Analysis of Concurrent TCP and Streaming Traffic Over a Wireless Link

Tuomas Kulve

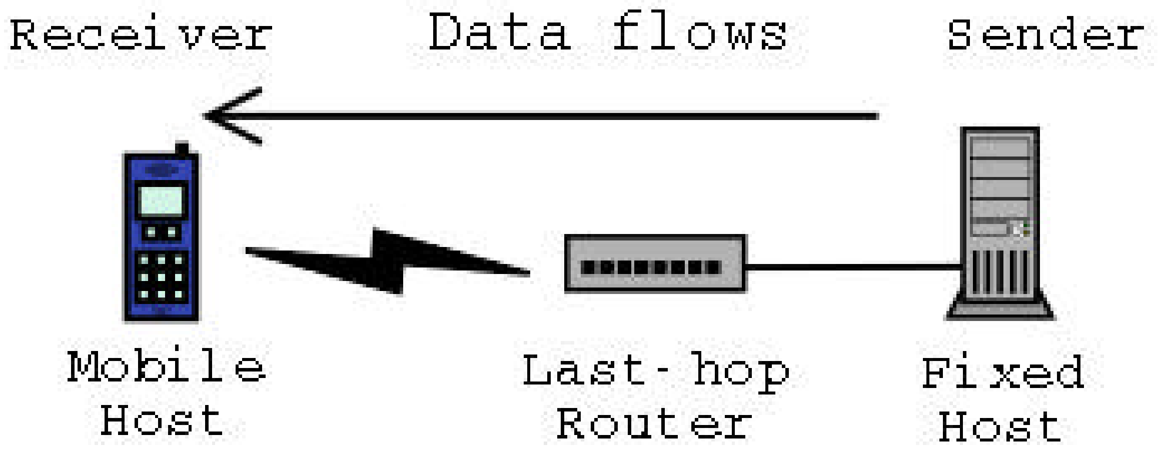
tuomas.kulve@cs.Helsinki.FI

Contents

- Target environment
- Link characteristics
- Workloads
- Baseline and Enhancements
- Summary of Results
- Conclusion

Target Environment

- Emulated wireless link from mobile host to last-hop router
- Real end hosts



Link Characteristics

- Symmetric 64kbit/s bandwidth
- Router queue 20 packets
- Link buffers roughly 2*BDP
- Two state error model
  - Good state: long (mean 15s), no errors
  - Bad state: 0.2-1.5s, 63% packet error rate
  - Link layer retransmissions
    - 700ms delay for each retransmission

Link Characteristics

- Optimal link
  - No errors
- One link layer retransmission
  - Drop packet, if packet loss after 1st retransmission
  - Results to delays and packet losses
- Three link layer retransmissions
  - Drop packet, if packet loss after 3rd retransmission
  - Results to delays
  - Persistent enough to avoid packet losses

Link Characteristics

- Six link layer retransmissions
  - Drop packet, if packet loss after 6th retransmission
  - Different bad state
    - Length: 0.5-4.0s
    - 95% packet error rate
  - Results to long delays
  - Persistent enough to avoid packet losses
  - Spurious RTOs

Workloads

- 2 TCP connections start at the same time
- 2 TCP connections start with 5s difference
- 2 TCP connections and 1UDP flow
  - UDP: CBR flow with 32kbit/s of payload stream
  - 512byte payload per UDP packet

Baseline

- Timestamps (with and without)
- Limited transmit
- SACK
- Initial Window was 2 segments
  - Except with CBI
- Delayed ACKs: 200ms
- No quick ACKs
- No ratehalving
- No CBI

