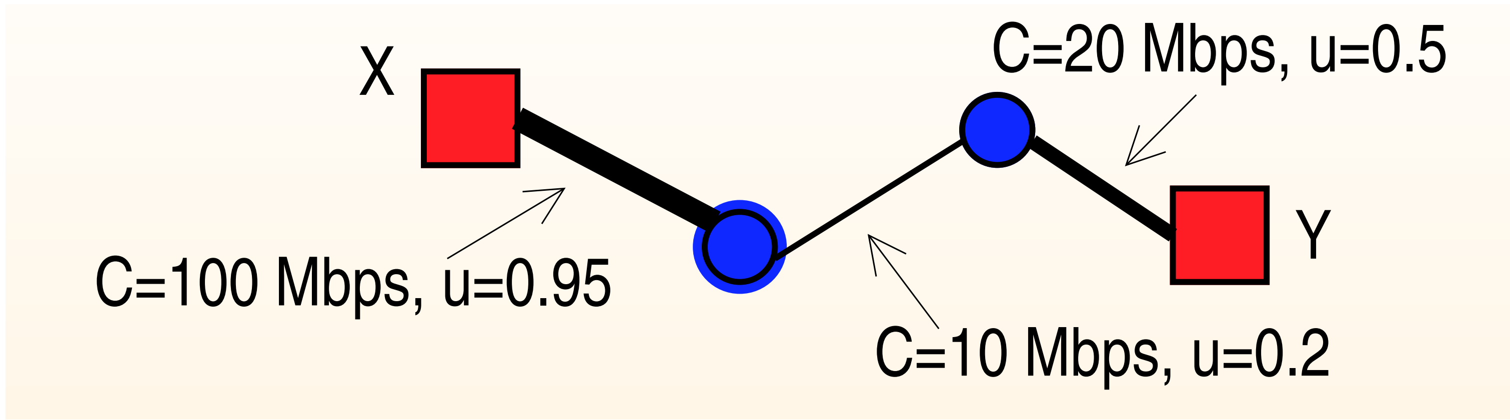
Goal: Bandwidth-Aware Applications

- Is a path fast enough?
- Which path is faster?

What is bandwidth?

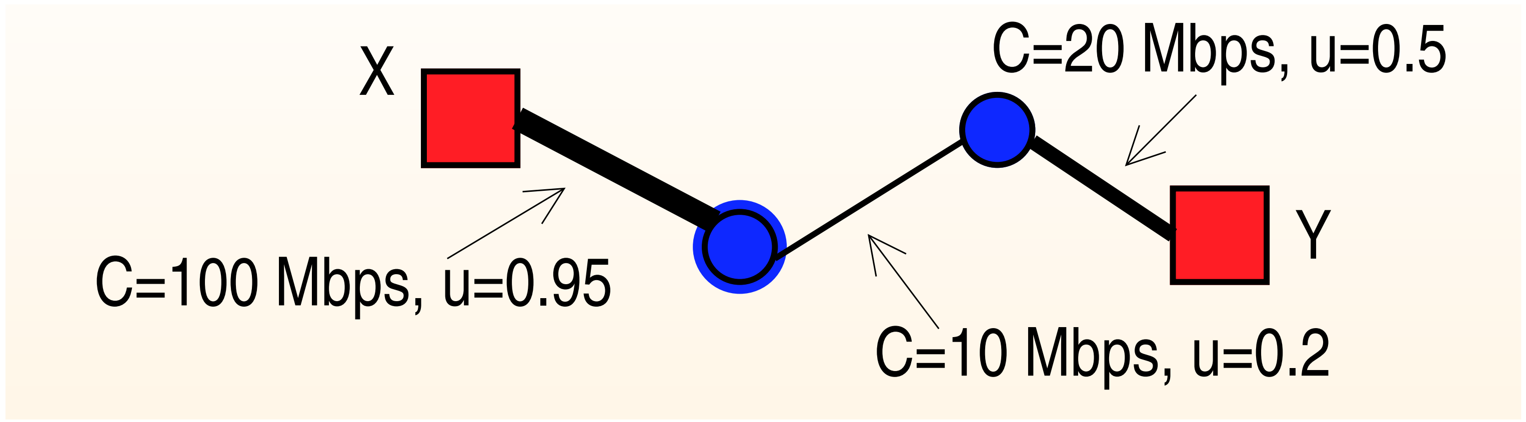
- *Link Capacity*: physical link speed (e.g. 100 Mbps Ethernet)
  - ▷ for link i, link capacity c_i.
- *Path Capacity*: minimum capacity link along a path
  - ▷ $C = \min_{i=0\ldots n}(c_i)$
- *Link Utilization*: fraction of link capacity used over some time interval
  - ▷ $0 \leq u_i \leq 1$
- *Available Bandwidth*: minimum spare capacity along a path over some interval
  - ▷ $A = \min_{i=0\ldots n}(1 - u_i)c_i$

Example Topology



Capacity from X to Y?

