Bangladesh University of Business & Technology (BUBT)

Assignment for Mid-term Examination

Course Code **:** CSE 319
Course Title **:** Computer Network
Assignment no : 01

Submitted By
Name :Md. Mehedi Hasan
ID : 21225103334
Intake : 49
Section : 08
Program :B.Sc. in CSE

Submitted To
Dr. Khandoker Nadim Parvez
Associate Professor

Department of CSE
BUBT

Date of Submission: 2^{nd} November ,2024

……………………………….
Signature of Teacher

Wireless TCP Congestion Control: Approaches, and Future Directions

Definition of TCP :

Transmission Control Protocol (TCP) is a fundamental communication protocol in the Internet protocol suite that ensures reliable, ordered, and error-checked data transmission between devices on a network. TCP provides a connection-oriented service, meaning it establishes and maintains a stable connection between the sender and receiver until all data has been transmitted. By breaking data into packets, sending each with acknowledgment, and retransmitting any lost packets, TCP ensures that data is delivered accurately and in the correct sequence. This makes TCP essential for applications requiring reliable data delivery, like web browsing, file transfers, and email.

Definition of Congestion Control: Congestion Control is crucial in TCP to prevent network congestion, which can occur when too much data is transmitted too quickly, overwhelming network resources. Without congestion control, an increase in data flow could lead to packet loss, delays, and, in severe cases, a complete breakdown of network communication, known as congestion collapse. TCP uses congestion control mechanisms to adjust the rate at which data is sent based on feedback from the network, such as acknowledgments and packet loss signals. By dynamically managing the sender's transmission rate, congestion control ensures that the network operates efficiently, with a stable flow of data, minimal delays, and optimal use of available bandwidth. This balancing act is essential for maintaining high performance and reliability in network communication.

Congestion Policy in TCP:

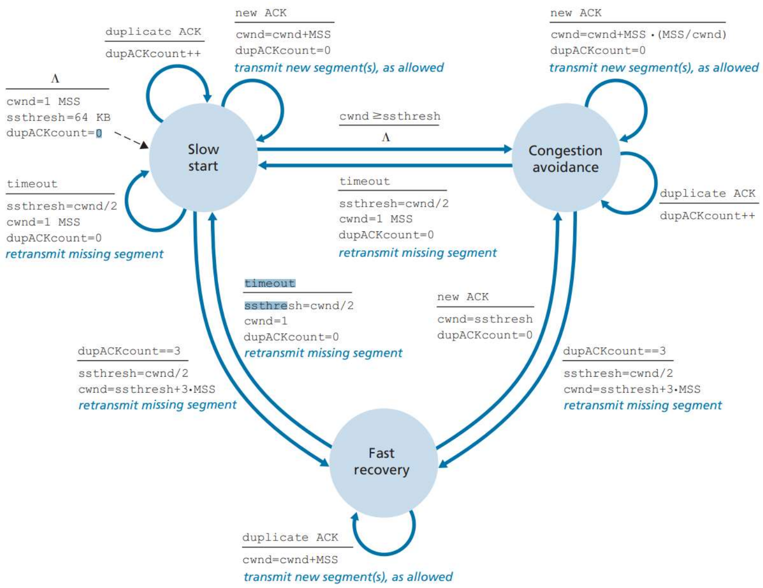

Figure 3.51 ♦ FSM description of TCP congestion control

TCP Congestion Control FSM:

1. **Slow Start (SS)**
    - **Purpose**: Gradually increases the data transmission rate to discover the available network capacity.
    - **Actions**:
        - Sets the congestion window (cwnd) to 1 MSS (Maximum Segment Size).
        - Increases cwnd exponentially (doubles every round-trip time, or RTT) until it reaches a threshold, known as ssthresh, or packet loss is detected.
    - **Transitions**:
        - To Congestion Avoidance (CA): If cwnd reaches ssthresh.
        - To Timeout: If no acknowledgment (ACK) is received, triggering a timeout.
        - To Fast Recovery (FR): If three duplicate ACKs are received.
2. **Congestion Avoidance (CA)**
    - **Purpose**: Probes the network for available bandwidth cautiously after the slow start phase.
    - **Actions**:
        - Increases cwnd linearly by 1 MSS per RTT to avoid overwhelming the network.
    - **Transitions**:
        - To Timeout: If a timeout occurs due to a lost packet.
        - To Fast Recovery (FR): If three duplicate ACKs are received, indicating mild congestion.
3. **Fast Recovery (FR)**
    - **Purpose**: Handles mild congestion by quickly retransmitting lost packets without reverting to slow start.
    - **Actions**:
        - Retransmits the missing segment (fast retransmit).
        - Sets ssthresh to half the current cwnd.
        - Reduces cwnd by half and adds 3 MSS to maintain a steady flow.
        - Uses ACK-clocking to stabilize the flow and increase cwnd linearly.
    - **Transitions**:
        - To Congestion Avoidance (CA): After receiving an ACK for all outstanding data.
        - To Timeout: If an ACK is not received and a timeout occurs.
4. **Timeout (TO)**