TCP Enhancements

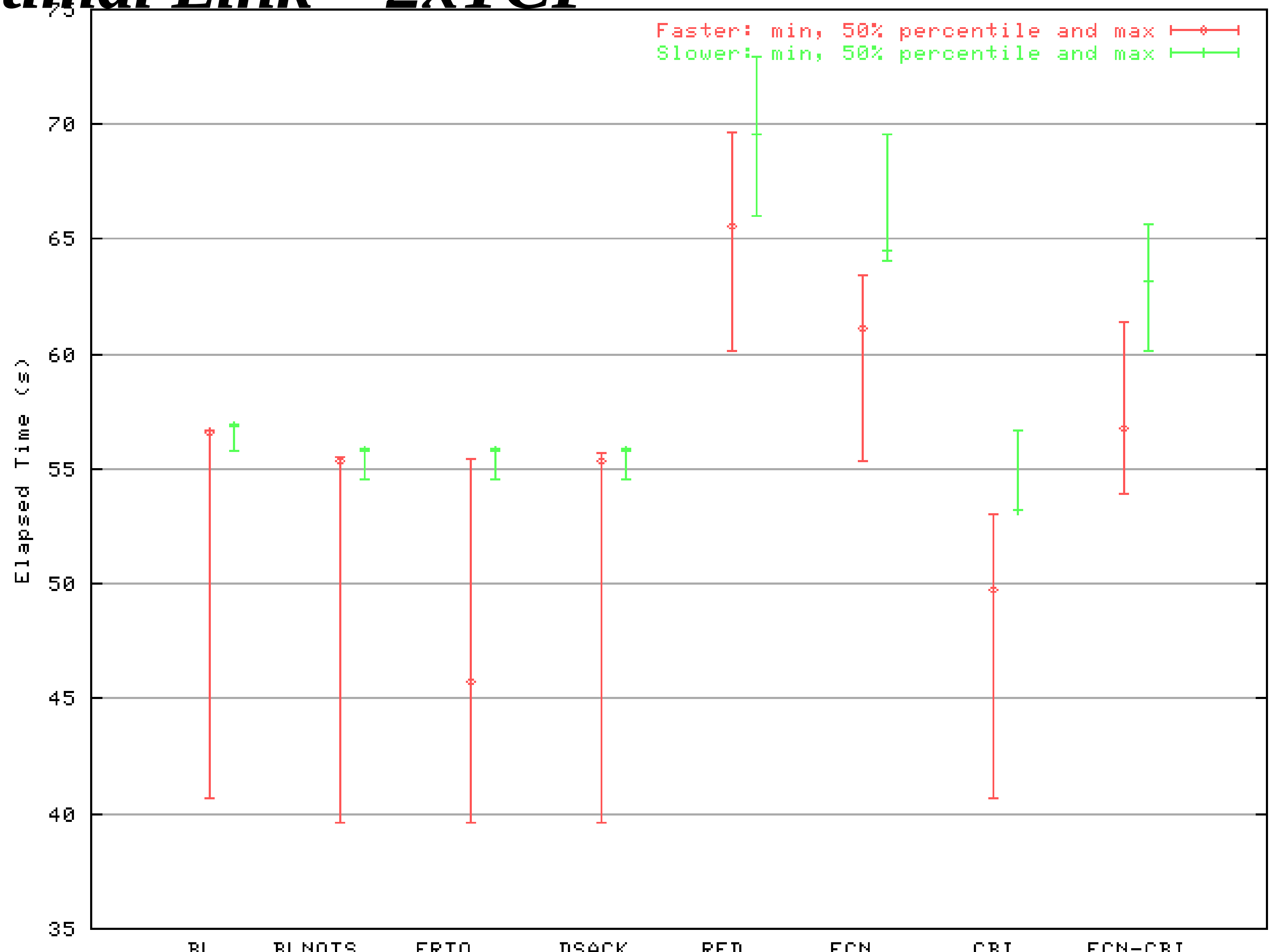
- Forward RTO-recovery (F-RTO)
  - If 2 ACKs after RTO advances window, then the RTO is assumed as spurious and the **CWND** is not set to one segment
- Duplicate-SACK (D-SACK)
  - Receiver explicitly informs about duplicate packets
  - If all retransmissions were unnecessary, undo

TCP Enhancements

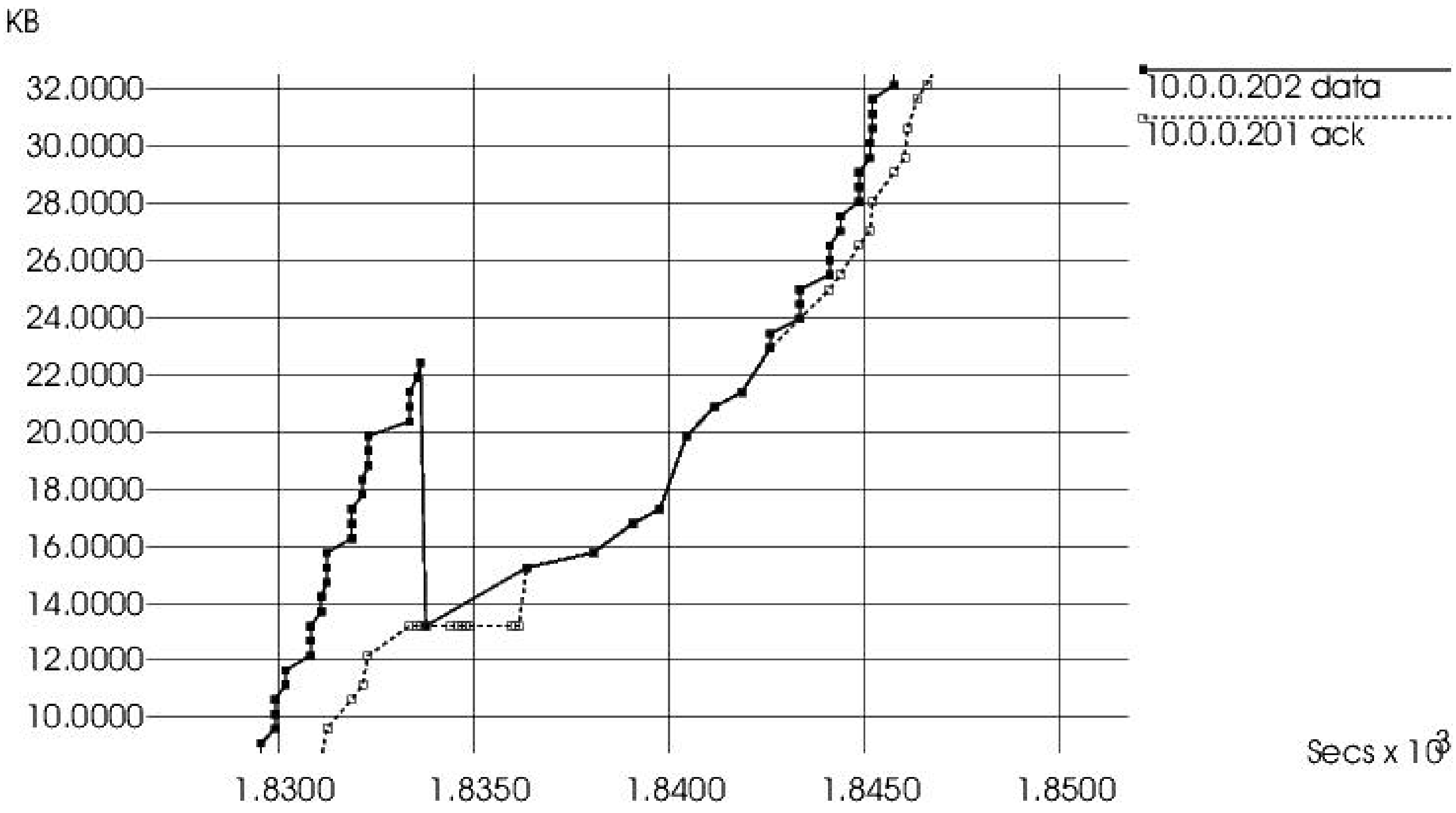
- With timestamps an Eifel-like mechanism is used
  - If first ACK is an acknowledgement to the original packet, undo
- Control Block Interdependence (CBI)
- Random Early Detection (RED)
- RED with Explicit Congestion Notification (ECN)
- Combination of CBI and ECN

