ADAPTING TCP/IP TO ATM NETWORKS

Douglas Niehaus, Furquan Ansari

EECS Department
University of Kansas
Lawrence, KS
niehaus@eecs.ukans.edu

March 18, 1997

Organization

- Motivation
- Modifications to adapt TCP to ATM
- Selected Results
- Conclusions and Future Work

Causes for Poor TCP Performance over ATM Networks

- TCP - has undergone many modifications and fine tuning over the years
  which makes it fragile.
- TCP - Packet based technology; ATM - Cell based technology
  Clash of model assumption.
- Very high operating speeds.
  Performance depends on the *bandwidth-delay* product.
  Retransmission timeout granularity inadequate
  Performance impact of idle periods.
- Very Bursty. Transmits data within a window as rapidly as it can - leads to congestion.
  Stresses buffers - buffer overflow and packet loss
- TCP assumes every loss occurs due to congestion and cuts down window.

Modifications to Adapt TCP to ATM

- TCP Packet level traffic shaping:

  Make a traffic description possible.

  Network determines if the traffic should be allowed.

  Network can monitor and penalize misbehaving traffic.

- Shaping reduces traffic burstiness and associated buffer problems.
- Both *how much* data is sent and *how* it is sent - important.
- Aggressive feedback retransmission policy.

  Works when multiple packets are lost within a window.

  Assumes in-order delivery - optimized case.

  Advantage: Reduces idle periods.

- Large Windows required for large bandwidth-delay product links.

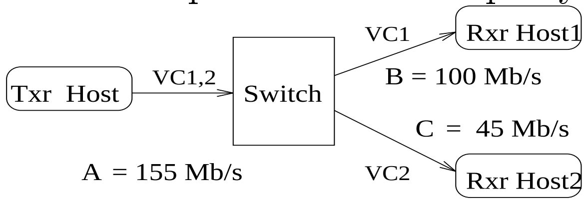
  Necessary but not sufficient for good performance.

- Eliminate TCP layer data checksumming - redundant function.

  CRC error detection provided by the AAL5 layer.

TCP (MTU level) Traffic Shaping

- Cell pacing - in Hardware. Expensive and at the lowest level of abstraction
- Requires Large buffer space on NICs.
- Buffering memory duplicated between TCP window and NIC.
- Shaping policy types restricted to the one provided by the manufacturer.
- Can often shape at only one rate. Example shows inadequacy.