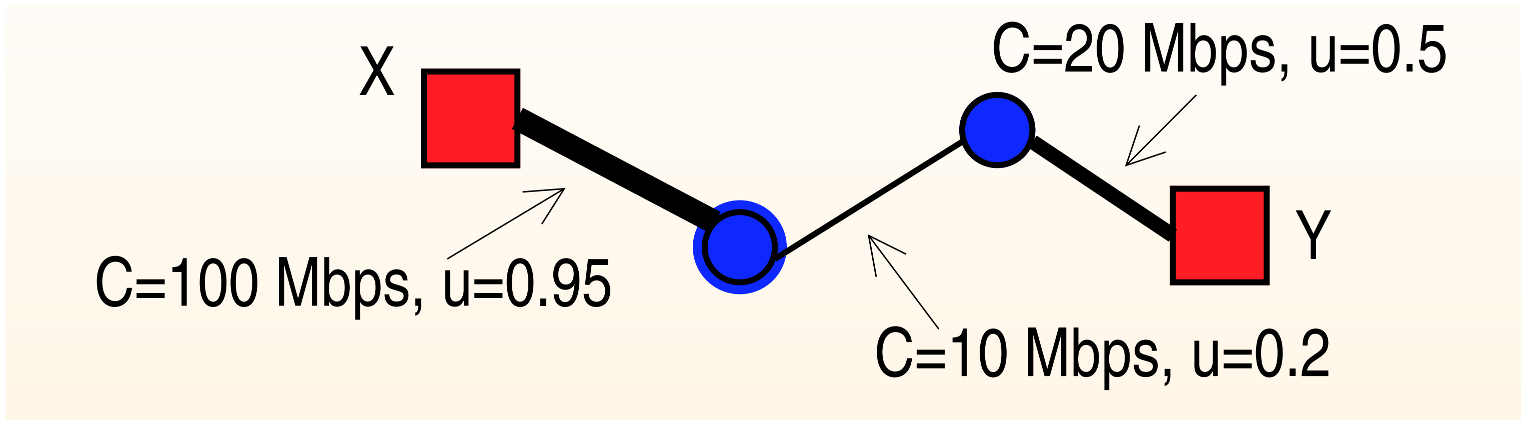
Example Topology



Capacity from X to Y? $C = 10Mbps$

Available bandwidth from X to Y?