Summary of Results

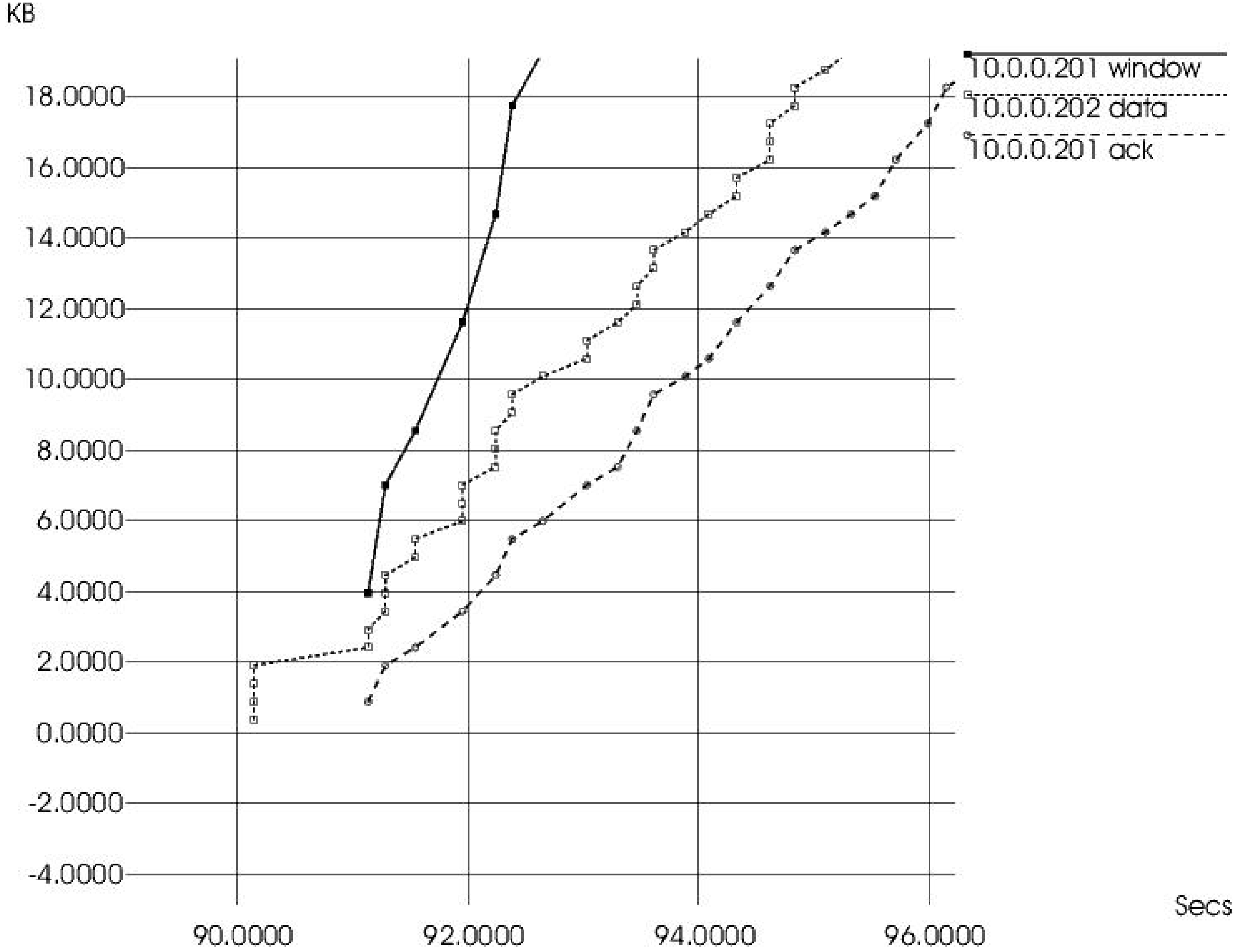
Optimal Link – 2xTCP



Slow Start Overshoot Recovery

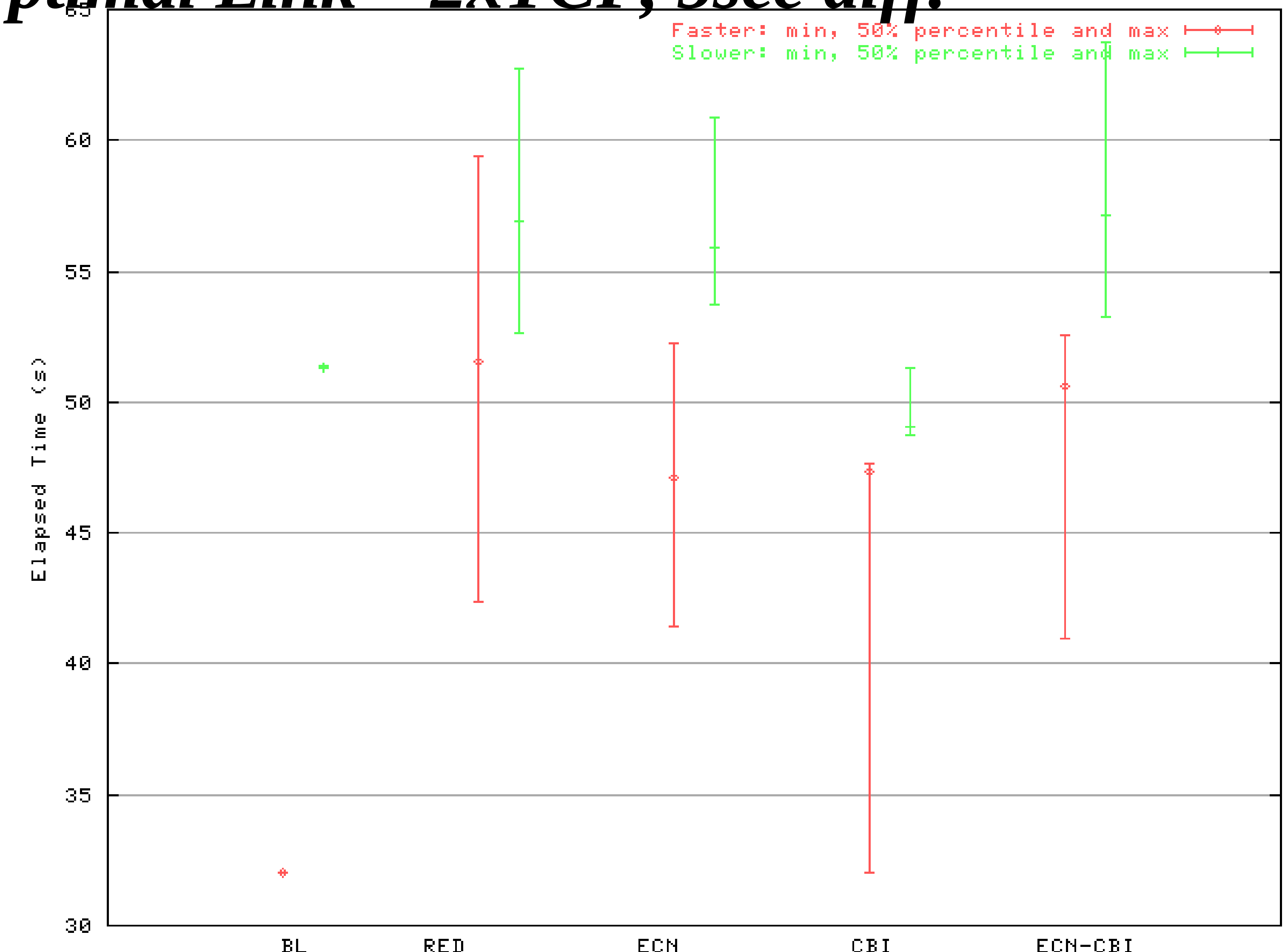


