

Identifying TCP Flows

- Two dictionaries, **tcp_flows** and **receiver_to_sender_packets**, track packets flowing in each direction. New flows are initiated upon encountering SYN packets, marking the start of a TCP connection, and populated with initial metrics like sequence numbers, timestamps, and TCP options.

Calculating Throughput

- For each flow in **tcp_flows**, the total number of bytes transferred is divided by the flow duration (difference between the first packet's timestamp and the last packet's timestamp).

Estimating Window Size

- I extract the window scaling factor from TCP options where available.

Congestion Window Estimation

- The function iterates over each TCP flow in the **tcp_flows** dictionary.
- It first checks if the initial round-trip time (RTT) for the flow is available.
- Only packets with payload are considered for cwnd estimation.

Estimating cwnd Sizes

- The estimation process begins with the timestamp of the first packet with payload.
- As it iterates through packets with payload, the function counts how many such packets are sent within an RTT interval which I take 0.07 and I calculated it within the TCP flow function (difference between SYN and SYN-ACK), from the reference timestamp.
- When the timestamp of a packet exceeds the current RTT interval, it signifies the start of a new RTT interval. The function then resets the reference timestamp to this packet's timestamp and starts counting packets for the next cwnd estimate.

Congestion window size grows constantly by 10s. It increases from 10 to 20 and then 33, having a difference of approximately 10 or more than that.

Analyzing Retransmissions and Triple Duplicate ACKs

- A dictionary **triple_dup_ack_flows** is initialized to store the count of triple duplicate ACK occurrences for each TCP flow identified in **receiver_to_sender_packets**.
- Also, I have made a dictionary of duplicate seq numbers with their timestamps.
- The function iterates through each flow's packets in **receiver_to_sender_packets**. For each packet starting from the third position in the list it performs the following checks:
- Checks if the current packet's ACK number matches the previous two packets' ACK numbers.
- For each ACK number encountering this situation, the function tracks the first occurrence's timestamp (**first_ts**), the latest occurrence's timestamp (**last_ts**), and a count of occurrences. If the count reaches 3 (indicating triple duplicate ACKs), the function proceeds to validate the retransmission.
- The function then checks if the ack number I got is in the duplicated sequence numbers or not and there was at least one packet retransmitted after the last triple

duplicate ACK by comparing the timestamps of packets with the expected sequence number's timestamps..

- If the conditions for a triple duplicate ACK are met, and a retransmission is validated, the count of triple duplicate ACKs for the flow is incremented. This process is repeated for each packet in each flow, compiling a comprehensive count of triple duplicate ACK events for all analyzed flows.

In my case, for the second flow, triple duplicate ack is not coming out right.

Calculating Potential Timeouts

- **Implementation:** For each sequence number, subtract the number of triple duplicate ACK-triggered retransmissions from the total retransmissions. Also, over here for the first and second flow, timeout is not coming out right. Its off by some numbers.

These are my answers:

```
TCP Flow: ('130.245.145.12', 43498, '128.208.2.198', 80)
  Transaction 1 SEQ: 705669103, ACK: 1921750144
  Transaction 2 SEQ: 705669127, ACK: 1921750144
  - Throughput: 5133395.748425832 bytes/sec
Scaled Window Size: 49152
TCP Flow: ('130.245.145.12', 43500, '128.208.2.198', 80)
  Transaction 1 SEQ: 3636173852, ACK: 2335809728
  Transaction 2 SEQ: 3636173876, ACK: 2335809728
  - Throughput: 1256538.3572691982 bytes/sec
Scaled Window Size: 49152
TCP Flow: ('130.245.145.12', 43502, '128.208.2.198', 80)
  Transaction 1 SEQ: 2558634630, ACK: 3429921723
  Transaction 2 SEQ: 2558634654, ACK: 3429921723
  - Throughput: 1448024.2286783182 bytes/sec
Scaled Window Size: 49152

TCP Flow: ('130.245.145.12', 43498, '128.208.2.198', 80)
  Congestion Window 1: 10 packets
  Congestion Window 2: 20 packets
  Congestion Window 3: 33 packets

TCP Flow: ('130.245.145.12', 43500, '128.208.2.198', 80)
  Congestion Window 1: 10 packets
  Congestion Window 2: 20 packets
  Congestion Window 3: 33 packets

TCP Flow: ('130.245.145.12', 43502, '128.208.2.198', 80)
  Congestion Window 1: 10 packets
  Congestion Window 2: 20 packets
  Congestion Window 3: 33 packets

Flow: ('130.245.145.12', 43498, '128.208.2.198', 80), Timeouts: 3
Flow: ('130.245.145.12', 43500, '128.208.2.198', 80), Timeouts: 94
Flow: ('130.245.145.12', 43502, '128.208.2.198', 80), Timeouts: 0
Flow: ('128.208.2.198', 80, '130.245.145.12', 43498), Triple Duplicate ACKs: 2
Flow: ('128.208.2.198', 80, '130.245.145.12', 43500), Triple Duplicate ACKs: 27
Flow: ('128.208.2.198', 80, '130.245.145.12', 43502), Triple Duplicate ACKs: 0
```