UNIVERSITEIT GENT

FACULTEIT INGENIEURSWETENSCHAPPEN

Design of a Protocol Booster:

an ARQ Error Control Scheme for Real-Time Media Streams over Burst-Error Channels

Abram Schoutteet, Bart Dhoedt, Ingrid Moerman, Piet Demeester

Overview

- The Nature of Errors and Channel Models
- Real-Time Media
- Error Control Techniques
- Protocol Boosters
- The Real-Time Media Protocol Booster
- Evaluation

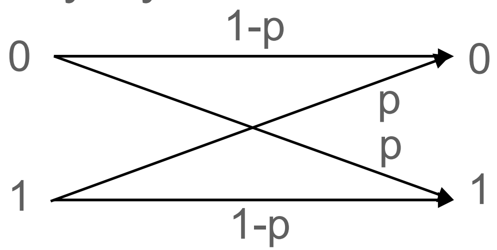
- **The channel**
  - **The medium over which information is conveyed**
    - Telephone lines
    - Fiber-optic lines
    - Microwave radio channels (e.g. IEEE 802.11 WLAN)
  - **As signals travel through a channel they are corrupted**
    - Additive noise
    - Attenuation due to propagation distance
    - Reflections
    - Absorption
    - Interference

Mathematical Channel models

- Gaussian Noise Channel

$$r(t) = s(t) + n(t) \qquad n(t) : N(0, \frac{N_0}{2})$$

- Binary Symmetric Channel



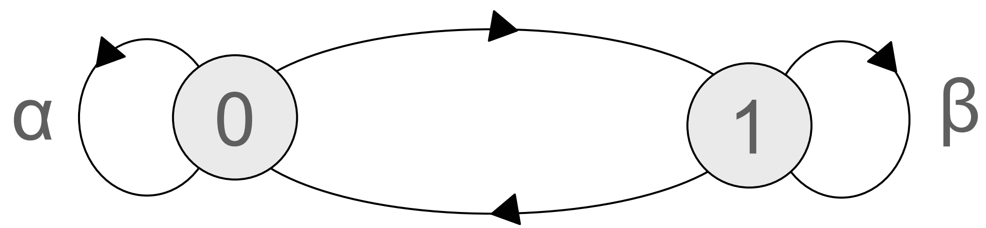
- Convolutive Channel

$$r_t = \sum_{i=0}^{L} s_{t-i} h_i + n_t$$

- Rayleigh Fading Channel

$$r(t) = g(t)s(t) + n(t) \qquad g_I(t), g_Q(t) : N(0, \sigma^2)$$

- **Mathematical models**
  - The Gilbert Channel
    - Obtained by analyzing the distributions of the lengths of error bursts and error gaps of wireless communication links.
    - Used to simulate the digital error performance of Burst-Error Channels



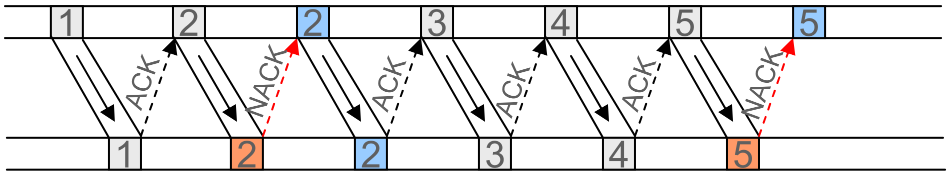
- State 0: $P_e(0) = 0$ State 1: $P_e(1)$
- $P(1) = \varepsilon = (1-\alpha)/(2-\alpha-\beta)$
- Average error rate = $\varepsilon P_e(1) + (1-\varepsilon) P_e(0) = \varepsilon P_e(1)$
- Simplification: $P_e(1) = 1$

Real-Time Media

- VoIP, Video Conferencing, virtual classroom, multiplayer gaming, …
- Require a timely delivery of data
- Can tolerate only a small amount of packet loss
- Use a play-out buffer to eliminate jitter introduced by the network
- This limited the window of time that can be used by error control techniques

Automatic Repeat Request Error Control

- Simple and highly reliable
- Large packet delays on links with a long RTT, reduced throughput on channels with a high error rate
- Addition of parity check bits for error detection
- If the syndrome, calculated at the receiver, is not zero, the presence of an error is detected and a retransmission is requested
- Basic ARQ schemes:
  - Stop and Wait ARQ