Slow Start With CBI



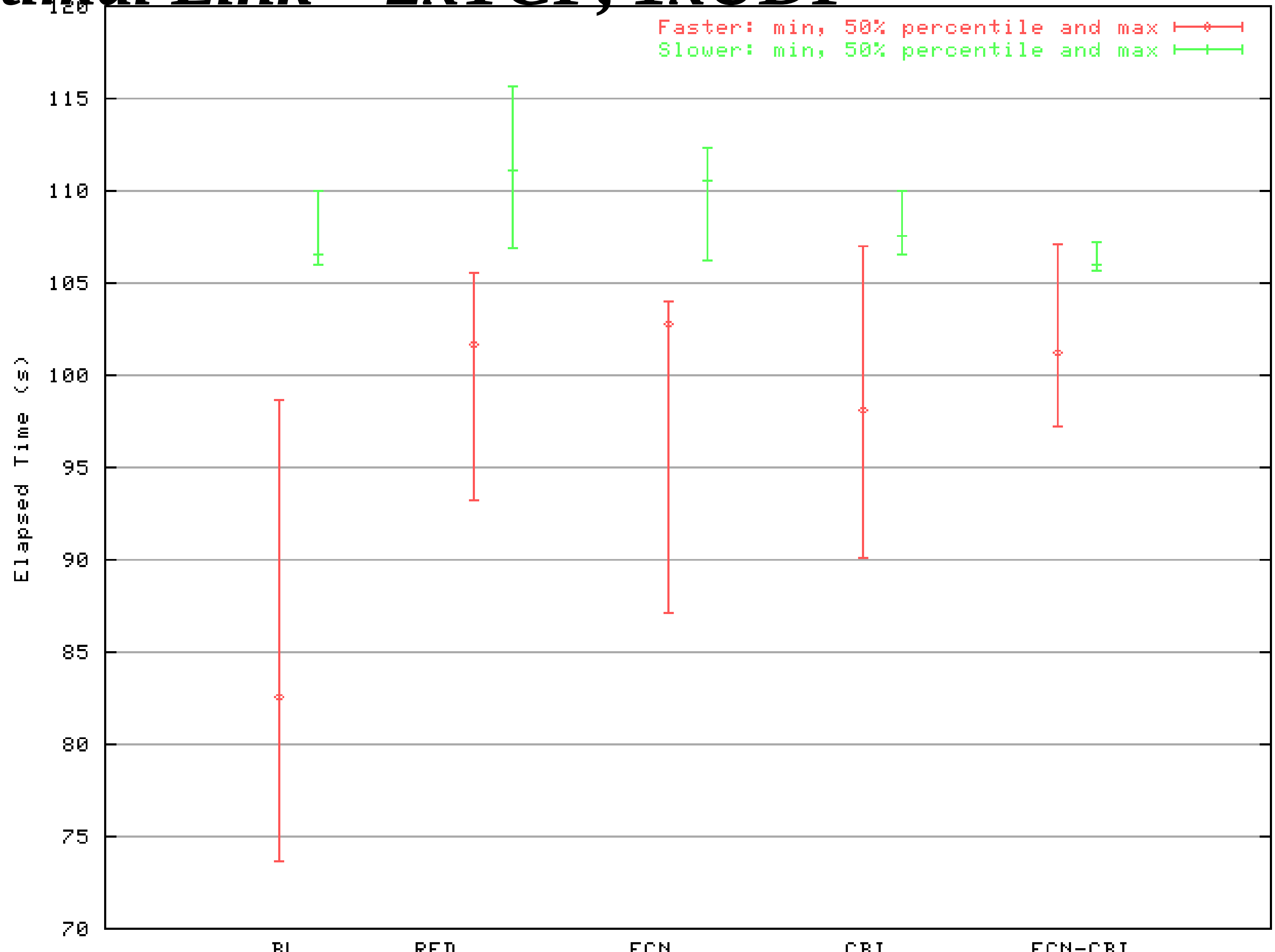
Summary of Results

Optimal Link – 2xTCP, 5sec diff.



