Forward Error Control in Wireless Local Area Networks

ENSC835 Network Protocols & Performance

Final Project Presentations

Spring 2002

Ivy Chow (ischow@sfu.ca)

Horace Yat-Hong Chan (horace_chan@pmc-sierra.com)

Sameer Khushal (skhushal@sfu.ca)

Introduction

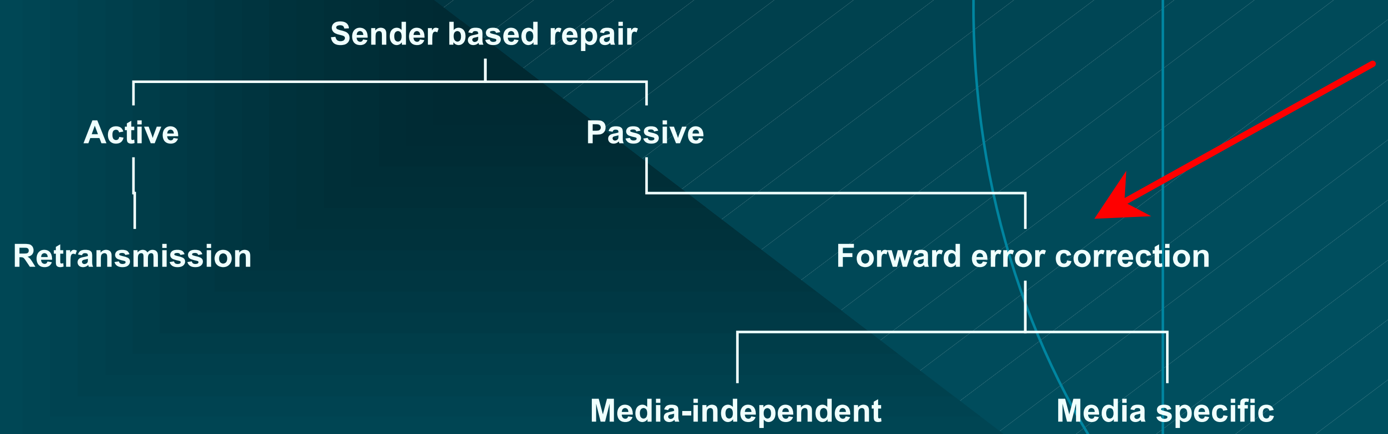
- Apply Forward Error Correction (FEC) to IEEE 802.11b standard
- Bit errors from channel noise protected by Cyclic Redundancy Check (CRC)
- End to end connection provided by TCP which needs very low error rates for efficiency
- 2 Approaches for FEC implementation in TCP wireless LANs
  - Reed Solomon Code
  - Turbo Code (Berrou)
- Use OPNET to show both implementations give less congestion in the network

Why is FEC Important for WLAN

- Packet loss in TCP causes window size decrease and retransmission of the lost packet
- The goal is to maximize the window size, for optimal utilization of bandwidth in channel
- In wired network, mostly traffic congestion lead to decrease in window size
- In a noisy wireless networks, window size is very small due to the high bit error rate of the channel
- FEC reduces retransmission by transmitting error correction code with packets

Introduction to FEC

- FEC Overview
  - Repair losses of data during transmission by adding repair data
  - Computation to recover the error is more time efficient than retransmission



Introduction to FEC (cont.)