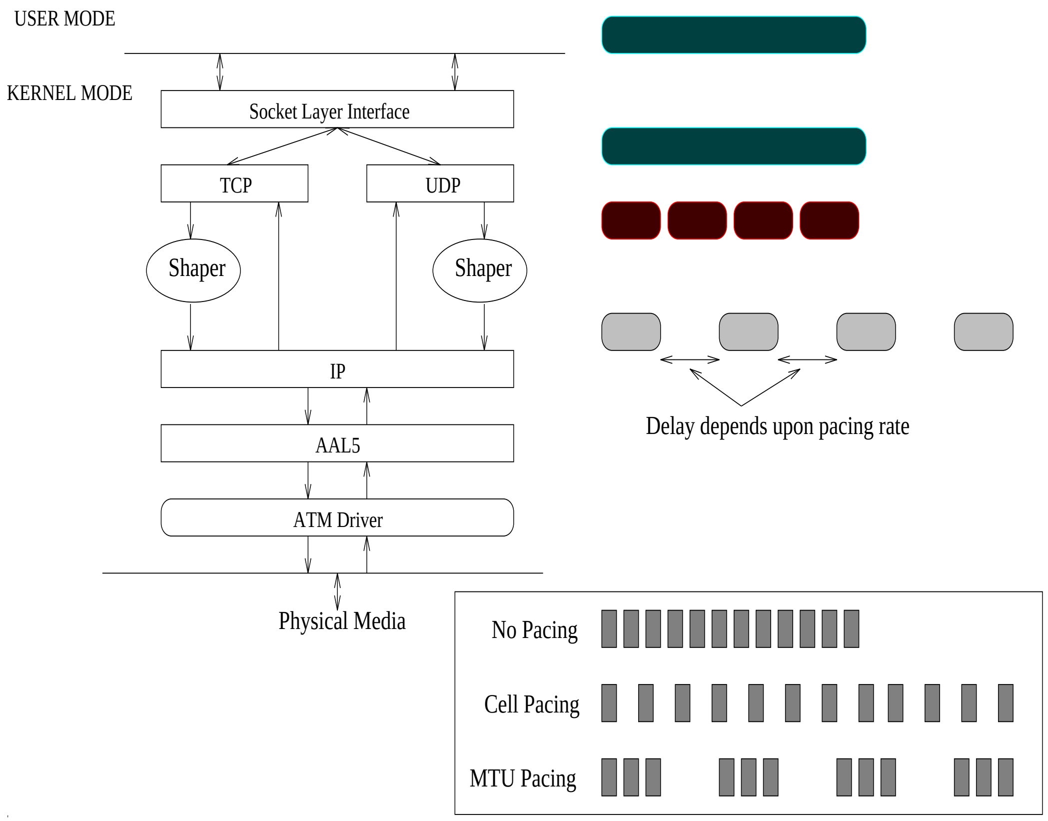


- More flexible hardware generally implies more expensive.
- TCP Packet pacing - packets (MTU) paced rather than cells.
- Packet(MTU) pacing transmits cells within MTU length at line speed.

  Switch buffer capacity *at least* 256 cells available.

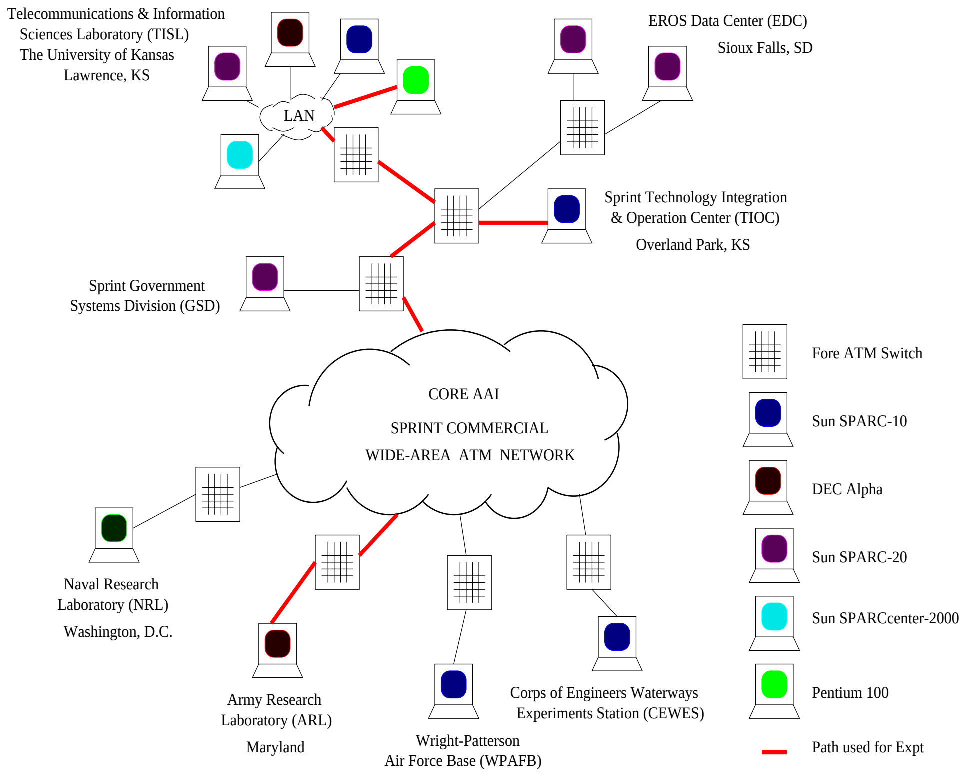
  MTU pacing - burst of *at most* 192 cells.

TCP Traffic Shaping (contd.)