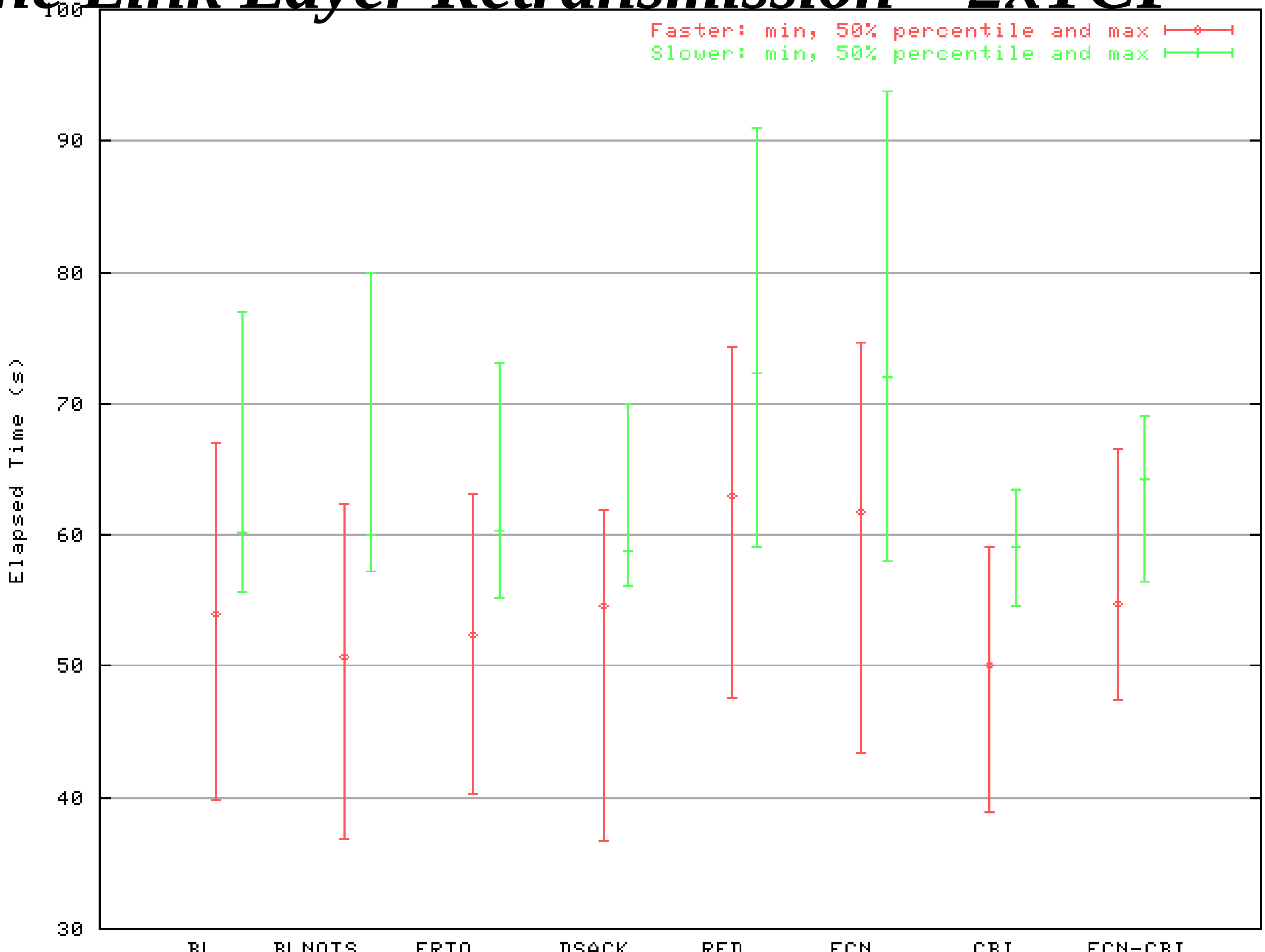
Summary of Results

Optimal Link – 2xTCP, 1xUDP



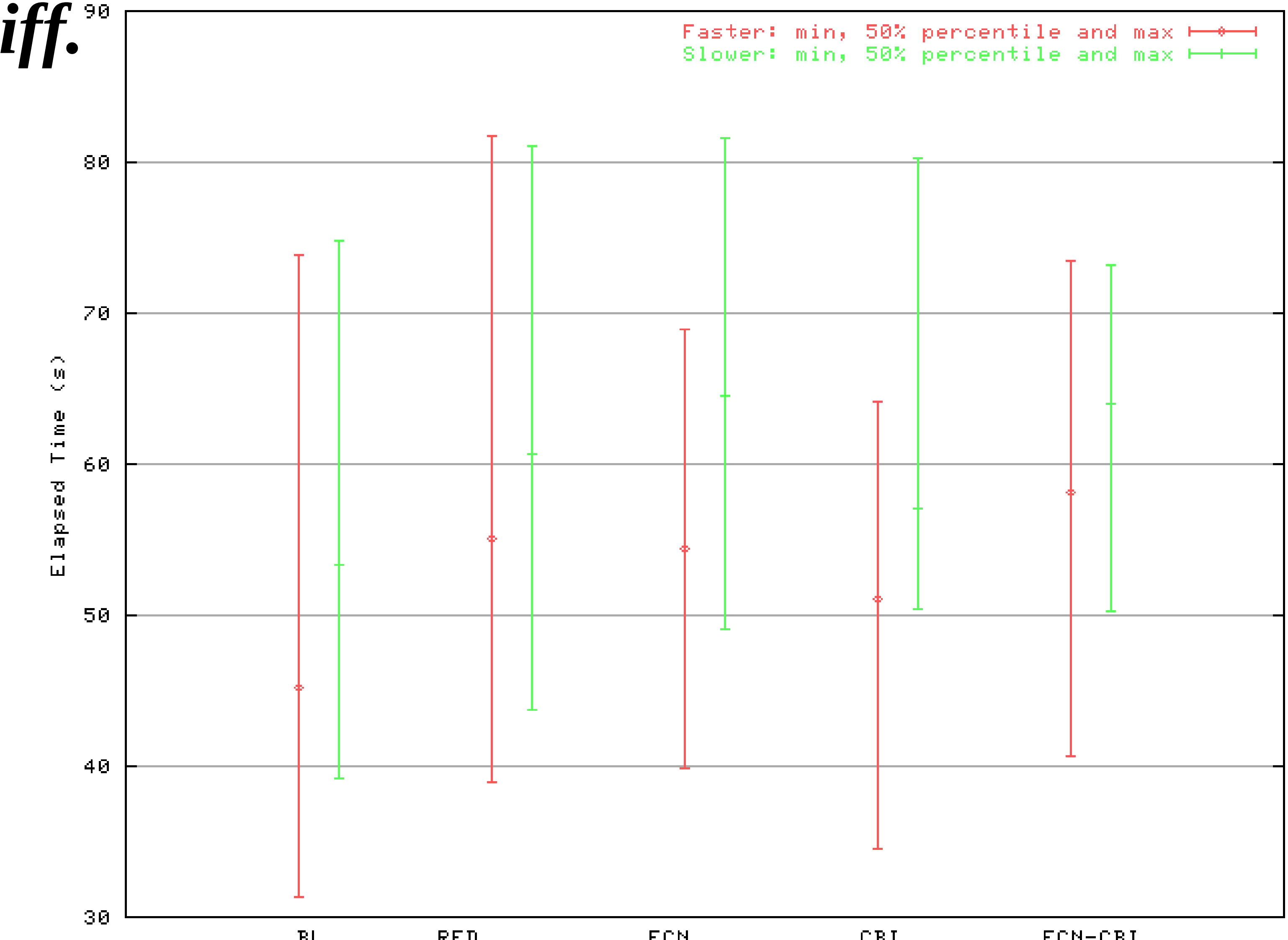
Summary of Results