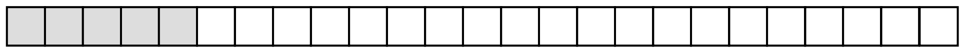


Error Control Techniques

- Forward Error Correction Error Control
  - Able to recover loss in less then an RTT for short periods of loss
  - Constant transmission overhead, correlated errors on burst-error channels (finite interleaving length)
  - Block Codes:
    - Each message (k-tuple) is mapped onto a codeword (n-vector)
    - BCH Codes, Reed-Solomon codes, Reed-Muller Codes, Golay Codes, …
  - Convolution Codes:
    - Can be viewed as a set of digital filters with the code sequence being the interleaved output of the filter outputs.
  - Iterative Decoded Codes:
    - Turbo Codes, Low-Density Parity-Check Codes

Hybrid ARQ: Incremental redundancy

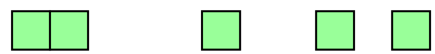
- A Rate (1,5) mother code



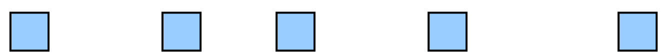
At the transmitter

- 1st transmission:

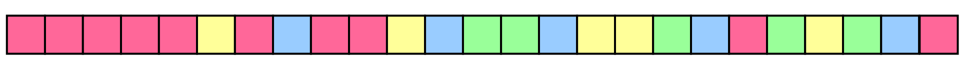
- 2nd transmission:



- 3rd transmission:



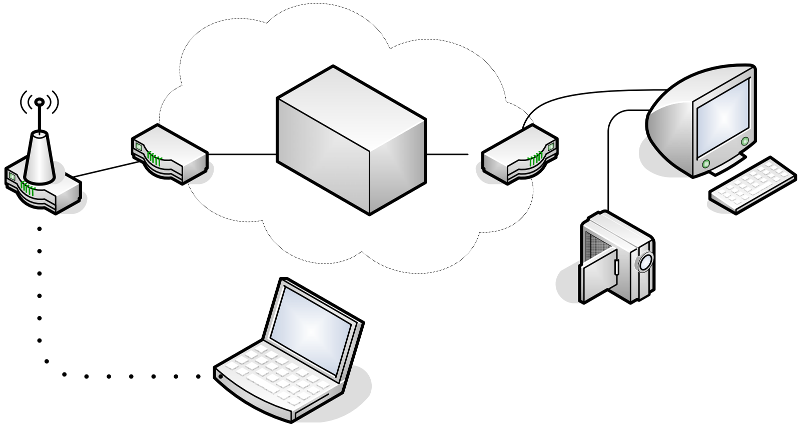
- 4th retransmission:



At the receiver

- **Methodology for protocol design**
  - Is a hardware or software module that transparently improves the performance of a protocol
  - It can reside anywhere in the network or end systems and may operate independently or in cooperation with other protocol boosters.
  - A protocol booster may add, delete or delay protocol messages but never originate, terminate or convert a protocol.
  - A multi-element protocol booster may define new messages to exchange between the elements but these are invisible and meaningless outside the booster element pair.
  - A protocol booster is transparent to the protocol being boosted and as such will not prevent the protocol from functioning if the booster is eliminated.

- In high speed WLAN-like links ARQ schemes perform better then FEC.
- Assumptions:
  - The booster will be deployed on commercial equipment and will not receive corrupted packets nor will be informed on packet loss.
  - The roundtrip delay between the two booster elements is small (compared to the delay constraints). E.g. the booster will be deployed in the access network