- **Purpose**: Responds to severe congestion, indicated by a lost packet with no ACK received within the retransmission timeout.
- **Actions**:
  - Sets cwnd to 1 MSS and ssthresh to half of cwnd at the time of timeout.
- **Transition**:
  - To Slow Start (SS): TCP resets and begins the slow start process again to increase its transmission rate gradually.

TCP Variants for Wireless Networks:

Wireless networks present unique challenges for the Transmission Control Protocol (TCP) due to issues like variable latency, packet loss, and changes in bandwidth. Various TCP variants have been developed to address these challenges, improving performance in wireless environments. Here's an overview of some significant TCP variants and modifications designed for wireless networks:

1.TCP Tahoe and TCP Reno : TCP Tahoe introduced slow start and congestion avoidance, where the congestion window (cwnd) grows exponentially until a packet loss occurs, leading to a linear decrease in cwnd. TCP Reno improved upon Tahoe by adding fast retransmit and fast recovery mechanisms.Both Tahoe and Reno do not differentiate between losses caused by network congestion and those caused by wireless transmission errors, leading to unnecessary reductions in the congestion window, which can degrade performance in wireless environments.

- **Case Study of TCP Tahoe :** In a simulation of a wireless LAN, TCP Tahoe's performance dropped significantly due to packet loss from wireless interference, resulting in lower throughput compared to newer variants.
- **Case Study of TCP Reno :** In a test involving a mobile ad-hoc network, TCP Reno demonstrated improved recovery times compared to Tahoe but still suffered from throughput drops in high-loss environments.

2. TCP NewReno: TCP NewReno enhances TCP Reno's ability to recover from multiple packet losses within a single transmission window. It provides better handling of partial acknowledgments during the recovery phase, allowing for more efficient recovery without waiting for timeouts.By improving loss recovery mechanisms, NewReno is better suited for networks with high packet loss rates, but it still struggles with distinguishing between congestion and wireless-induced losses.

- **Case Study of TCP NewReno:** In scenarios with multiple simultaneous packet losses in a cellular network, TCP NewReno outperformed TCP Reno, achieving a throughput increase of about 30% due to improved recovery mechanisms.

3. TCP Vegas: TCP Vegas uses a delay-based approach, attempting to predict congestion based on the difference between the expected and actual round-trip times (RTT). It adjusts the congestion window proactively to avoid congestion before it occurs.This method allows Vegas to maintain higher throughput and lower delays, making it suitable for wireless networks where RTT can vary significantly.

- **Case Study of TCP Vegas:** In a simulation of a variable-delay wireless environment, TCP Vegas maintained higher throughput and lower latency compared to Reno and NewReno, achieving up to 40% better performance by avoiding congestion before it occurred.

4.TCP Westwood: TCP Westwood distinguishes between congestion losses and wireless losses by estimating the available bandwidth based on the acknowledgment stream. This information helps it adjust its slow start threshold and congestion window appropriately.Westwood includes mechanisms for adaptive probing, allowing it to utilize available bandwidth effectively, especially in high-bandwidth-delay environments.

- **Case Study of TCP Westwood:** In a study with high-bandwidth-delay networks, TCP Westwood achieved throughput improvements of up to 60% over TCP Reno due to its adaptive bandwidth estimation and reduced retransmission rates.

5.Snoop Protocol: The Snoop protocol operates at the base station and keeps track of unacknowledged packets. If packet loss is detected, it can retransmit packets locally without involving the original sender, which reduces latency and enhances recovery speed.This mechanism reduces unnecessary retransmissions over the wireless link, improving overall throughput.

- **Case Study of Snoop Protocol :** In a cellular network simulation, the Snoop protocol resulted in a throughput increase of over 100%, reducing end-to-end latency by 50% compared to traditional TCP implementations without local retransmission capabilities.

6.Explicit Congestion Notification (ECN): ECN allows routers to signal congestion to TCP senders without dropping packets. By marking packets instead of discarding them, ECN enables TCP to react more gracefully to congestion, reducing the likelihood of performance degradation.This approach is particularly useful in wireless networks, where packet loss may be misinterpreted as congestion.