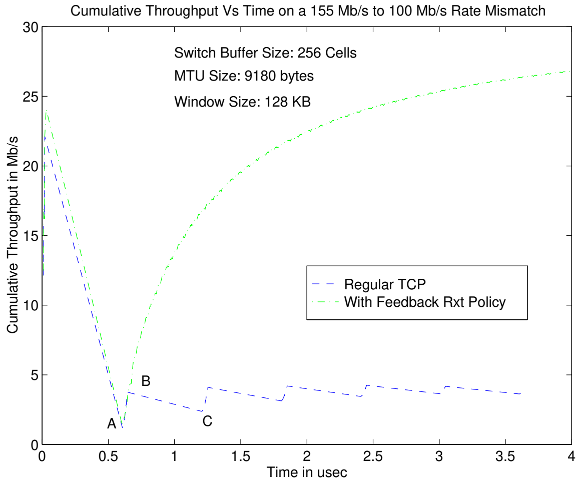
WAN Test Scenarios

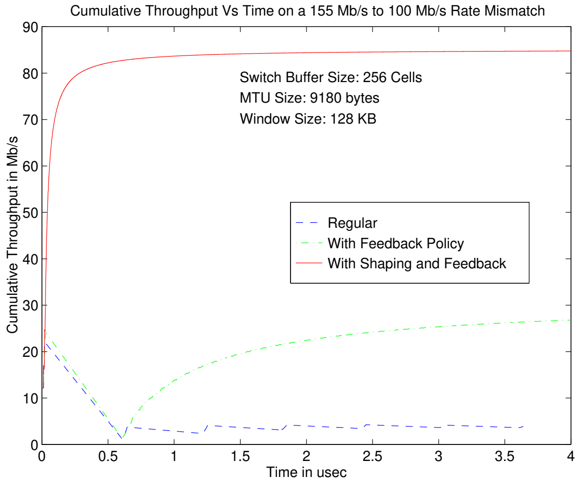
- WAN experimental Structure.



Effect of Feedback Retransmission Policy& Shaping

- Rate Mismatch Link - 155 Mb/s to 100 Mb/s.
- Bandwidth-Delay Product 37,000 Bytes. Delay ≈ 3ms. TISL, KU to Sprint TIOC.
- Cumulative Throughput $= \dfrac{\text{Total Num Of Bytes } \textit{Successfully} \text{ transmitted}}{\text{Current Time}}$
- Point 'A' - TCP Climb. Point 'B' - Packet Lost. Point 'C' - Retransmit timeout and another TCP climb.



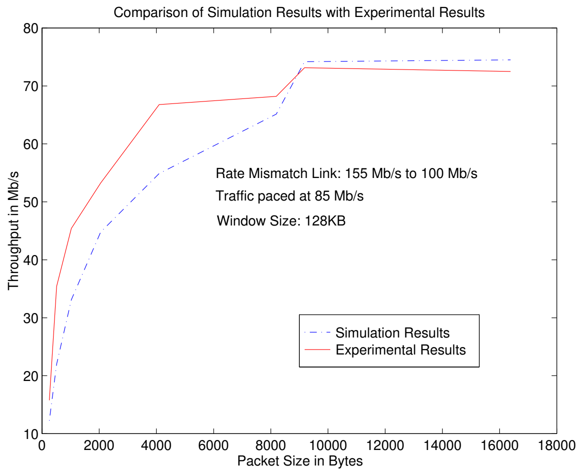


Validation of Simulation Results

- Validated with two *completely* different set of experiments.

- No Pacing Used.
- No Rate Mismatch employed.
- LAN Environment - Zeno to Phaedrus. Window size 64 KB.

- Rate Mismatch Link - 155 Mb/s to 100 Mb/s.
- WAN Environment - TISL, KU to Sprint TIOC. Window size 128 KB.