One Link Layer Retransmission – 2xTCP



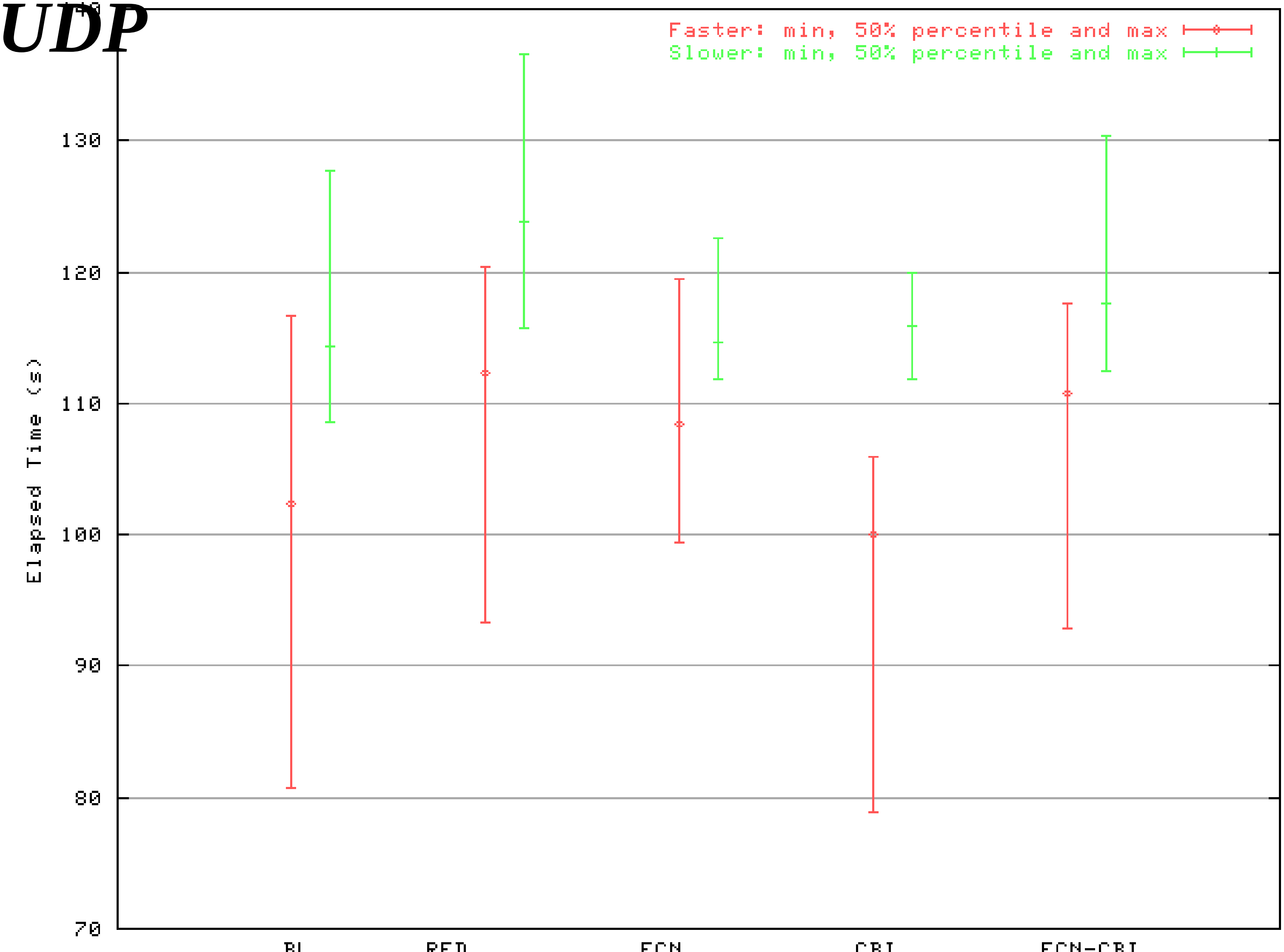
Summary of Results

One Link Layer Retransmission – 2xTCP, 5sec diff.