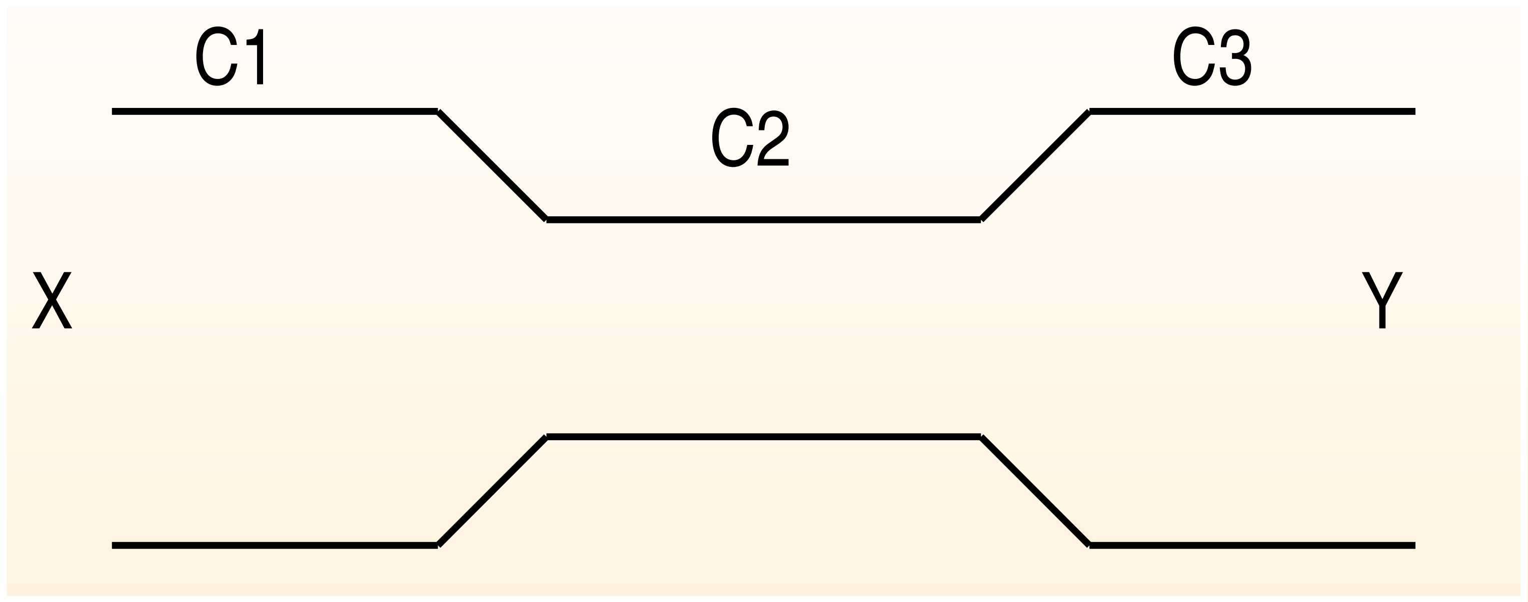
Example Topology



Capacity from X to Y? $C = 10Mbps$

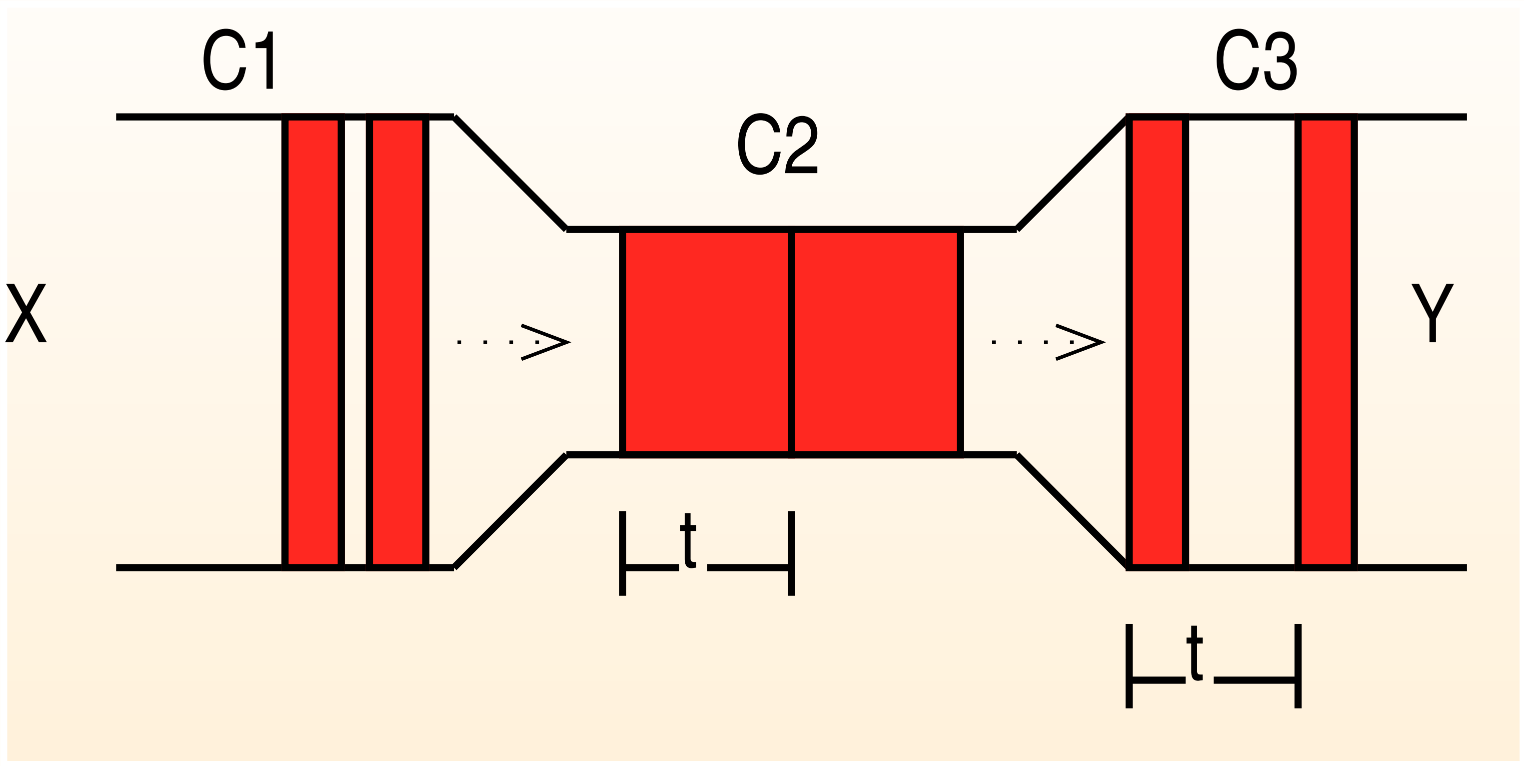
Available bandwidth from X to Y? $A = 5Mbps$

Problem: Capacity Estimation



How can we measure the path capacity between X and Y?