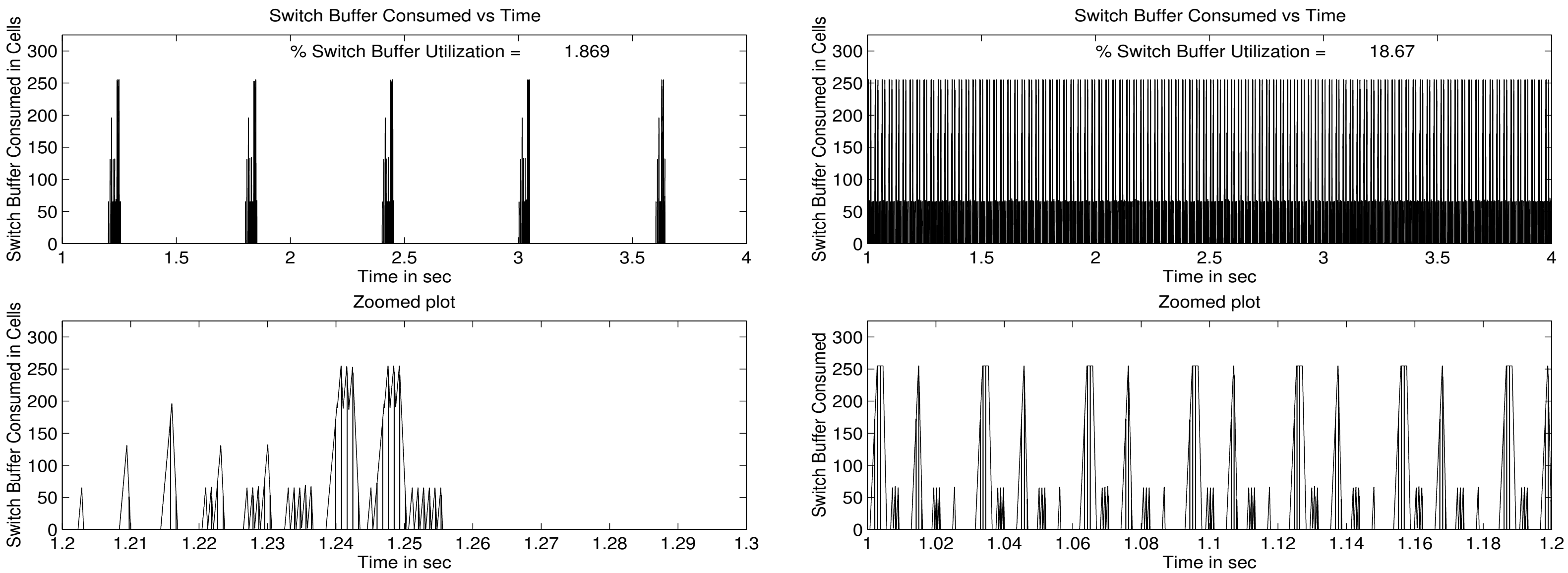


- Paced at 85 Mb/s.

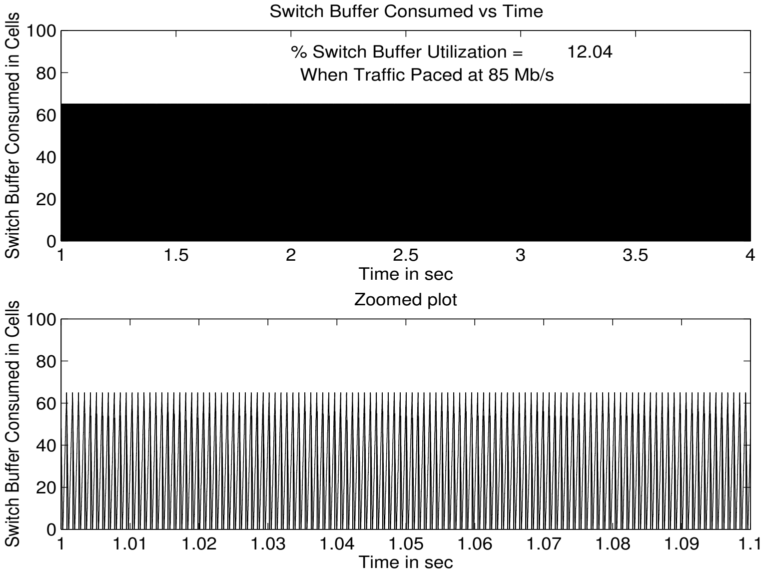
- Simulation results closely agree with the experimental results.

Switch Buffer Utilization

- Rate mismatch - 155 to 100 Mb/s.
- Switch Buffer Utilization (%) $= \dfrac{\text{Total Num Cells Consumed}}{\text{Maximum Cell Consumption}} * 100\%$



Switch Buffer Utilization (contd.)

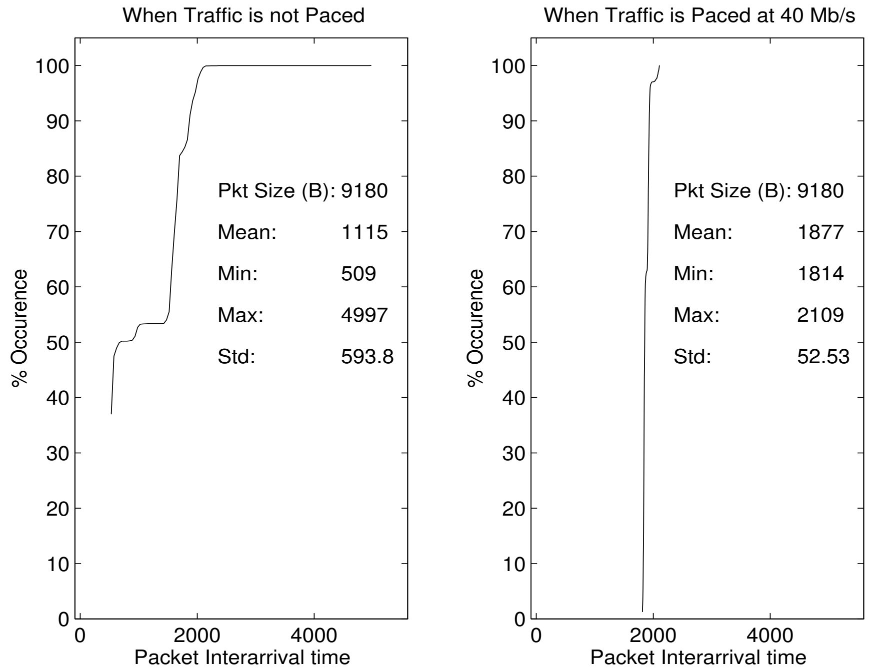


TCP	Thru'put (Mb/s)	Util (%)
Regular	3.915	1.869
With FB Policy	43.36	18.87
Shaping + FB / Shaping	84.76	12.035