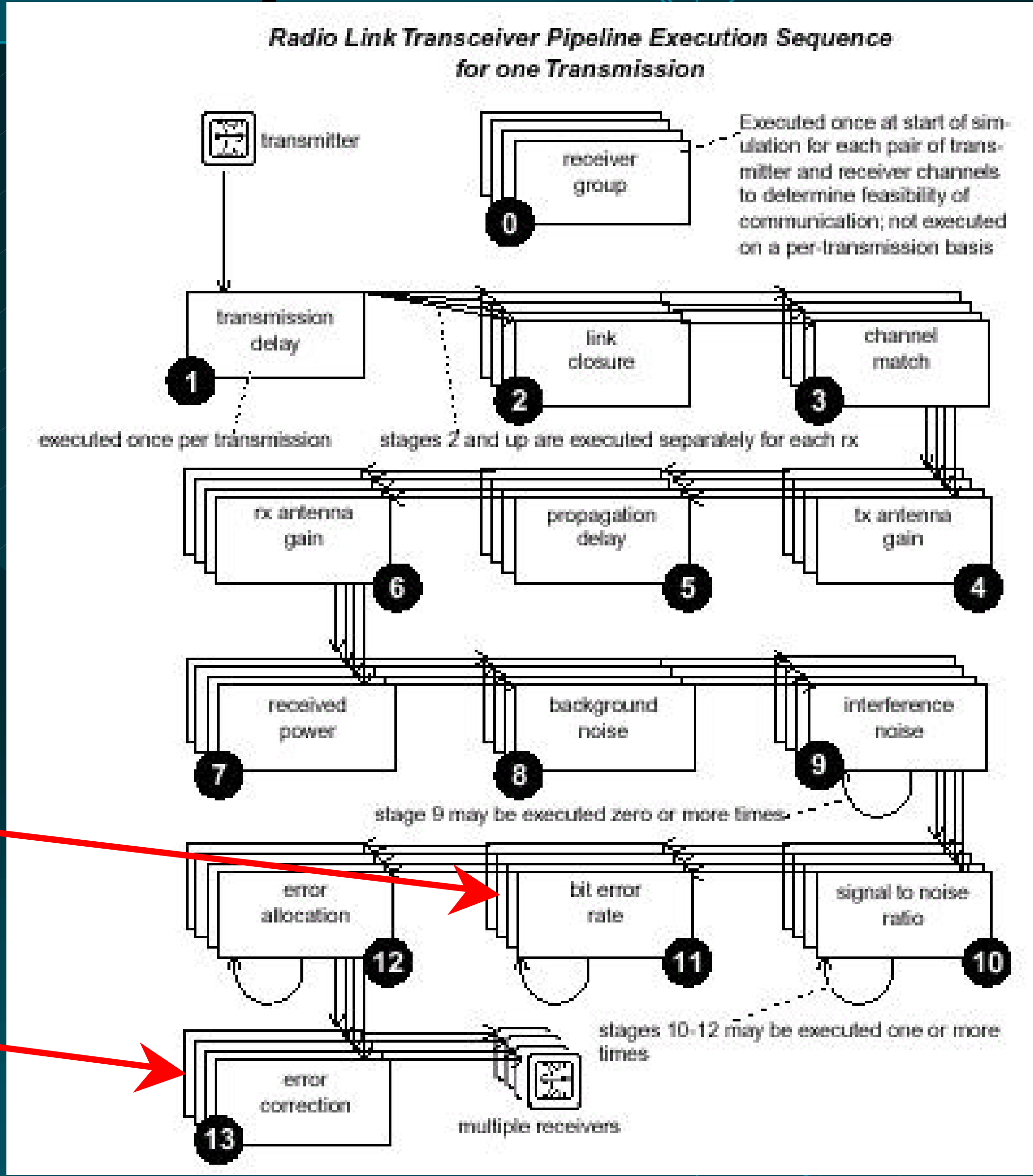
- **Media Indpendent, Reed Solomon Code**
  - Use block or algebraic codes to produce additional packets for transmission
  - Takes k data packets and generates n-k check packets to transmit n packets over network
  - Additional process delay & increased bandwidth
- **Media Dependent, Turbo Code**
  - Concatenated coding and iterative decoding
  - Achieves close to Shannon capacity of channel
  - Very complex to implement: need memory management scheme for decoder