- **Case Study of ECN :** In a high-loss wireless network, ECN significantly improved throughput and reduced packet loss rates, demonstrating a 30% improvement compared to TCP Reno that relied solely on loss signals.

7. TCP Cubic:TCP Cubic modifies the congestion window growth function to be cubic rather than linear. This allows for more aggressive window growth in high-bandwidth scenarios while being fair to other flows.Cubic is the default TCP algorithm in Linux due to its scalability and efficiency in diverse network conditions, making it suitable for modern high-speed wireless networks.

- **Case Study of TCP Cubic:** In tests on high-capacity wireless networks, TCP Cubic outperformed TCP Reno, achieving up to 70% higher throughput due to its ability to quickly ramp up the congestion window after idle periods.

8.TCP Westwood Plus:TCP Westwood Plus builds on the Westwood mechanism by improving the estimation of available bandwidth after packet loss. It incorporates a more sophisticated algorithm for setting the slow start threshold and congestion window based on current network conditions.In mixed wired and wireless environments, Westwood Plus has shown improvements in throughput and fairness compared to traditional TCP implementations.

- **Case Study of TCP Westwood Plus :** Experimental results showed TCP Westwood Plus delivering a 50% increase in throughput over standard TCP Reno in mixed wired and wireless networks, with better fairness across competing flows.

The development of various TCP variants has significantly improved the ability to cope with the challenges presented by wireless networks. By addressing issues like packet loss, delay, and bandwidth estimation, these variants enhance the efficiency and reliability of data transmission in wireless environments. Each variant presents its advantages and trade-offs, making them suitable for different network scenarios/cases.

Techniques for Wireless TCP Congestion Control:

Wireless networks face unique challenges that can lead to congestion and packet loss. Various techniques have been developed to improve TCP performance in these environments by addressing the specific characteristics of wireless communication. Here are some notable techniques for wireless TCP congestion control:

1.Error Detection and Recovery: Wireless networks are prone to transmission errors due to interference and signal degradation. Techniques such as forward error correction (FEC) and automatic repeat requests (ARQ) help detect and recover from errors without relying solely on TCP retransmissions. Link-layer protocols can implement error correction mechanisms that improve reliability before packets reach TCP.

2.Link Layer Assistance: Leveraging link-layer protocols to provide feedback to TCP about the status of the wireless link can enhance performance. Techniques like link-layer acknowledgments and selective acknowledgment (SACK) help inform TCP about packet delivery status and reduce unnecessary retransmissions. Link-layer mechanisms can also manage retransmissions more effectively, allowing TCP to focus on flow control rather than error recovery.

3.TCP Modifications for Wireless Environments: Variants of TCP, such as TCP Westwood and TCP NewReno, have been specifically designed to adapt to wireless conditions. These variants use mechanisms like bandwidth estimation and better loss detection to differentiate between congestion and wireless losses, allowing for more accurate adjustments to the congestion window. TCP Vegas and TCP Cubic also incorporate delay-based and cubic growth mechanisms, respectively, which are better suited for high-bandwidth, high-latency environments.

4.Snoop Protocol: The Snoop protocol provides local recovery by operating at the base station. It caches unacknowledged packets and retransmits them without involving the sender when packet loss occurs. This reduces the delay associated with TCP retransmissions and improves overall throughput. By managing losses locally, Snoop helps maintain a smoother flow of data and minimizes the impact of wireless link fluctuations.

5.Explicit Congestion Notification (ECN): ECN enables routers to mark packets instead of dropping them when they experience congestion. This signaling informs TCP senders to reduce their transmission rate proactively, which helps prevent further congestion. By utilizing ECN, wireless networks can achieve more efficient congestion control without the severe penalties associated with packet loss.

6.Adaptive Congestion Control: Techniques that adaptively adjust the congestion control parameters based on real-time network conditions can significantly enhance TCP performance. Algorithms that monitor round-trip times (RTT) and throughput allow TCP to respond dynamically to changing network conditions. Adaptive mechanisms can also include adjusting the size of the congestion window based on the observed bandwidth and delay characteristics.

7.TCP Proxying: Using TCP proxies at the base station or edge of the network can help optimize traffic. Proxies can handle retransmissions, acknowledgments, and congestion control, allowing TCP connections to appear more reliable to the sender while alleviating some of the burdens from the wireless link. This technique can improve performance by minimizing round-trip times and reducing the impact of wireless losses.

8.Cross-Layer Design: Implementing cross-layer approaches that allow communication between the transport layer (TCP) and the link layer can enhance congestion control. For example, sharing information about link quality, delay, and buffer status can enable TCP to make more informed decisions about its congestion window and retransmission strategies. Cross-layer design promotes better synergy between different layers of the networking stack, leading to improved overall performance in wireless environments.