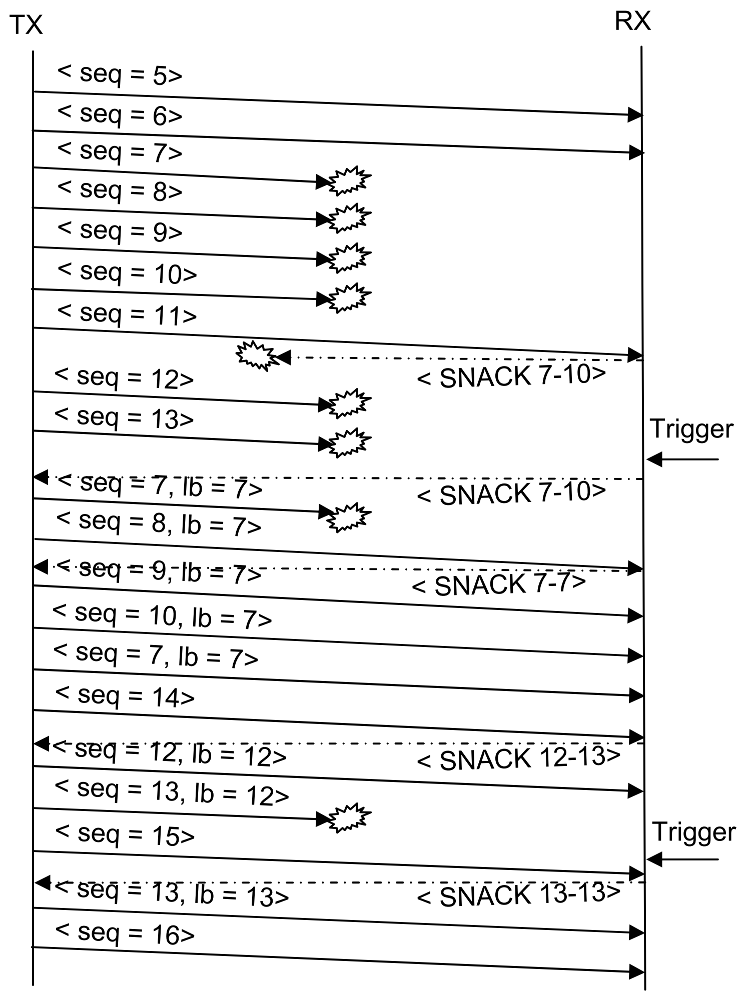
- **First Booster Element:**
  - Adds a sequence number to all passing packets of (UDP) real-time media streams.
  - Caches packets and the time of arrival of the packets.
- **Second Booster Element:**
  - Sends a SNACK to the first booster element when a gap in the sequence numbering is detected.
    - SNACK(first lost packet, last lost packet)
    - SNACK also includes the maximum delay packets may have at the first booster element to be considered for retransmission.
    - SNACK contains a key identifying the different packet streams
  - Calculates the maximum delay using measurements of the duration of a retransmission $T_{retrans}$ and, if possible, an estimation of the roundtrip time between the end hosts

- **ARQ Scheme:**
  - Loss of SNACKs:
    - Addition of a timer at the second booster element which triggers the retransmission of a SNACK
    - T_{expire} = upper bound($T_{retrans}$)
  - Introduction of a maximum retransmission threshold (vs. the infinite retransmissions of the basic schemes)
  - Discarding of duplicate packets at the second booster element
  - Los of retransmitted packets:
    - **Start of the sequence:**
      - Time constraint or packet loss?
      - Addition of a retransmission flag and a lower bound sequence number field to retransmitted packets.

ARQ Scheme:

- Los of retransmitted packets:
  - **End of the sequence:**
    - Reuse of the timer to trigger the retransmission of an updated SNACK
    - Initialized at arrival of the first packet of a retransmission
    - T_{expire} = upper bound($T_{remaining\ packets}$)
- Stream is not suspended
  - **Need for multiple timers (two)**