Packet Pair on an unloaded path

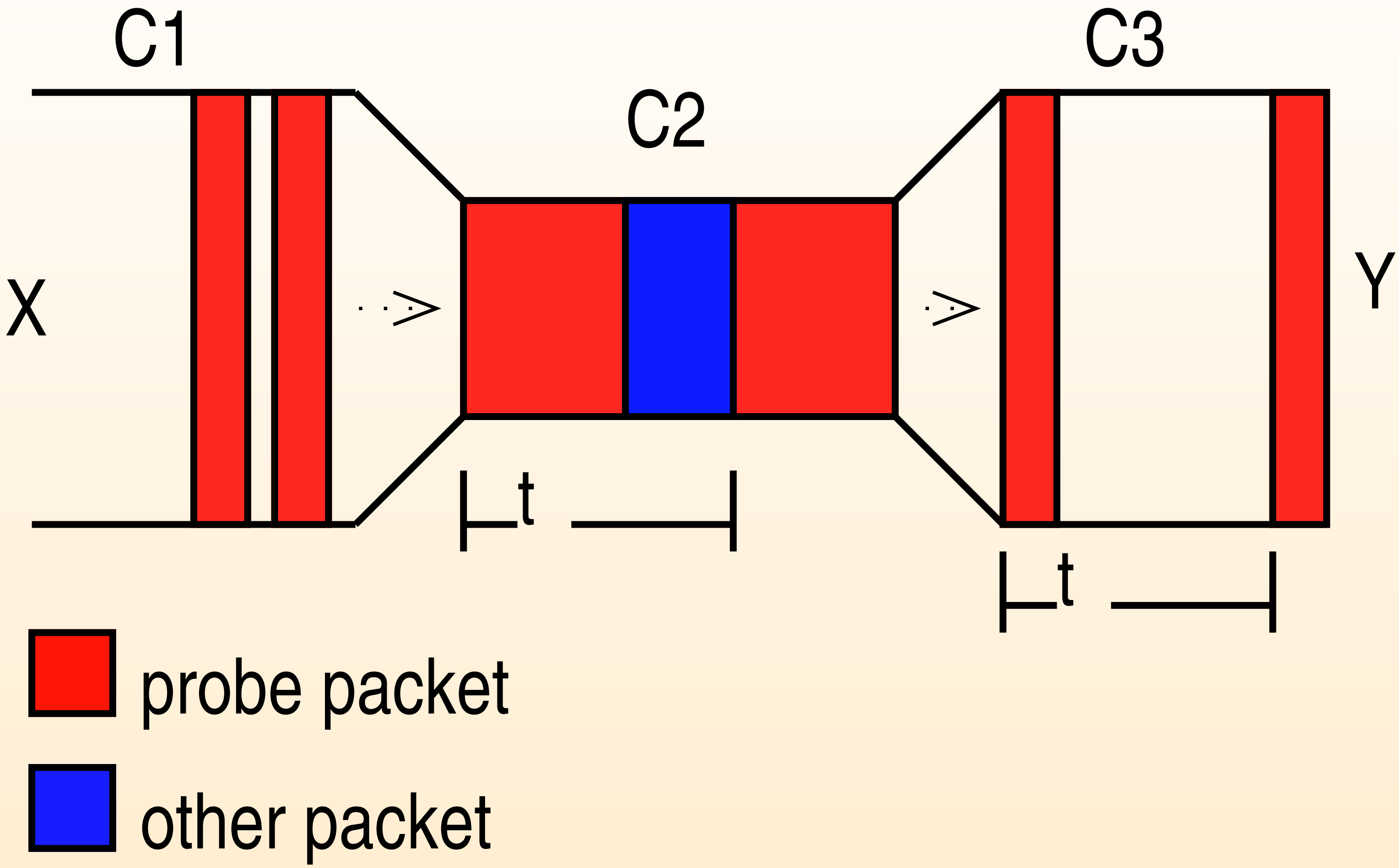


Send two packets, of P bits each, back-to-back

Record the difference between arrival times, Δt.