- Regular TCP performs very badly. Low throughput and poor buffer utilization.
- Feedback Policy improves performance by an order of magnitude over regular TCP.
- Shaping provides best performance. High throughput with reduced buffer requirements.
- Feedback Policy works harder (Retransmissions) to achieve that throughput.

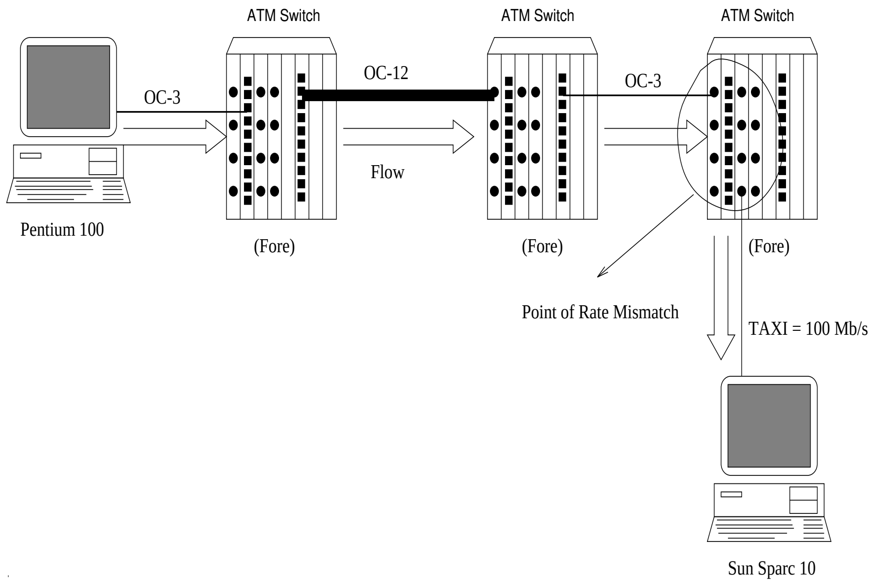
How Good is the Shaper?

- The shaper should be able to follow the specs closely.
- eg.: pacing MTU size packets (9180 bytes) at 40 Mb/s gives:
- Interspacing time $= \frac{9180*8*10^6}{40*10^6} = 1836\mu sec.$
- Datastream driver used to collect packet timestamps.