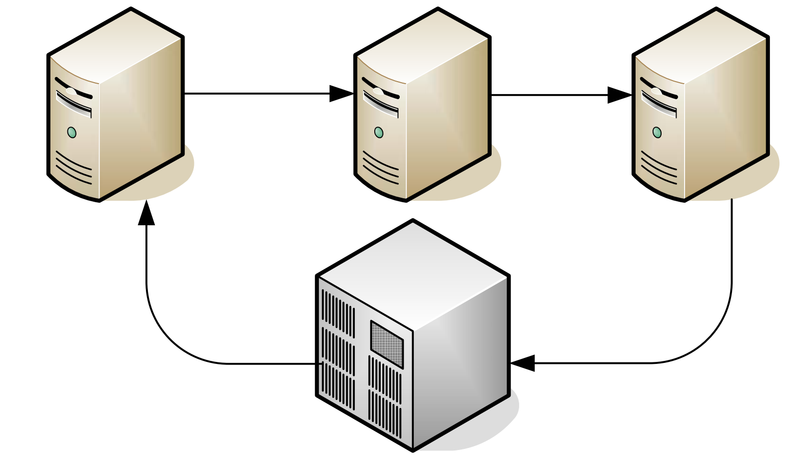
Real-Time Media Protocol Booster



Real-Time Media Protocol Booster

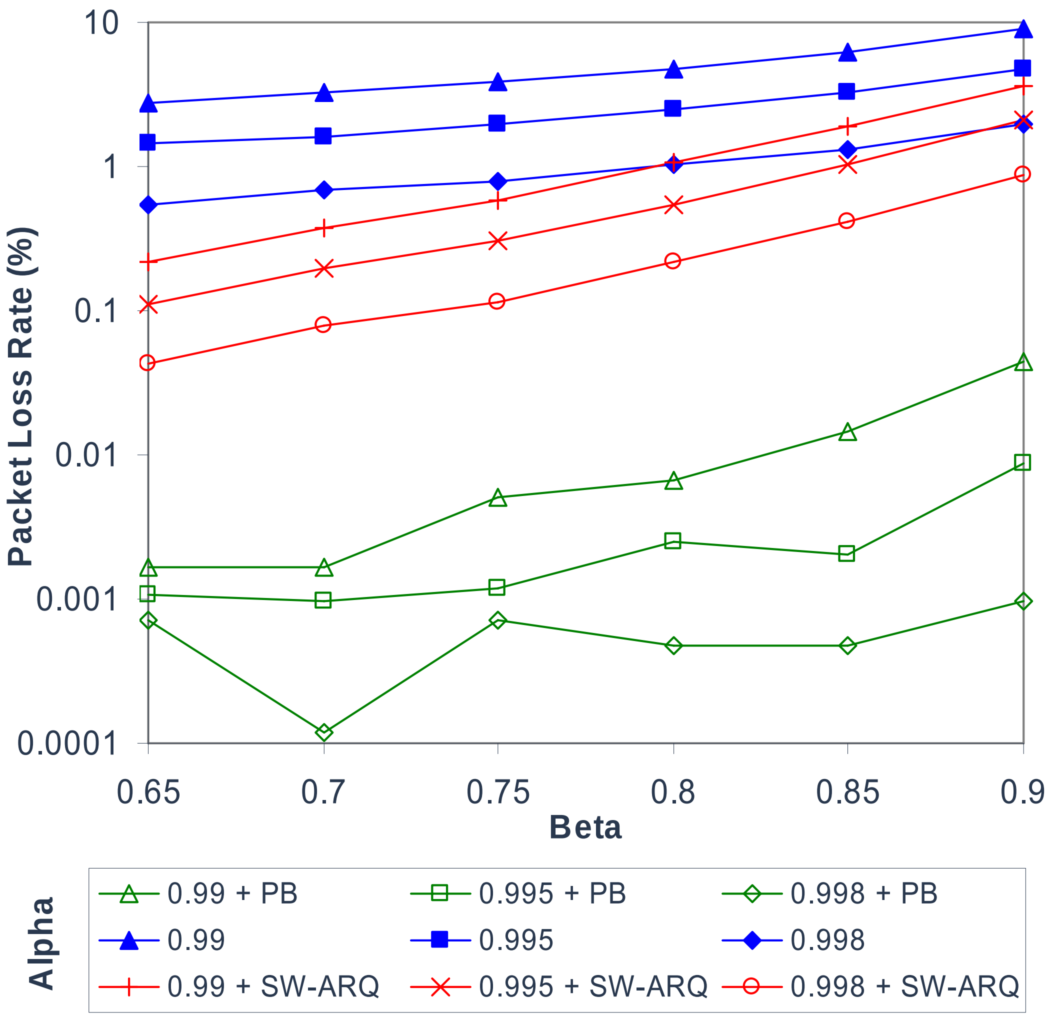
- The initial detection of packet loss occurs only at the arrival of an uncorrupted packet
  - **Speed up the detection by adding a timer**
  - **Initialized at the arrival of a packet**
  - **T_{expire} > upper bound(packet inter-arrival time)**
  - **SNACK(last received packet + 1, X)**
  - **Addition of a SNACK-type field to discern it from the previously discussed SNACK**
  - **Addition of a SNACK sequence number to detect duplicate SNACKs at the First Booster Element ⇨ discard**
- Bursty packet streams
  - **E.g. video streams**
  - **Adapt the previous timer:**
  - **T_{expire} > upper bound(time between bursts)**