Implementation In Opnet

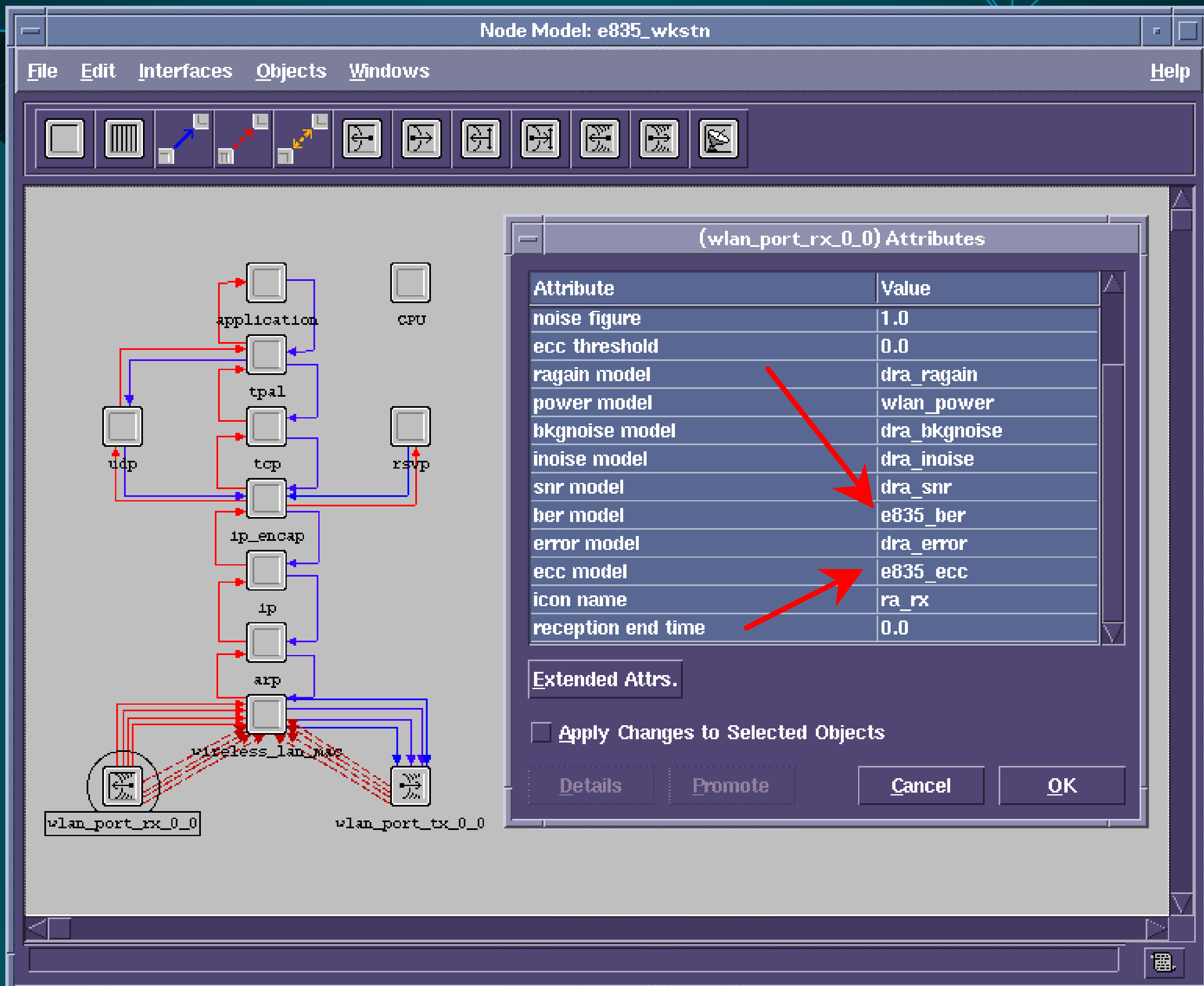
- Trace driven simulation
- Implemented in the Radio Link Transceiver Pipeline