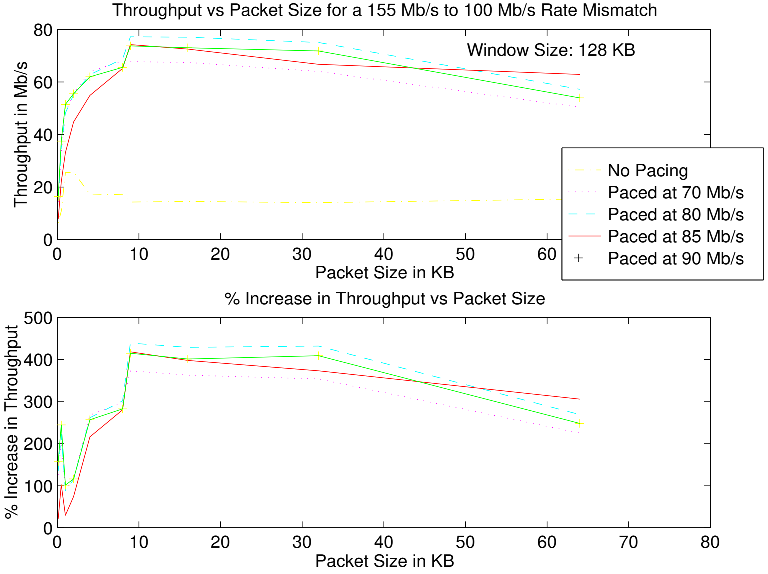
WAN Experiments

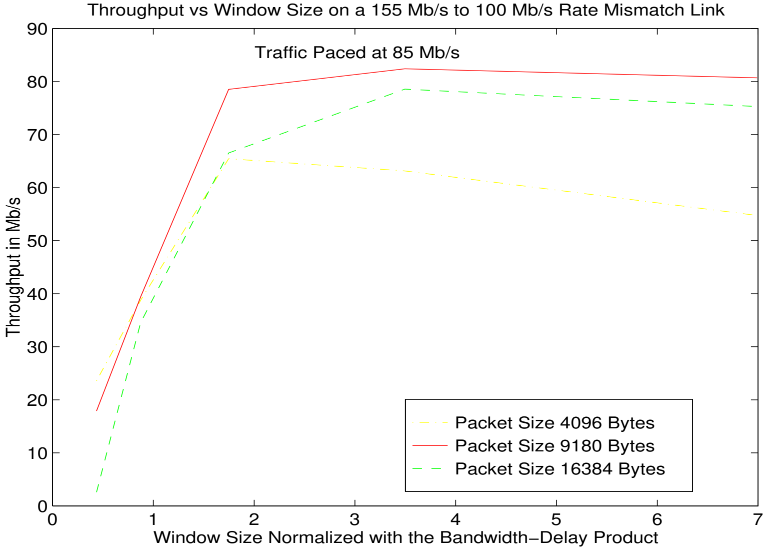
- Rate Mismatch - 155 Mb/s to 100 Mb/s - TISL, KU to TIOC, Sprint.
- Rate mismatch - implies service rate lower than arrival rate.
- Delay ≈ 3ms. Bandwidth-Delay product ≈ 37,500 Bytes.
- Switch buffer queue at rate mismatch drops cells.
- Frequent TCP retransmission timeouts leads to poor performance due to idle periods.



WAN Experiments (contd.)

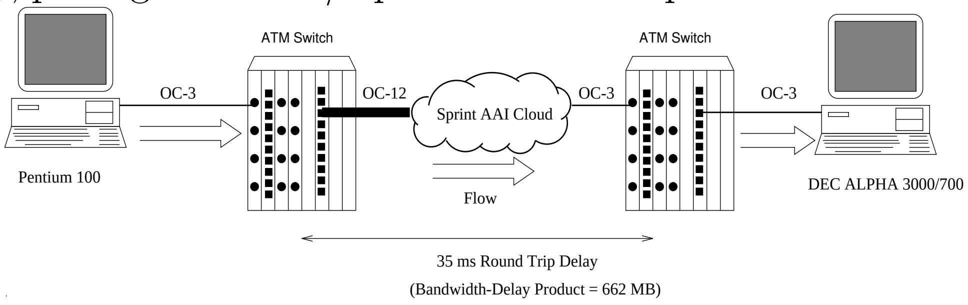
- Pacing reduces/eliminates cell loss at switch buffer.
- Pacing improves performance by as much as 400%.
- Window Size (beyond minimum necessary) has *little* effect on performance when traffic is paced.





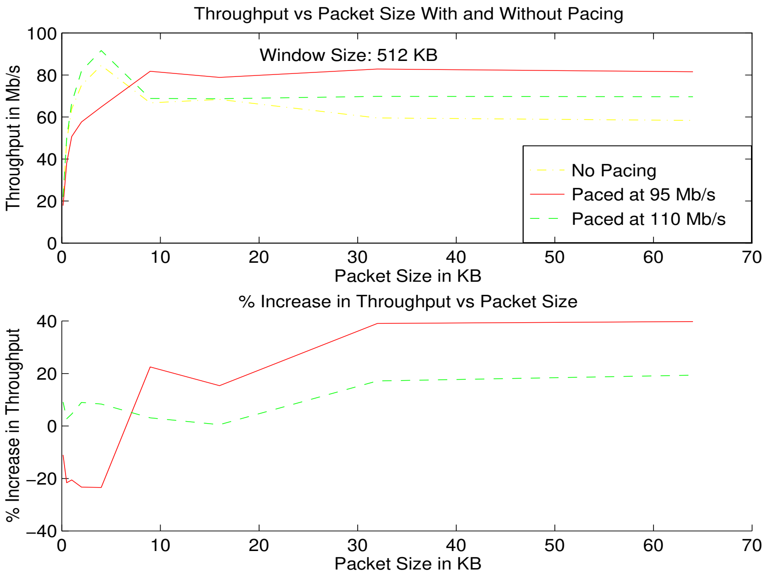
WAN Experiments (contd.)