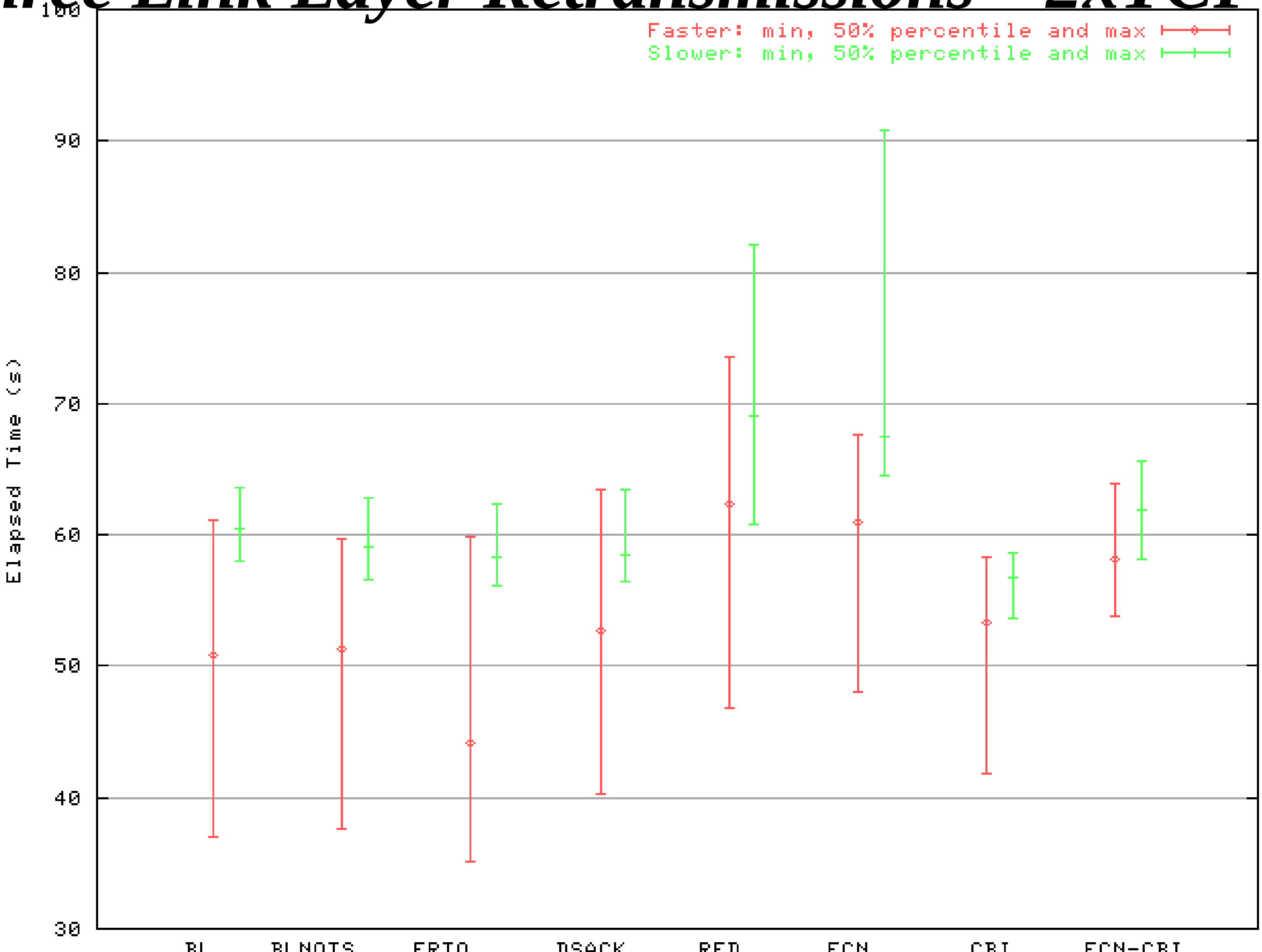
Summary of Results

One Link Layer Retransmission – 2xTCP, 1xUDP



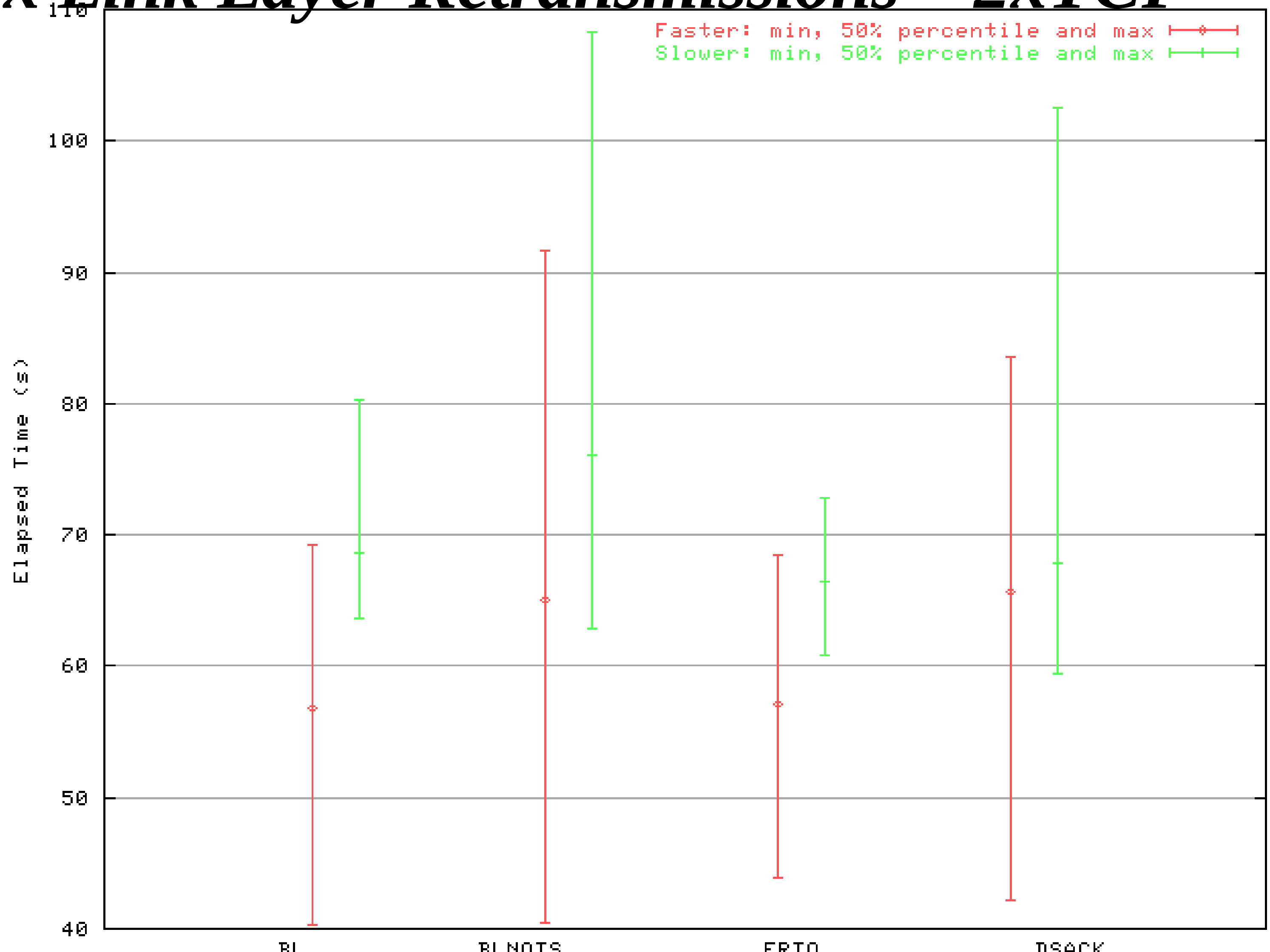
Summary of Results

Three Link Layer Retransmissions – 2xTCP



Summary of Results

Six Link Layer Retransmissions – 2xTCP



Conclusion

- Fairness usually poor with baseline TCP
  - Especially if connections did not start simultaneously
- CBI improves the fairness because of the absence of the slow start overshoot
- RED
  - Improves the fairness
  - Slows down the connections
- ECN introduces a slight gain compared to pure RED

Conclusion

- Baseline with timestamps and F-RTO recovered spurious RTOs efficiently
- Baseline without timestamps and D-SACK made unnecessary retransmissions and unnecessarily slowed down