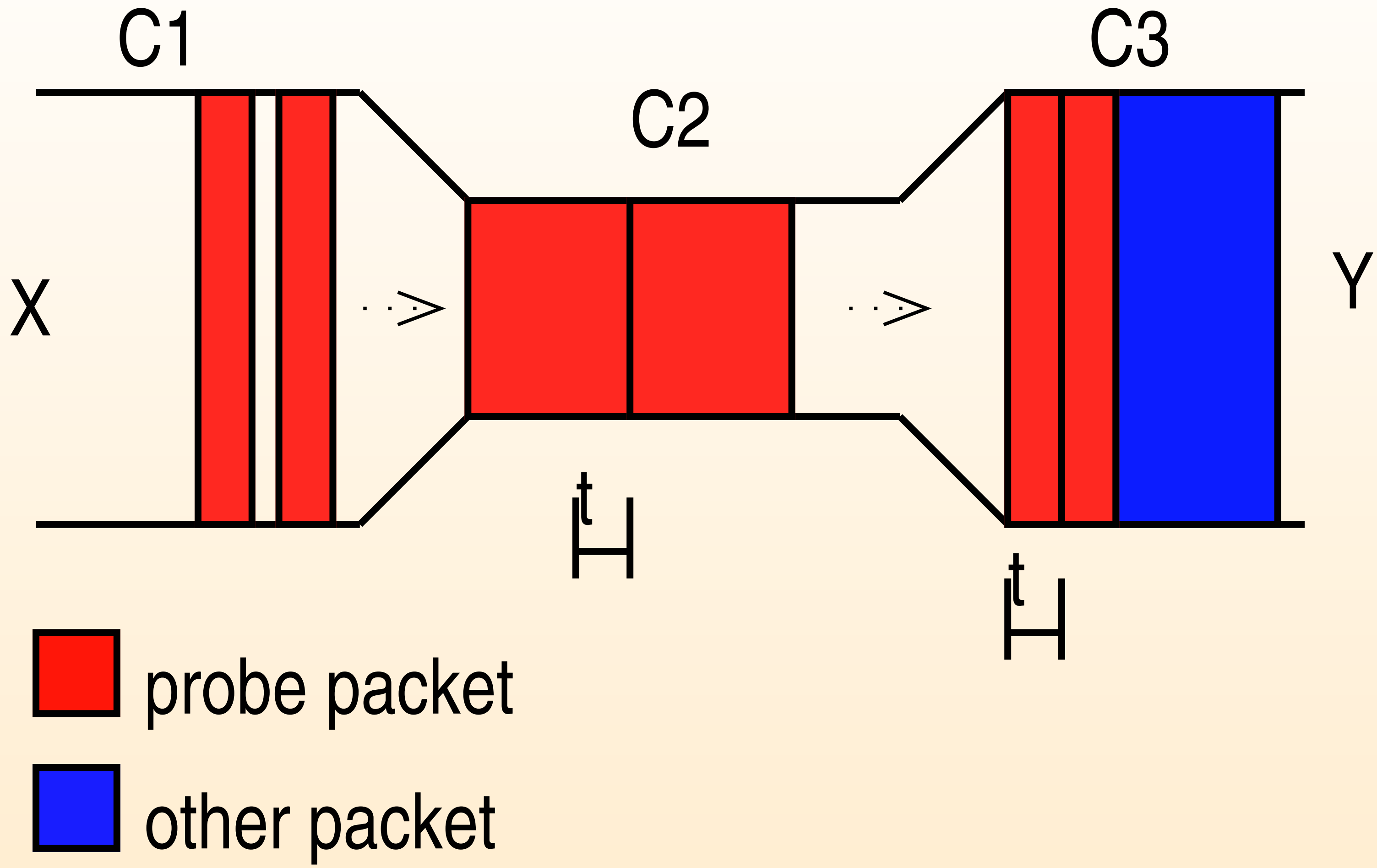
Path Capacity is $\frac{P}{\Delta t}$

Packet Pair Complications: Cross Traffic



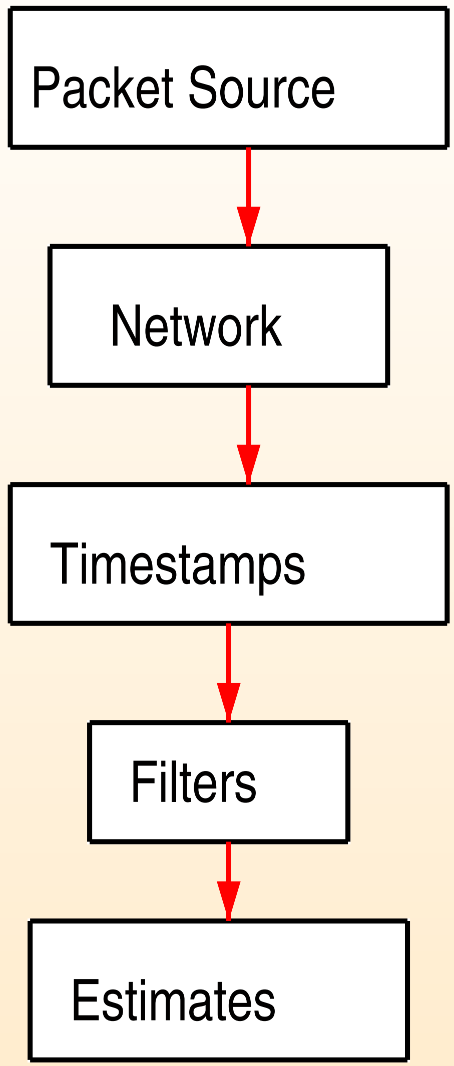
$$\Delta t > \frac{P}{C}$$

Packet Pair Complications: Multiple Queues



$$\Delta t < \frac{P}{C}$$

Delay Measurements Block Diagram



New Problem: Path capacity and packet size distribution

Long train of packets with size P, sent every δ seconds

Record either one-way-delay or round-trip time for each packet

- RTT is easier to measure. Why?

Use phase plots:

- For each n, plot a point at $(rtt_n, rtt_n + 1)$

Delay Components & Model

Components

- Propagation delays
- Queuing delays
- Processing delay (lookup & scheduling)
- Transmission delay

Model assuming a single queue

- Fixed delay: D
- waiting time: w_n
- service time: $y_n = \frac{P}{\mu}$
- total delay: $rtt_n = D + w_n + \frac{P}{\mu}$

Explain the Plot: light load conditions

Assume that the tight link load is low:

- $w_{n+1} = w_n + \epsilon_n$
  $\Rightarrow rtt_{n+1} = rtt_n + \epsilon_n$

Explain the Plot: light load conditions

Assume that the tight link load is low:

- $w_{n+1} = w_n + \epsilon_n$
  $\Rightarrow rtt_{n+1} = rtt_n + \epsilon_n$

Points plotted near the line $y = x$ above $(x, y) = (D, D)$

Explain the Plot: heavy load conditions

Assume B bits in between probe packets n and $n+1$

$$w_{n+1} = w_n + \frac{B}{\mu} - \delta$$

Explain the Plot: heavy load conditions

Assume B bits in between probe packets n and $n+1$

$w_{n+1} = w_n + \frac{B}{\mu} - \delta$

Now assume $\frac{B}{\mu} \gg \delta$

- We have probe compression: $k-1$ packets arrive before B clears the queue
- These k packets depart every P/μ seconds

for the k packets: $w_{n+i} - w_{n+i-1} = P/\mu - \delta$

- why is $P/\mu - \delta < 0$?

Explain the Plot: heavy load conditions

Assume B bits in between probe packets n and $n+1$

$w_{n+1} = w_n + \frac{B}{\mu} - \delta$

Now assume $\frac{B}{\mu} \gg \delta$

- We have probe compression: $k-1$ packets arrive before B clears the queue
- These k packets depart every P/μ seconds

for the k packets: $w_{n+i} - w_{n+i-1} = P/\mu - \delta$

- why is $P/\mu - \delta < 0$?

result is $rtt_{n+1} = rtt_n + (P/\mu - \delta)$

Explain the Plot: heavy load conditions

Assume B bits in between probe packets n and $n+1$

$w_{n+1} = w_n + \frac{B}{\mu} - \delta$

Now assume $\frac{B}{\mu} \gg \delta$

- We have probe compression: $k-1$ packets arrive before B clears the queue
- These k packets depart every P/μ seconds

for the k packets: $w_{n+i} - w_{n+i-1} = P/\mu - \delta$

- why is $P/\mu - \delta < 0$?

result is $rtt_{n+1} = rtt_n + (P/\mu - \delta)$

We now know the bottleneck capacity μ from the plot.