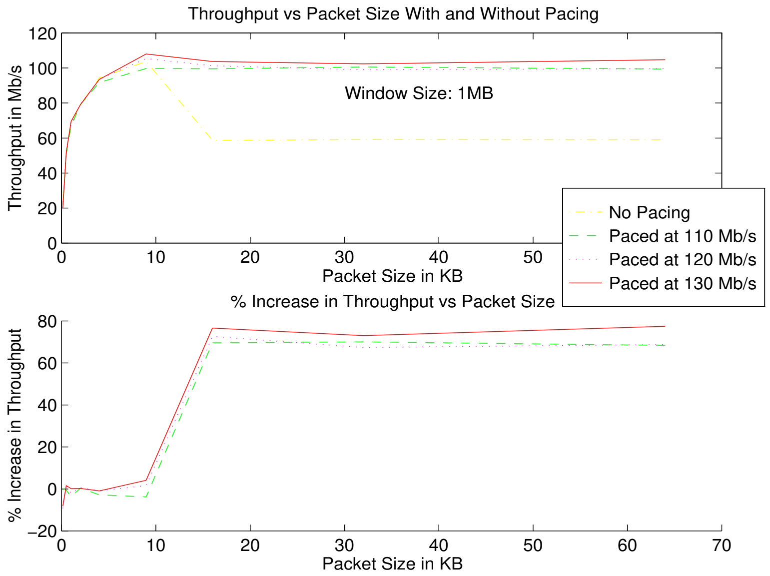
- Experiment over a *large bandwidth-delay product* network. TISL, KU to ARL, Maryland.
- Delay ≈ 35ms. Bandwidth-Delay product ≈ 662 KB.
- At least 7 ATM switches in the path.
- Pacing rate plays a large role when window size less than B-D product.
- For a window size of 512 KB (=0.77 B-D), pacing at 95 Mb/s provided better performance than at 110 Mb/s.



WAN Experiments (contd.)

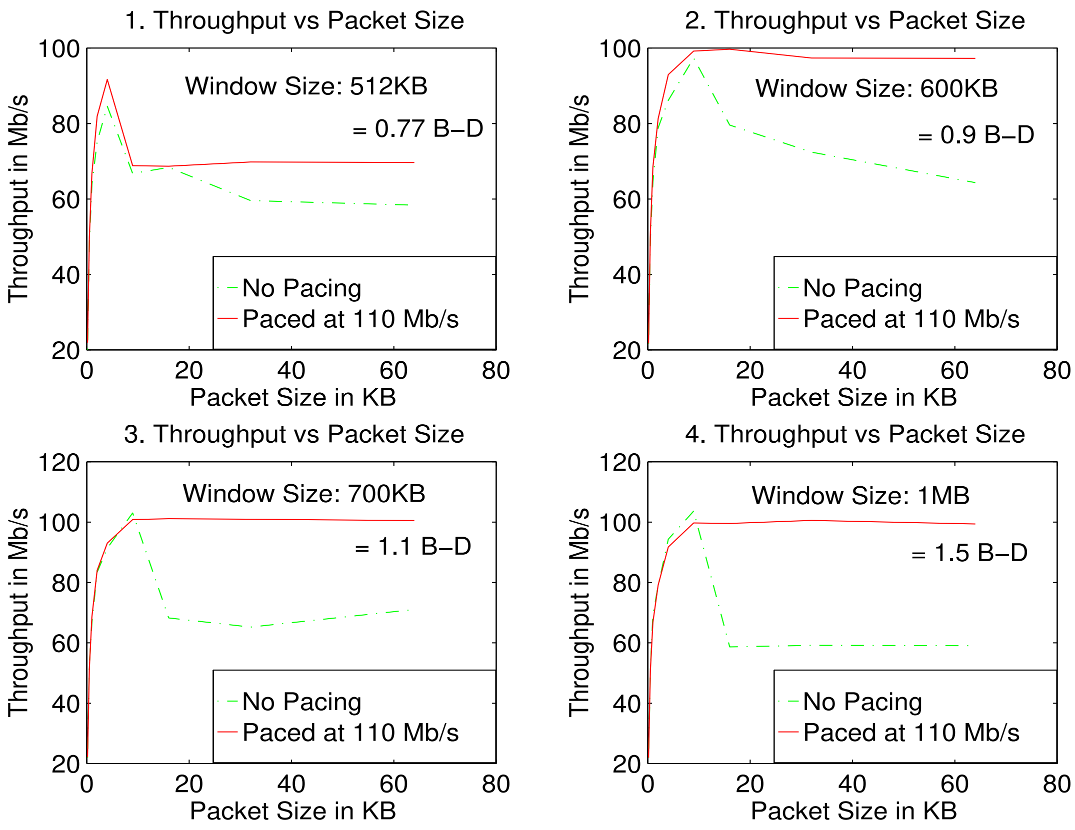
- Pacing can be looked at as increasing the window size in the time domain.
- Pacing provides constant throughput at all packet sizes above MTU size.
- Reason: Pacing is done at the MTU level, not packet level.





