I. MOTIVATION

- End-to-End Transport and the Evolution of Networks
  The growth of communication networks and the evolution of the Internet has resulted in various transport layer protocols attempting to suit different types of network environments. Communication networks have seen a rapid growth in the number of users, the capacities needed by each user, and the heterogeneity of the network. In this regard, we have observed a shift from wired access to wireless networks. This has resulted in a frequent re-examination of the algorithms that ensure the end-to-end transport of data over the transport layer. The de facto standard of the Internet transmission control protocol (TCP) – has evolved over time giving rise to a number of variants in the process. In this study, we focus on a comparison of the TCP variants.

- Corruption based losses
  Wireless networks are generally known to possess a high bit error rate due to the sporadic nature of the wireless channel. TCP interprets this loss to occur due to congestion and thus lessens its sending rate conservatively when there is no such need. Thus this contributes to a degrade in performance in such high error wireless environments.

- Congestion based losses
  Overflowing a link results in dropped packets due to buffer overflows within the intermediate systems such as routers. This invariably occurs when multiple flows share a link thereby causing the shared buffer to reach its threshold and cause packet drops.

- Variants of TCP
  To cover whole space of variant evolution and significance, we chose Tahoe, Reno, NewReno, SACK, Westwood, and Westwood+. Westwood caters to wireless networks while Westwood+ allows for congestion control increasing fairness amongst flows. SACK implements a conservative loss recovery algorithm using the previously proposed SACK option. NewReno is the de facto standard of the Internet. Reno was a dominant protocol in the 90s and Tahoe, the earliest congestion control TCP variant serves as a baseline for our comparison.

II. RESEARCH GOALS

- Develop a framework to compare different transport protocols
- Understand the behavior of different congestion control mechanisms
- Understand how these protocols fit in a particular network scenario
- Study the performance of different protocols under lossy channels and congestion based scenarios
- Test our new protocols and validate their performance against existing protocols
- Develop various transport protocol modules for ns-3
- Develop a number of protocols as benchmarks for comparing with our Res-TP

III. TCP VARIANTS

- TCP Tahoe
  TCP Tahoe is the first variant to adopt congestion control algorithm that includes three phases. The slow-start phase allows for an exponential increase in the congestion window until a threshold is reached upon which the congestion avoidance phase starts thereby lessening the rate of increase of the window. The fast retransmit is triggered upon a triple DUPACK event allowing for an earlier retransmission of missing data when compared to the traditional TCP.

- TCP Reno
  TCP Reno is Tahoe’s successor that introduces the fast recovery mechanism to take advantage of available bandwidth indicated by the arrival of a DUPACK. After the fast retransmit, Reno enters the fast recovery and transmits new data segments for every received DUPACK until the arrival of a new ACK causing Reno to transfer back to the congestion avoidance phase.

- TCP NewReno
  TCP NewReno contains a slight modification to Reno’s fast recovery algorithm by staying in the phase until the reception of a full ACK ensuring the recovery of all lost segments.

- TCP SACK
  TCP SACK implements a conservative loss recovery algorithm to improve TCP performance, especially in facing multiple losses. By using the information in SACK option, TCP SACK can make more accurate retransmission and congestion control decisions.

- TCP Westwood
  To enhance TCP performance in wireless networks, TCP Westwood modifies Reno’s sender side to adjust the sending rate based on bandwidth estimation when a loss occurs. Westwood estimates the bandwidth by counting the number of received ACKs between two consecutive ACK receptions.

- TCP Westwood+
  A major drawback of Westwood is the overestimation of bandwidth in the presence of congestion thus accelerating congestion collapse and in the process being unfair to other flows. Westwood+ is a slight enhancement to the Westwood protocol to tackle the overestimation of bandwidth in the presence of congestion. Westwood+ estimates the bandwidth for every RTT period thus allowing for a higher sampling period for estimation of bandwidth.

IV. SIMULATION MODEL

- The simulation model consists of 2 scenarios; corruption and congestion based losses.
- The bulk transfer application is used to transfer data between communicating entities.
- The results are averaged over 20 simulation runs of 5000 second each.
- Throughput is used as the performance metric for this work.

V. RESULTS and ANALYSIS

- Varying error rate levels

  As the error rate increases, throughput decreases due to multiple retransmissions. TCP Westwood performs best followed by Westwood+.

- Varying bottleneck bandwidth levels

  Throughput increases as bottleneck bandwidth increases. SACK outperforms both Westwood and Westwood+ starting at bandwidth of 3 Mb/s.

- Estimated bandwidth in presence of ACK compression

  Westwood shows an overestimation of bandwidth due to arrival of multiple closely spaced ACKs. On the other hand, Westwood+ accurately estimates the bottleneck bandwidth.

VI. FUTURE WORK

- Usage of DCE framework to allow use of real world protocol implementations
- Increase protocol space by adding BIG, CUBIC, CTCP, SCPS-TP, AeroTP, and Res-TP
- Compare performance in congested networks with multiple flows
- Develop a framework choosing an appropriate protocol for given network scenario

Acknowledgements: This project is supported in part by the National Science Foundation FIND and GENI programs, and the EU FIRE Programme

May 2013