Packet size distribution

Assume that between packets n and $n+1$, b_n bits from other flows join queue

- Use Lindley's recurrence:
  $w_{n+1} = (w_n + y_n - \delta_n)^+$, where x^+ means $\max(x, 0)$

Packet size distribution

Assume that between packets n and $n+1$, b_n bits from other flows join queue

- Use Lindley's recurrence:
  $w_{n+1} = (w_n + y_n - \delta_n)^+$, where x^+ means $\max(x, 0)$

Apply twice:

$$\begin{aligned} wb_n &= (w_n + P/\mu - \delta_b)^+ \\ w_{n+1} &= \big((w_n + P/\mu - \delta_b)^+ + b_n/\mu - (\delta - \delta_b)\big)^+ \end{aligned}$$

Packet size distribution

Assume that between packets n and $n+1$, b_n bits from other flows join queue

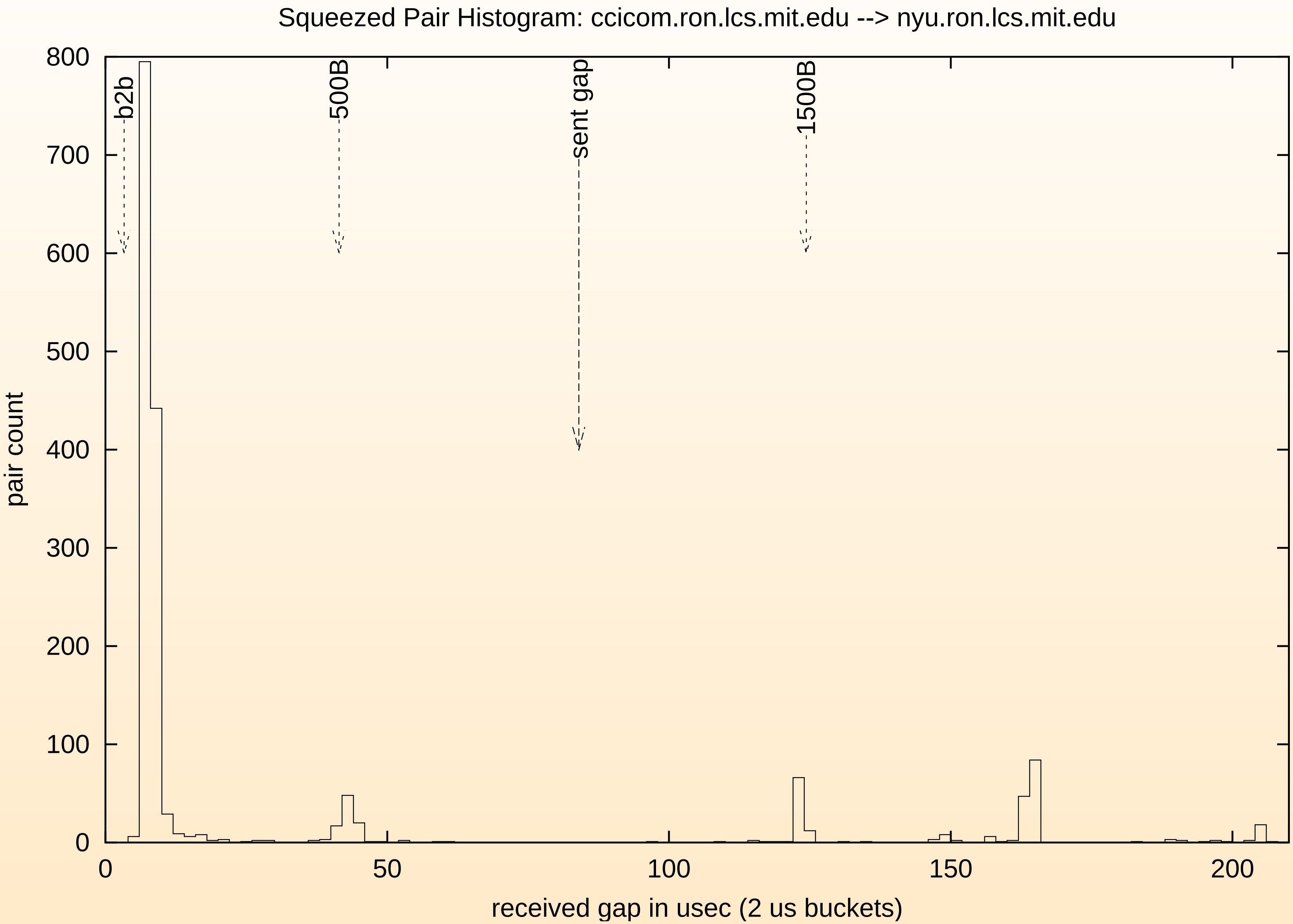
- Use Lindley's recurrence:
  $w_{n+1} = (w_n + y_n - \delta_n)^+$, where x^+ means $\max(x, 0)$

Apply twice:

$$\begin{aligned} wb_n &= (w_n + P/\mu - \delta_b)^+ \\ w_{n+1} &= \big((w_n + P/\mu - \delta_b)^+ + b_n/\mu - (\delta - \delta_b)\big)^+ \\ w_{n+1} &= w_n + (P + b_n)/\mu - \delta \\ b_n &= \mu(w_{n+1} - w_n + \delta) - P \end{aligned}$$

Now we can find packet sizes from peaks in PDF of $w_{n+1} - w_n + \delta$

Example Distribution Plot



New Problem: Available Bandwidth Estimation

Can we use the same packet train to estimate A?