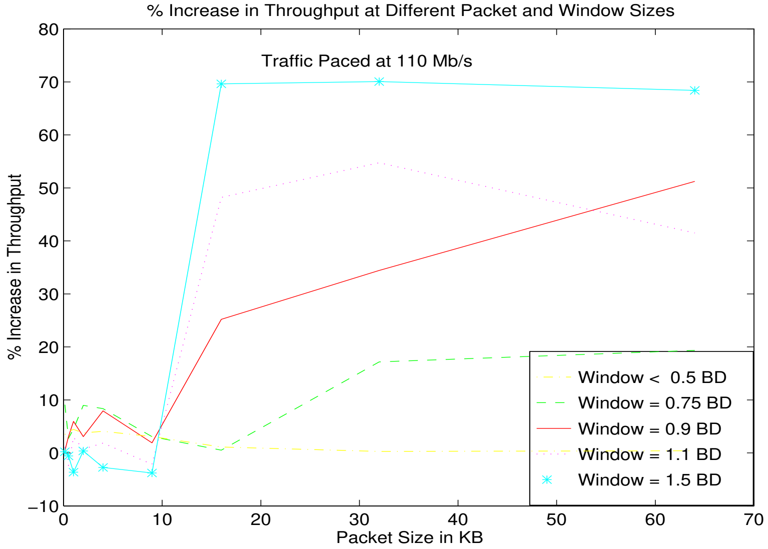
WAN Experiments (contd.)

- Standardized behavior when traffic is paced.

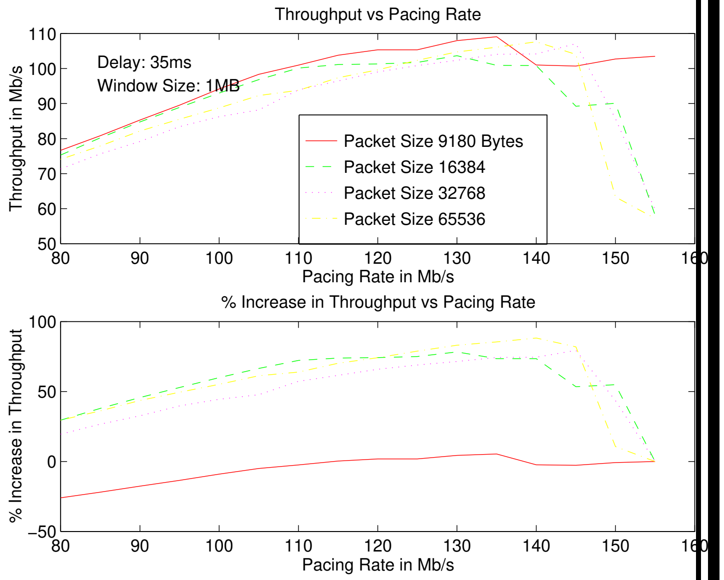


WAN Experiments (contd.)

- Effect of Window Size on throughput.
- At very small windows pacing does not help.
- Best performance gain when paced at or above B-D product window sizes.



- Effect of pacing on throughput
- Best performance seen at 135Mb/s to 145 Mb/s.
- For 9180 byte packet performance at 155 Mb/s remains almost same.



Conclusions

- TCP has been modified by many and its performance has become fragile.
- Requires fine tuning to find the right/optimal operating point. Small changes in parameters dramatically effect performance.
- Main cause for poor performance of TCP over ATM - Burstiness of TCP traffic and cell drops (by switch) belonging to multiple packets.

  Packet loss can be reduced by MTU level traffic shaping - smoother traffic.

  A shaping/pacing rate exists which provides best performance.
- Multiple packet losses can be recovered quickly using the feedback retransmission policy - without waiting for a retransmission timeout.

Conclusions (contd.)

- Simulations were used to show the effect of feedback retransmission policy and TCP traffic shaping on throughput.
- Feedback retransmission policy has higher switch buffer consumption because of retransmitted packets.
- Shaping helps achieve higher throughput with lower switch buffer consumption.
- In WAN - shaping *necessary* to reduce congestion and buffer overflow.

  In rate mismatch

  In environments where traffic flows through multiple switches.

  In cases where multiple streams are multiplexed onto a single stream.

Future Work

- Testing the feedback retransmission policy in WAN environment.
- Experimenting with new shaping algorithms and implementations
- Providing QoS guarantees to the application layer.

  Relationship between buffer allocation and pacing - during CAC.
- Multiple streams scenarios should be further investigated.