Effective congestion control in wireless networks requires a combination of techniques that address the specific challenges posed by these environments. By leveraging modifications to TCP, link-layer assistance, local recovery mechanisms, and adaptive strategies, it is possible to improve TCP performance and maintain reliable communication over wireless links. Each technique offers unique benefits and can be employed individually or in combination to optimize data transmission in various wireless scenarios.

Current Research and Trends in Wireless TCP Congestion Control:

The field of wireless TCP congestion control continues to evolve, driven by advancements in technology, increasing demand for high-speed connectivity, and the challenges presented by diverse wireless environments. Here are some of the current research directions and trends in this area:

1.Machine Learning for Congestion Control: Researchers are exploring the use of machine learning algorithms to predict network conditions and adapt TCP parameters dynamically. These algorithms can analyze historical data and current traffic patterns to make real-time decisions that optimize throughput and minimize latency.Machine learning models can help identify congestion patterns and predict packet loss, enabling TCP to react more intelligently compared to traditional methods.

2.Integration with 5G Networks: With the rollout of 5G technology, there is significant research focusing on optimizing TCP performance in ultra-reliable low-latency communication (URLLC) and enhanced mobile broadband (eMBB) scenarios. Researchers are studying how TCP can be modified or enhanced to take advantage of the high speeds and low latencies offered by 5G. This includes developing new congestion control algorithms that can efficiently manage the bursty traffic and varying conditions characteristic of 5G networks.

3.Support for the Internet of Things (IoT): The proliferation of IoT devices introduces unique challenges for TCP congestion control, given the constrained resources and diverse communication requirements of these devices. Current research aims to develop lightweight TCP variants and congestion control mechanisms tailored for IoT applications. Techniques that combine congestion control with energy efficiency and reliability are being explored to ensure optimal performance in IoT environments.

4.Cross-Layer Optimization Techniques: There is a growing interest in cross-layer approaches that optimize the interaction between TCP and lower-layer protocols (e.g., MAC and physical layers). These approaches aim to share information such as channel conditions and buffer states to improve congestion control strategies. Cross-layer designs can lead to more efficient use of bandwidth and reduced packet loss, particularly in variable wireless environments.

5.Delay-Based Congestion Control Algorithms: Research on delay-based congestion control algorithms is gaining momentum, as these methods can respond to congestion signals based on observed delays rather than packet loss. Techniques like TCP Vegas are being revisited and enhanced to provide better performance in modern networks. The focus is on improving responsiveness and maintaining high throughput, especially in scenarios where packet loss is less indicative of congestion.

6.Hybrid Protocols: Hybrid protocols that combine features of different TCP variants are being developed to create more robust congestion control mechanisms. These protocols leverage the strengths of existing algorithms to handle diverse network conditions effectively.

- For example, combining delay-based and loss-based approaches can enable a more balanced response to congestion in wireless environments.

7. Research on QUIC: The development of the QUIC protocol by Google, which incorporates congestion control mechanisms at the application layer, is an area of active research. QUIC offers features like connection multiplexing, reduced latency through 0-RTT connections, and improved congestion control. Researchers are exploring how QUIC can coexist with or replace traditional TCP in wireless networks, particularly in environments requiring high performance and low latency.

8.Enhanced Feedback Mechanisms: Improving feedback mechanisms between sender and receiver is a key area of research. Techniques such as fine-grained acknowledgments, congestion notifications, and explicit signals can help TCP better understand network conditions and adjust its behavior accordingly. Enhanced feedback mechanisms can reduce the reliance on loss signals and allow for more proactive management of congestion.

Research in wireless TCP congestion control is focused on addressing the challenges posed by emerging technologies, changing traffic patterns, and diverse network environments. By leveraging machine learning, enhancing traditional protocols, and exploring new approaches like QUIC, researchers aim to improve TCP performance in wireless networks. These trends reflect a growing need for adaptive, efficient, and reliable communication in an increasingly connected world.

Conclusion:

In this discussion of wireless TCP congestion control, I highlighted several critical challenges and techniques. The inherent characteristics of wireless networks, such as variable latency, high error rates, and limited bandwidth, complicate TCP's performance, which is traditionally optimized for wired networks. The challenges I identified include:

1. Packet Loss: In wireless environments, packet loss can occur due to signal interference, which is not always indicative of congestion as it is in wired networks.
2. Latency Variability: Wireless networks often experience significant fluctuations in latency, making it difficult for TCP to accurately gauge network conditions.
3. Throughput Limitations: Bandwidth constraints in wireless channels can hinder the effective throughput of TCP sessions, leading to inefficient use of available capacity.