New Problem: Available Bandwidth Estimation

Can we use the same packet train to estimate A?

No, but we can determine whether $P/\delta > A$ or $P/\delta < A$

- Later we'll build on this test to estimate A

New Problem: Available Bandwidth Estimation

Can we use the same packet train to estimate A?

No, but we can determine whether $P/\delta > A$ or $P/\delta < A$

- Later we'll build on this test to estimate A

Basic idea: instead of just comparing w_{n+1} to w_n, compare w_n to $w_{n+1}, w_{n+2}, w_{n+3}, \ldots$

Method: Plot evolution of w_n versus n

Pathload Delay Model

The packet train consists of K packets of size L sent at a constant rate R.

Path consists of H links, each with capacity C_i, available bandwidth A_i, and queue length q_i^k when the kth probe packet arrives

Pathload Delay Model

The packet train consists of K packets of size L sent at a constant rate R.

Path consists of H links, each with capacity C_i, available bandwidth A_i, and queue length q_i^k when the kth probe packet arrives

The One-Way-Delay for probe k is:

$$D^k = \sum_{i=1}^{H} \left(\frac{L}{C_i} + \frac{q_i^k}{C_i} \right)$$

Pathload Delay Model

The packet train consists of K packets of size L sent at a constant rate R.

Path consists of H links, each with capacity C_i, available bandwidth A_i, and queue length q_i^k when the kth probe packet arrives

The One-Way-Delay for probe k is:

$$D^k = \sum_{i=1}^{H} \left(\frac{L}{C_i} + \frac{q_i^k}{C_i} \right)$$

$$\Delta D^k \equiv D^{k+1} - D^k = \sum_{i=1}^{H} \frac{\Delta q_i^k}{C_i}$$

Increasing Trends

If $R > A$, then $\Delta D^k > 0$ for $k \geq 1$

If $R \leq A$, then $\Delta D^k = 0$ for $k \geq 1$

Increasing Trends

If $R > A$, then $\Delta D^k > 0$ for $k \geq 1$

If $R \leq A$, then $\Delta D^k = 0$ for $k \geq 1$

Intuition for a single link i with $A_i < R$:

- Queue grows longer with each probe packet that arrives
- $\Delta q_i^k = (L + u_i C_i T) - C_1 T = (R - A_i)T > 0$

For a single link with $A_i \geq R$:

- $\Delta q_i^k = 0$

By induction, show that $\Delta D^k > 0$ when $R > A$

Increasing Trends

If $R > A$, then $\Delta D^k > 0$ for $k \geq 1$

If $R \leq A$, then $\Delta D^k = 0$ for $k \geq 1$

Intuition for a single link i with $A_i < R$:

- Queue grows longer with each probe packet that arrives
- $\Delta q_i^k = (L + u_i C_i T) - C_1 T = (R - A_i) T > 0$

For a single link with $A_i \geq R$:

- $\Delta q_i^k = 0$

By induction, show that $\Delta D^k > 0$ when $R > A$

Plot D^k – if $\Delta D^k > 0$ then $R > A$

Complications

A_i not a constant

- varies on both short and long time scales

Need to choose K and L carefully

- Too short – can't tell if $\Delta D^k > 0$
- Too long – flood link
- Compromise – use multiple trains

How did we choose R initially?

Iterative Search for A

maintain two state variables: R^{max} and R^{min}.

always have $R^{min} \leq A \leq R^{max}$.

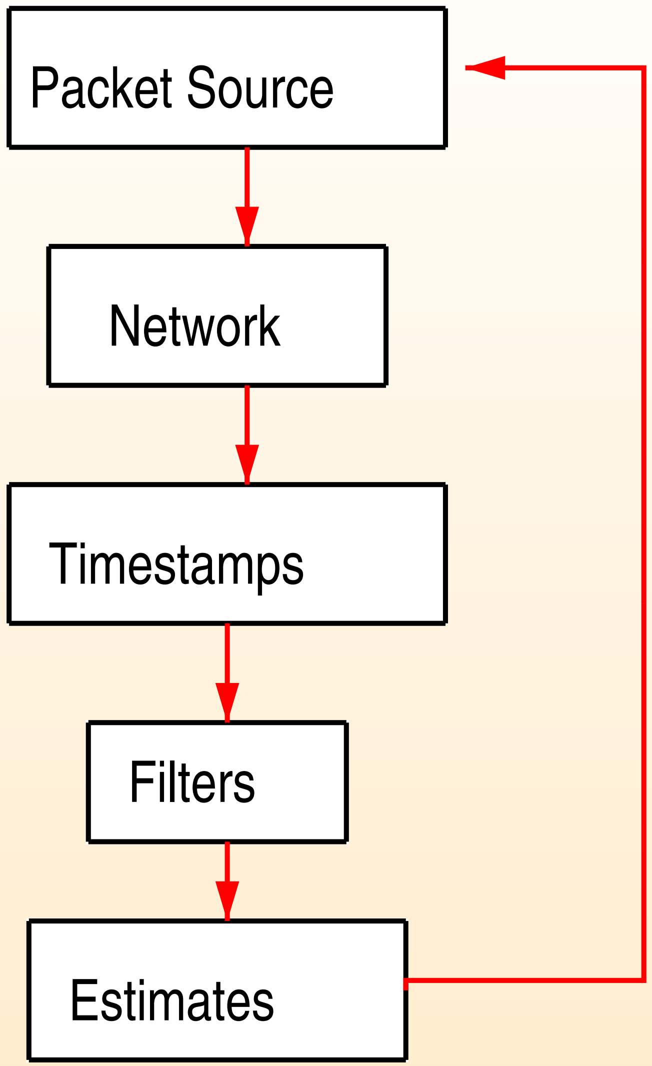
Pick a new R halfway between R^{max} and R^{min}