Evaluation



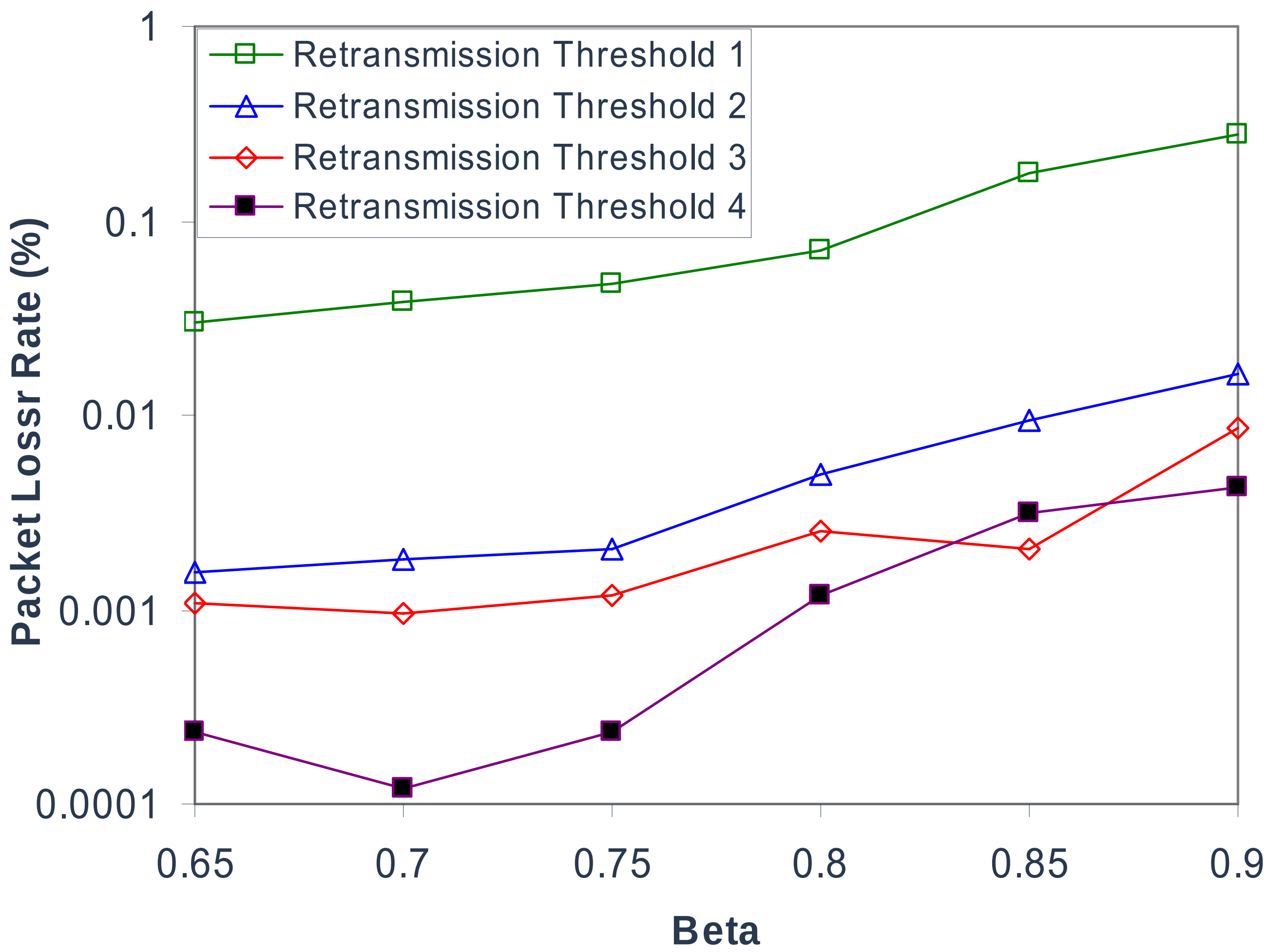
- Click Router Test Bed
- 5mbps UDP stream, bursts of 7 packet
- Play-out buffer size: 100ms
- Maximum retransmission threshold: 3
- Gilbert model: divide silent time into timeslots, size: upper bound(packet inter-arrival time)