Bit Error Rate Generation

Error Correction Models

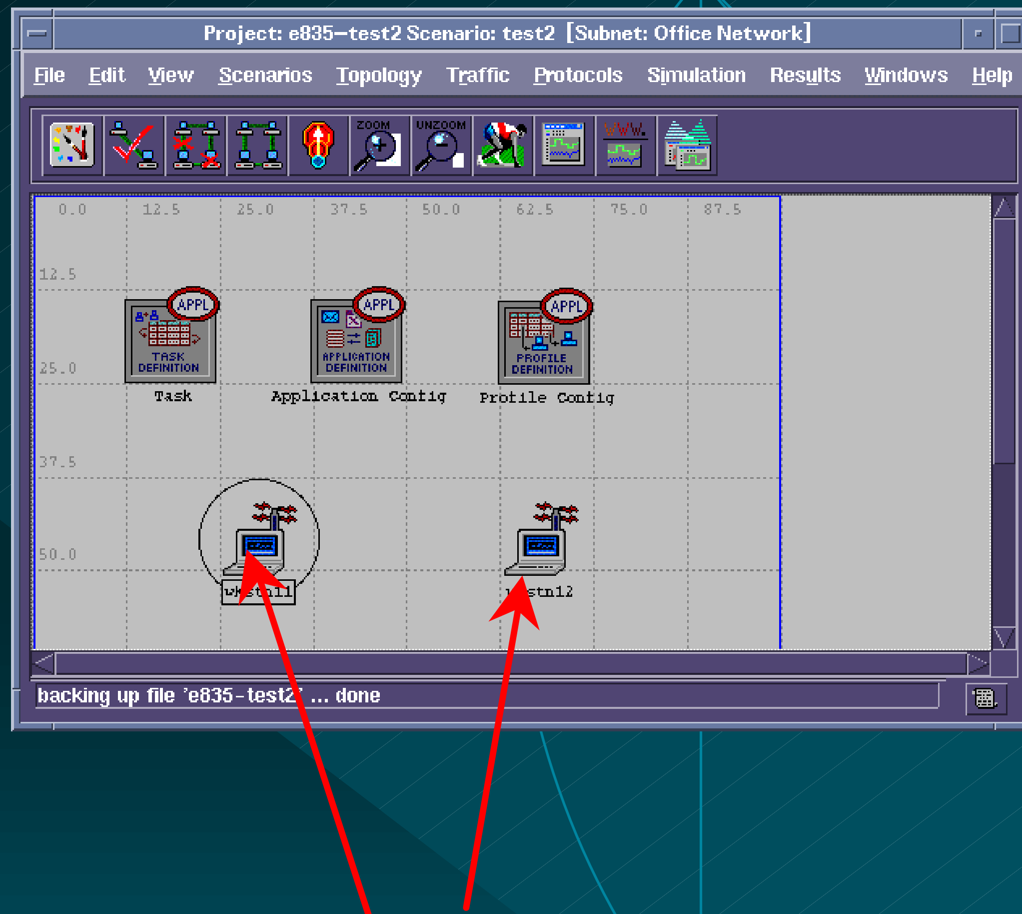


Node Implementation



Simple Topology

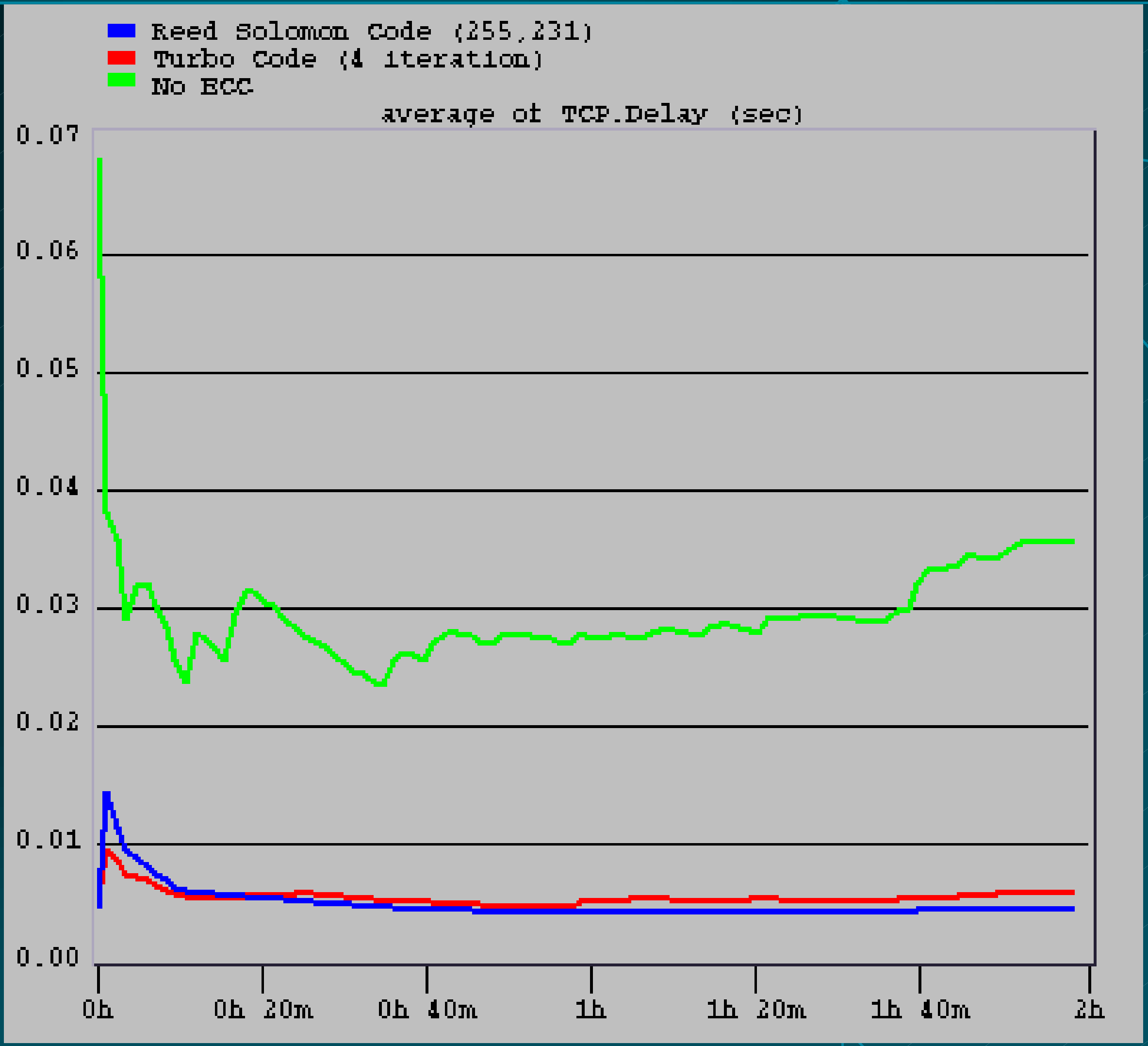
- TCP Reno (2272 Segment Size)
- No retransmission in WLAN layer
- Bit Error Rate = 10e-6
- Reed Solomon Code (255,231)
- Turbo Code (4 iterations)