Test whether $R <> A$, update either R^{max} or R^{min}

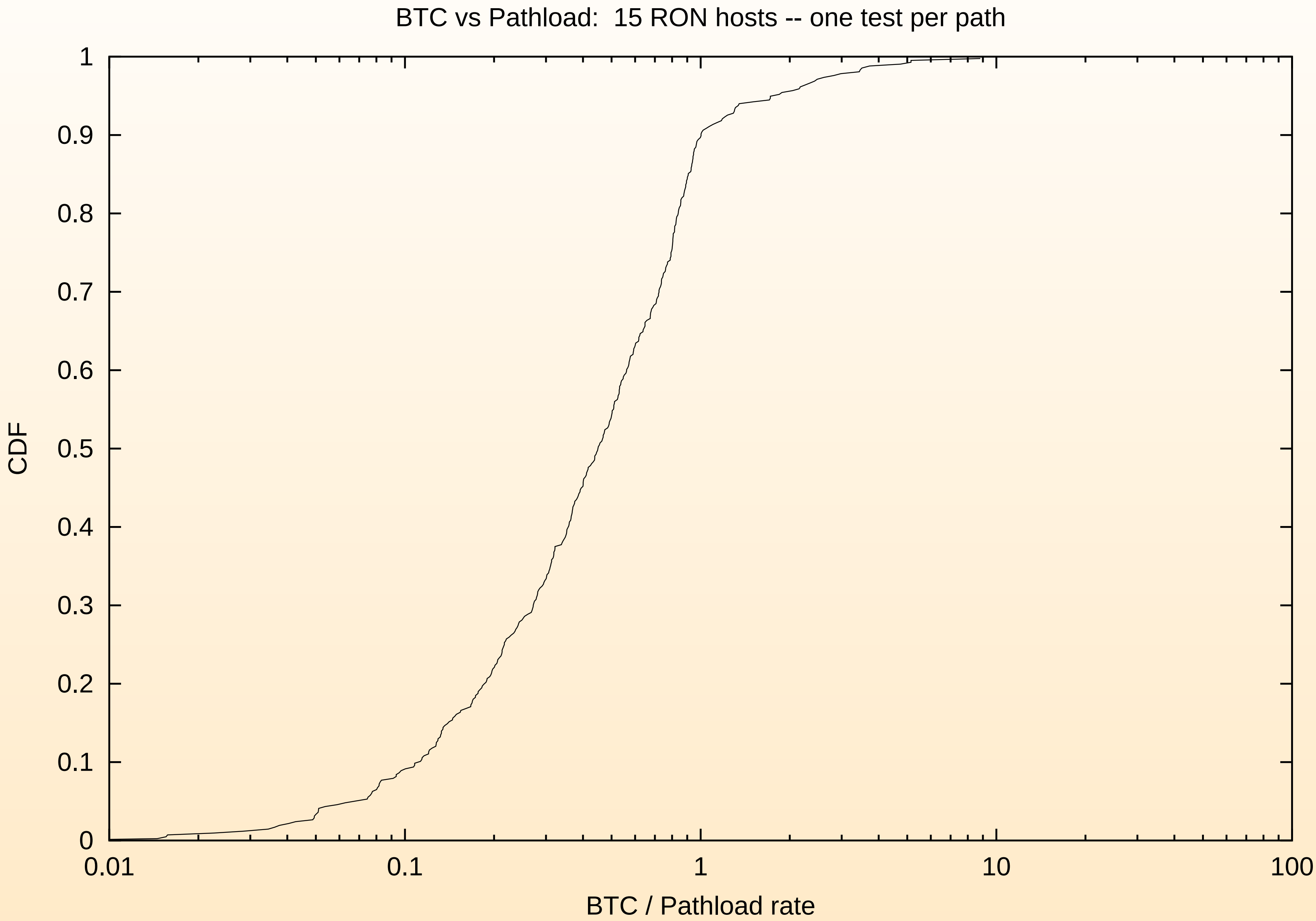
Stop when R^{max} and R^{min} are close enough.

In practice the search is more complicated because the outcome of testing $R <> A$ is sometimes unsure

Pathload Block Diagram



Available Bandwidth isn't the full story



Conclusion

There's a lot you can learn with simple probes

- packet pair
- packet trains — regular interval, constant packet size

Just by looking at packet delay variations you can determine

- Path Capacity
- Common Packet Sizes
- Available Bandwidth