Evaluation

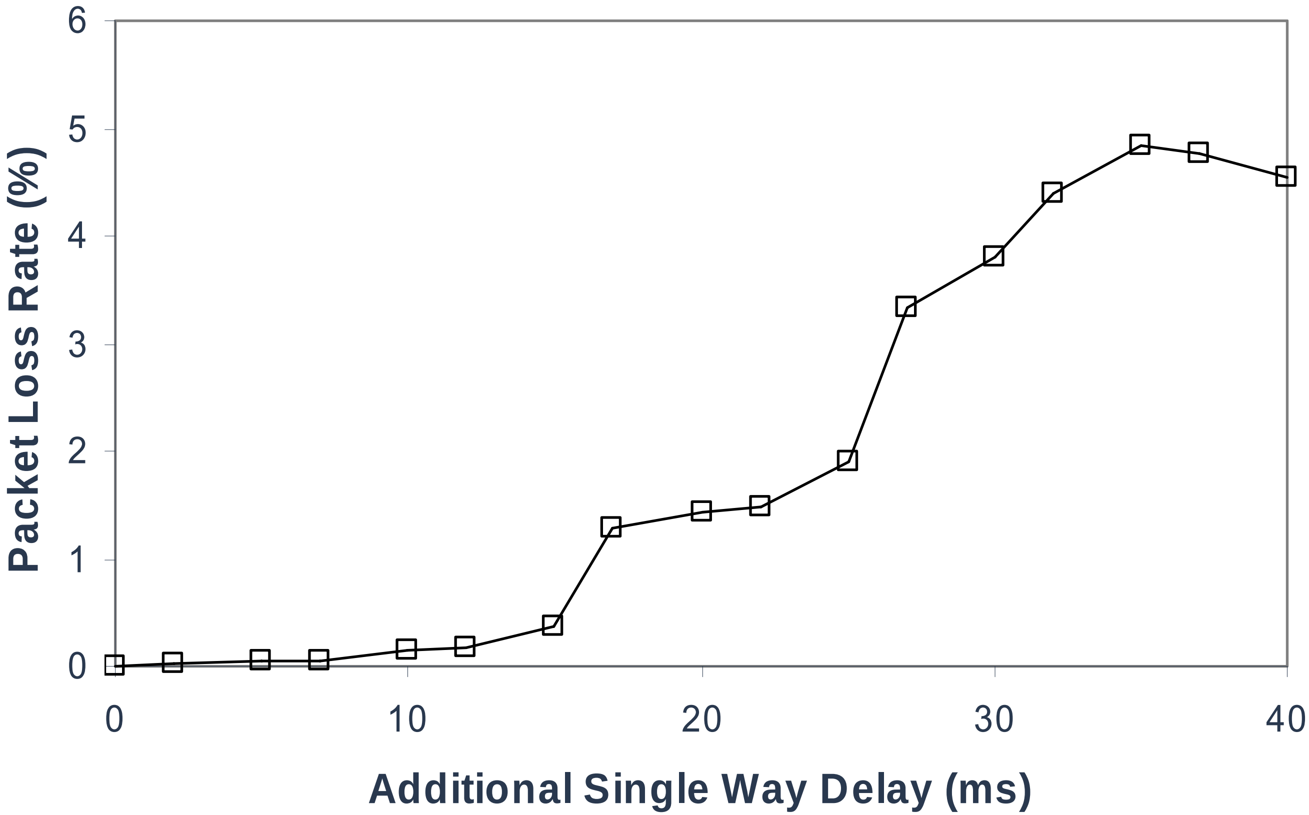


Packet Loss Rate (%)
10
1
0.1
0.01
0.001
0.0001
0.65
0.7
0.75
0.8
0.85
0.9
Beta
Alpha
0.99 + PB
0.995 + PB
0.998 + PB
0.99
0.995
0.998
0.99 + SW-ARQ
0.995 + SW-ARQ
0.998 + SW-ARQ