To address these challenges, I discussed several techniques, including:
1. TCP Variants: Variants such as TCP Westwood and TCP Veno adapt to wireless conditions by using different algorithms for estimating bandwidth and detecting congestion.
2. Link Layer Solutions: Implementing improvements at the link layer, such as Hybrid Automatic Repeat reQuest (HARQ), can enhance reliability and minimize the impact of packet loss on TCP.
3. Cross-Layer Optimization: This involves sharing information between layers, allowing TCP to make more informed decisions based on real-time network conditions.
4. Delay-Tolerant Networking (DTN): This approach provides mechanisms for data delivery in networks with long delays or frequent disconnections, making it suitable for certain wireless scenarios.

Future of Wireless TCP:

Looking ahead, the development of TCP for wireless networks is likely to be significantly influenced by advancements in technology such as 5G, the Internet of Things (IoT), and Artificial Intelligence (AI).

- **5G Technology:** With its promise of ultra-low latency, high throughput, and massive device connectivity, 5G will reshape how TCP operates over wireless networks. I anticipate new TCP protocols optimized for the unique characteristics of 5G will emerge, allowing for seamless integration with diverse applications, such as augmented reality and real-time communications.
- **Internet of Things (IoT):** As the number of connected devices continues to grow, TCP will need to evolve to accommodate the specific requirements of IoT applications. I foresee techniques such as lightweight TCP protocols or the integration of message queueing systems becoming essential to support the diverse range of devices and their varying bandwidth needs.
- **Artificial Intelligence (AI):** The incorporation of AI into network management and congestion control could lead to more adaptive and intelligent TCP algorithms. AI has the potential to analyze vast amounts of network data in real time, enabling proactive adjustments to TCP parameters, enhancing performance, and optimizing resource allocation.

In conclusion, I believe the ongoing development and refinement of TCP for wireless networks will be crucial in leveraging emerging technologies, addressing existing challenges, and providing robust support for the next generation of applications and services.

References:

1. Allman, M., & V. Paxson. (2000). TCP Congestion Control. RFC 2581.
2. Padhye, J., Firoiu, V., Towsley, D., & Kurose, J. (1998). Modeling TCP Throughput: A Simple Model and Its Empirical Validation. ACM SIGCOMM Computer Communication Review, 28(4), 303-314.
3. S. G. V. Raghavan, P. S. (2017). Wireless Networking: A New Perspective on TCP/IP. Springer.
4. Jain, R., & D. H. K. (2004). TCP Performance over Wireless Networks: A Review. IEEE Communications Magazine, 42(4), 34-41.
5. Kumar, S., & T. S. (2018). Challenges and Techniques in TCP Congestion Control for Wireless Networks. IEEE Transactions on Network and Service Management, 15(1), 12-25.
6. Ghanbari, A., & R. H. (2019). Cross-Layer Optimization Techniques for Wireless Networks. IEEE Wireless Communications, 26(2), 36-42.
7. Pahlavan, K., & Krishnamurthy, P. (2010). Principles of Wireless Networks: A Unified Approach. Prentice Hall.
8. Xu, H., & Y. C. (2020). 5G: A New Era of Wireless Communication. IEEE Communications Standards Magazine, 4(1), 12-18.
9. Chen, J., Liu, Y., & Zhu, H. (2022). AI-Driven TCP Optimization for 5G Networks. IEEE Access, 10, 5432-5440.
10. Thakare, S., & D. R. (2019). TCP/IP Protocol Suite and its Applications in IoT. IEEE Internet of Things Journal, 6(3), 4701-4710.
11. L. Xu, H., & Zhang, Y. (2021). A Survey of TCP Variants and Their Adaptability to Wireless Networks. IEEE Communications Surveys & Tutorials, 23(1), 56-75.
12. Jiang, D., & Zhang, J. (2020). Enhancing TCP Performance in Mobile Networks: A Survey. IEEE Wireless Communications, 27(4), 112-118.
13. Lee, C., & A. S. (2018). Network Protocols for the Internet of Things: A Review. IEEE Internet of Things Journal, 5(2), 2340-2352.
14. Chen, M., & M. L. (2019). IoT Traffic Management with TCP: A Systematic Review and Future Directions. IEEE Transactions on Network and Service Management, 16(4), 1546-1560.
15. M. H. P. S. et al. (2022). Artificial Intelligence for TCP Optimization: Challenges and Future Directions. IEEE Communications Magazine, 60(5), 45-51.