802.11 Node

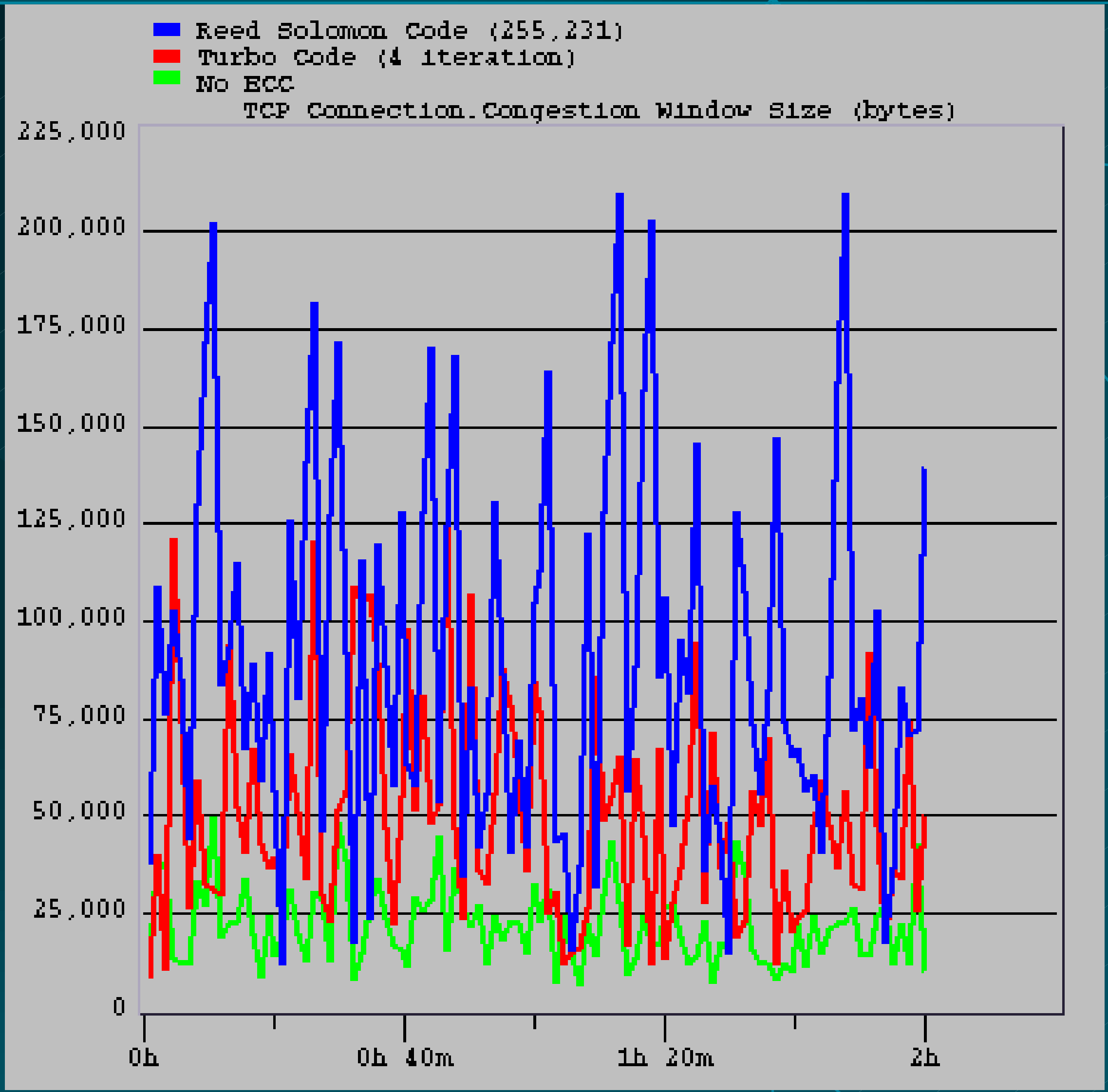
Simulation Results

- TCP Delay
- RS Code
- Turbo Code
- No FEC

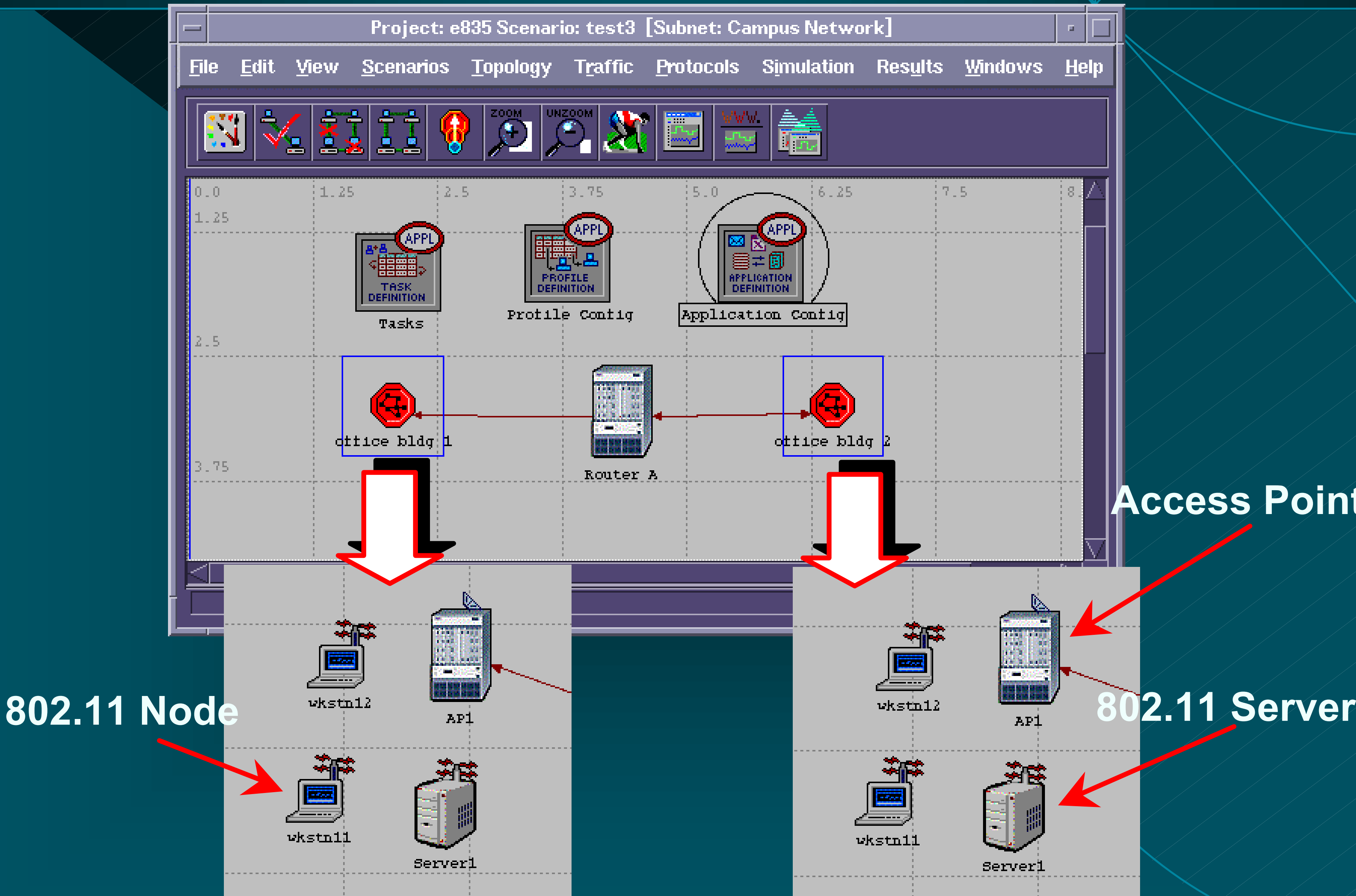


Simulation Results

- TCP Window Size
- RS Code
- Turbo Code
- No FEC

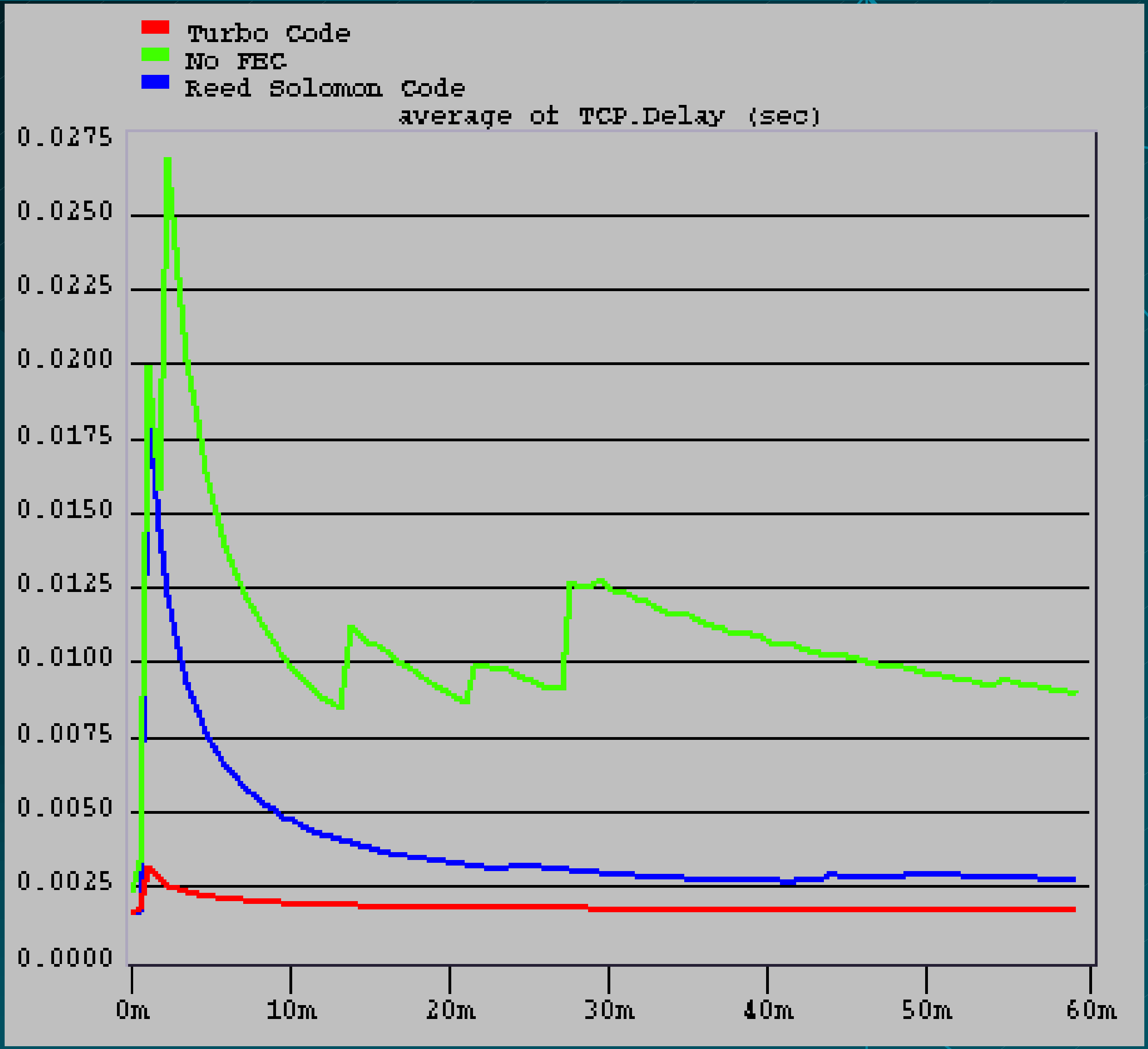


Complex Topology



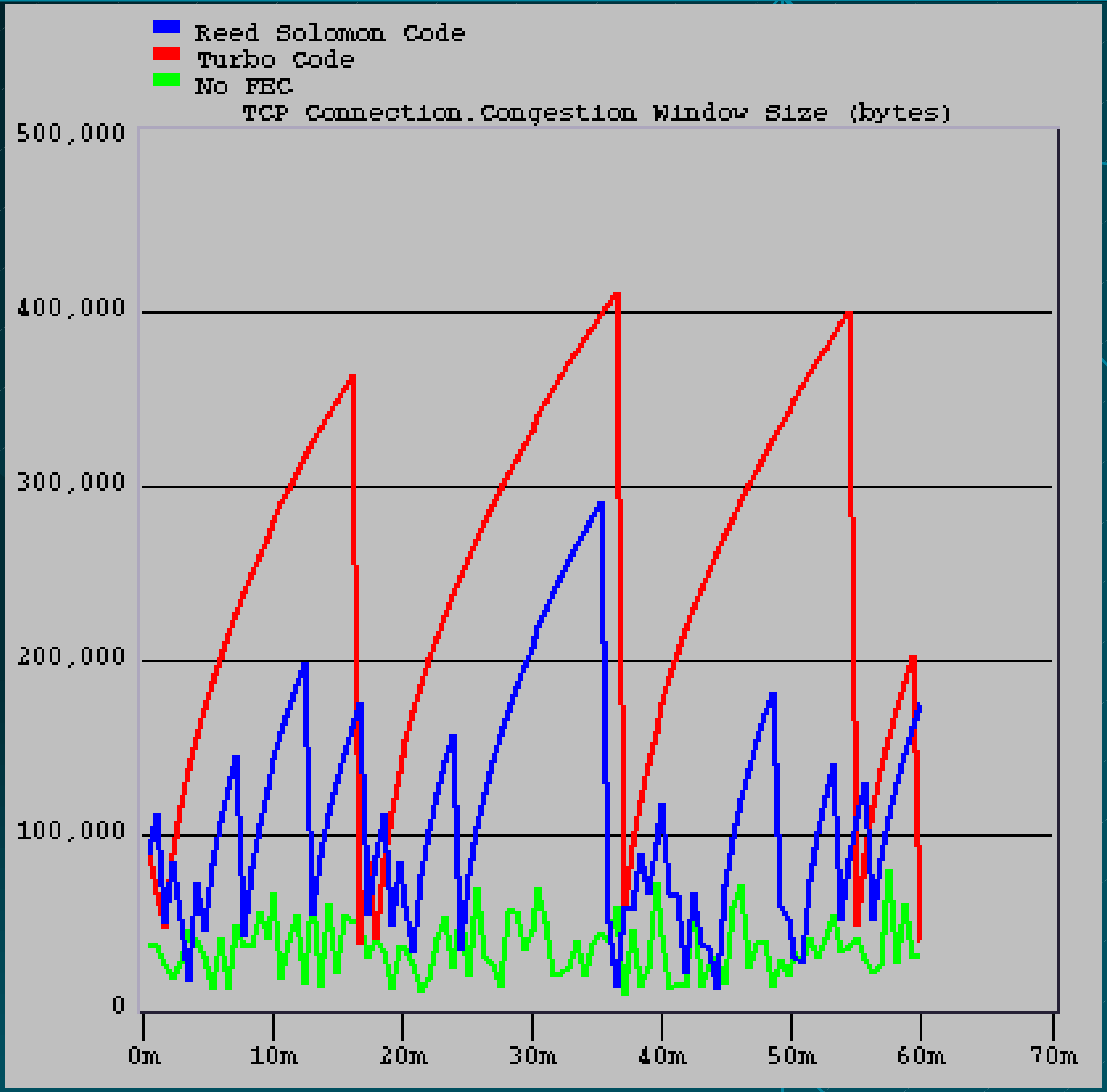
Simulation Results

- TCP Delay
- RS Code
- Turbo Code
- No FEC



Simulation Results

- TCP Window Size
- RS Code
- Turbo Code
- No FEC



Conclusion

- Implementation of FEC in 802.11 WLAN improved TCP performance
- Both FEC codes: Reed Solomon & Turbo Code show reduce the TCP delay and increase the window size
- Reed Solomon is easier to implement and has determined processing time
- Turbo Code can increase the channel capacity to the Shannon limit but it is more difficult to implement

References

- [1] Chris Heegard et al, High-Performance Wireless Ethernet, IEEE Communications Magazine, pp. 64-73, November 2001.
- [2] Gilbert Held, The ABCs of IEEE 802.11, IT Professionals, pp. 49-52, November 2001
- [3] A. Burr, Turbo-codes: the ultimate error control codes?, Electronics & Communication Engineering Journal, pp.155-165, August 2001
- [4] Ian F. Akyidiz et al, An Adaptive FEC Scheme for Data Traffic in Wireless ATM Networks, IEEE/ACM Transcations on Networking, pp.419-425, August 2001
- [5] Brett Schein and Steven Bernstein, A Forward Error Control Scheme for GBS and BADD, MILCOM 97 Proceedings, pp.710-715, Volume 2, 1997

Questions & Comments

- Thank you!