Evaluation

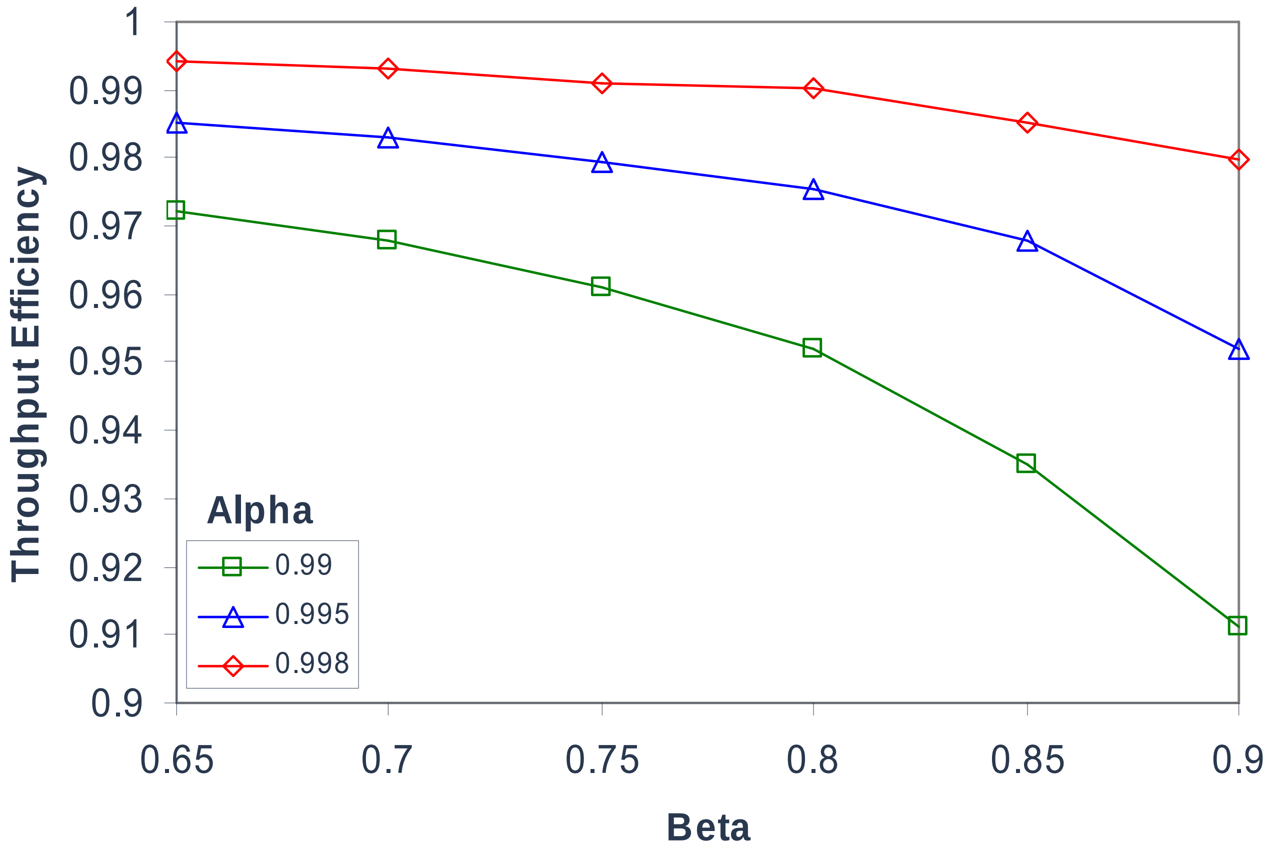

Retransmission Threshold 1
Retransmission Threshold 2
Retransmission Threshold 3
Retransmission Threshold 4
Packet Lossr Rate (%)
1
0.1
0.01
0.001
0.0001
0.65
0.7
0.75
0.8
0.85
0.9
Beta

Evaluation



Packet Loss Rate (%)
6
5
4
3
2
1
0
0
10
20
30
40
Additional Single Way Delay (ms)

Evaluation


1
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.92
0.91
0.9
Throughput Efficiency
Alpha
0.99
0.995
0.998
0.65
0.7
0.75
0.8
0.85
0.9
Beta

Conclusion and Questions