Improving TCP Performance Over Mobile Satellite Channels: The ACKPrime Approach

Keith Scott and Stephen Czetty
Jet Propulsion Laboratory, California Institute of Technology
http://eis.jpl.nasa.gov/~kscott

NASA Lewis Satellite Networks Workshop
June 1998

Outline

• TCP Over Mobile Satellite Links
  − Target Application: Once a packet crosses the satellite link it's gone forever.
  − Control Loop Includes Satellite Delay
• Ways of Breaking the Control Loop at the Groundstation
  − Proxy
  − I-TCP
  − Spoofing
  ≟ ACK'
• ACK' Implementation
  − Doesn't Break TCP End-2-End Semantics
  − Requires Few Resources at the Groundstation
  − Requires Minimal Changes to Sending TCP
  − ACK' Can Provide Corruption Notification at Little Extra Cost
  − IPSec Breaks ACK' Too
• Ack' Performance
  − Not as good as Proxying
  − Most Gain During Slow-Start
  − Increased ACK Traffic
  − Be Careful To Not Violate Congestion Control
TCP

- TCP is responsible for reliable, in-order, end-to-end delivery of information without duplications.
  - Number every byte; transmit bytes along with numbers, get acknowledgments from the receiver.
- Window-Based Flow Control & Congestion Control
  - Receiver’s Offered Window (Flow Control)
  - Sender’s Congestion Window (Congestion Control)
  - Sender can have at most MIN(cwnd, awnd) unacknowledged packets outstanding at any one time.
- Slow-Start
  - To keep a pair with a large awnd from injecting huge bursts of traffic into the network, cwnd starts at 1 and opens by 1 packet for every ACK received.
  - Sender sends 1 packet, waits for ACK, sends 2 packets, waits for ACKs, ...
- Congestion Avoidance
  - When a loss is detected, halve the sending rate and open cwnd by 1 packet for every window of data.
- Assumptions:
  - All losses are due to congestion within the network (i.e. overflows in router queues).
  - Delay * Bandwidth product is < 64k bytes (Can be circumvented)
  - Delay is small

TCP Slow-Start

![TCP Slow-Start Graph](image-url)

- Reno TCP
- 420ms RTT
- 1 Mbps Bottleneck Link

NASA Lewis Satellite Networks Workshop 6/98 KLS 3

NASA Lewis Satellite Networks Workshop 6/98 KLS 4

510
**TCP Over Mobile Satellite Links**

- Large BW * Delay Product
  - Use RFC 1323
- Higher BER
  - Use FEC
  - Depending on $f$, can have drop-outs of 10s to 100s of ms.
- Large Delays
  - Slow-Start is really slow.
  - Short transfers may never get out of slow-start.
  - The pipe refills at a rate of 1 packet per RTT during congestion avoidance. It can take 30s or more to recover from a loss when the session is using a geosynchronous satellite.
Target Application: End Users Of Satellite Links

- 250 ms (one-way)
- 20 ms (one-way)

Satellite RTT

Terrestrial RTT

Regular TCP RTT

Target Application Properties

- Once a TCP packet is forwarded towards the mobile, it has left the terrestrial network.
  - ACK is not designed for backhaul satellite links.
- Satellite channel contains most of the delay.
- Satellite channel has higher BER than terrestrial.
### Ways of Breaking The TCP Connection At The Groundstation

- **Proxy**
  - The mobile places a request with the proxy, the proxy executes the request and retrieves the information, the proxy passes the information back to the user.

- **Indirect-TCP (I-TCP)**
  - Similar to proxying; source sets up connection with intermediate node which terminates the connection and opens a new one to communicate with the destination.

- **Spoofing**
  - The groundstation/gateway actually acknowledges data flowing towards the mobile and suppresses acknowledgments from the mobile towards the server. The groundstation really should take responsibility for delivering packets it has acknowledged.

- **ACK’**
  - The groundstation/gateway provides extra information to the sender, in the form of ACKPrime’s.
    - Sender treats ACK’ like a regular acknowledgment for the purposes of increasing cwnd.
    - Mobile is still responsible for acknowledging data receipt.

### ACK’ Implementation

- Simulated a version of ACK’ in lbl’s network simulator (ns).
  - Modified snoop and NewReno elements to be an ACK’ gateway and an ACK’-capable sender.
  - Gateway keeps no state, it simply generates ACK’ packets whenever it forwards a TCP packet across the satellite link.
  - Topology includes 10Mbps terrestrial network with 10ms delay and 2Mbps satellite network with 200ms delay.
    - No contention for the terrestrial network or buffer space yet.
    - Assumes properly tuned windows & socket buffers ([http://www.psc.edu/networking/auto.html](http://www.psc.edu/networking/auto.html))

- Planned Improvements
  - Don’t violate congestion control!
  - Use ACK’ information along with regular acknowledgments to get (expensive) corruption notification.
  - Modified ACK’ scheme to reduce acknowledgment traffic
  - Ways around IPsec...
  - Plan a kernel implementation on JPL’s mobile satellite protocol testbed later this summer.
Performance During Slow-Start

Plain NewReno

ACK' NewReno

ns implementation

Source

Interface 10Mbps 20ms delay ACK' Agent

Groundstation

Satellite

Mobile

NASA Lewis Satellite Networks Workshop 6/98

KLS 11

KLS 12
Performance During Congestion Avoidance

Sequence #
0 1000 2000 3000 4000 5000
Packet Loss
0 5 10 15 20 25 Time

Plain NewReno

ACK' NewReno

526

5286

25.5s to refill pipe

14.5s to refill pipe

Sequence #
0 1000 2000 3000 4000 5000
Packet Loss
0 5 10 15 20 25 Time

Mobile Satellite Protocol Testbed

Simulated Mobile User With:
✓ SACK
✓ Window Scaling
✓ SCPS-TP
☐ TCP Vegas

Channel Emulator

Simulated Groundstation (Server or Gateway)
tcpxmp / tcptrace showing TCP behavior.

The Internet

NASA Lewis Satellite Networks Workshop 6/98

KLS 14