TCP Congestion Control: Slow Start

In this lesson, we're going to look at the slow start algorithm.

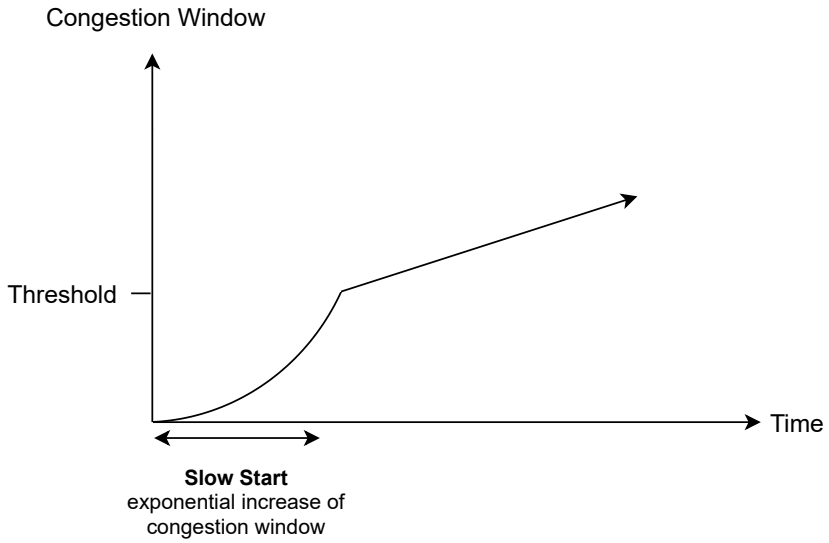
We'll cover the following

How Slow Start Works

The objective of TCP slow-start is to quickly reach an acceptable value for the congestion window.

During slow-start:

1. The congestion window is **doubled every round-trip time**.
2. The slow-start algorithm uses an additional variable in the TCB to maintain the **slow-start threshold**.
   - The slow-start threshold is an estimation of the last value of the congestion window that did *not* cause congestion.
   - It is initialized at the sending window and is updated after each congestion event.



Congestion Window Size Change with Slow Start

Not All Congestion Is the Same!

The TCP congestion control scheme distinguishes between two types of congestion:

1. Severe Congestion
2. Mild Congestion

Let's discuss these one by one.

Severe Congestion

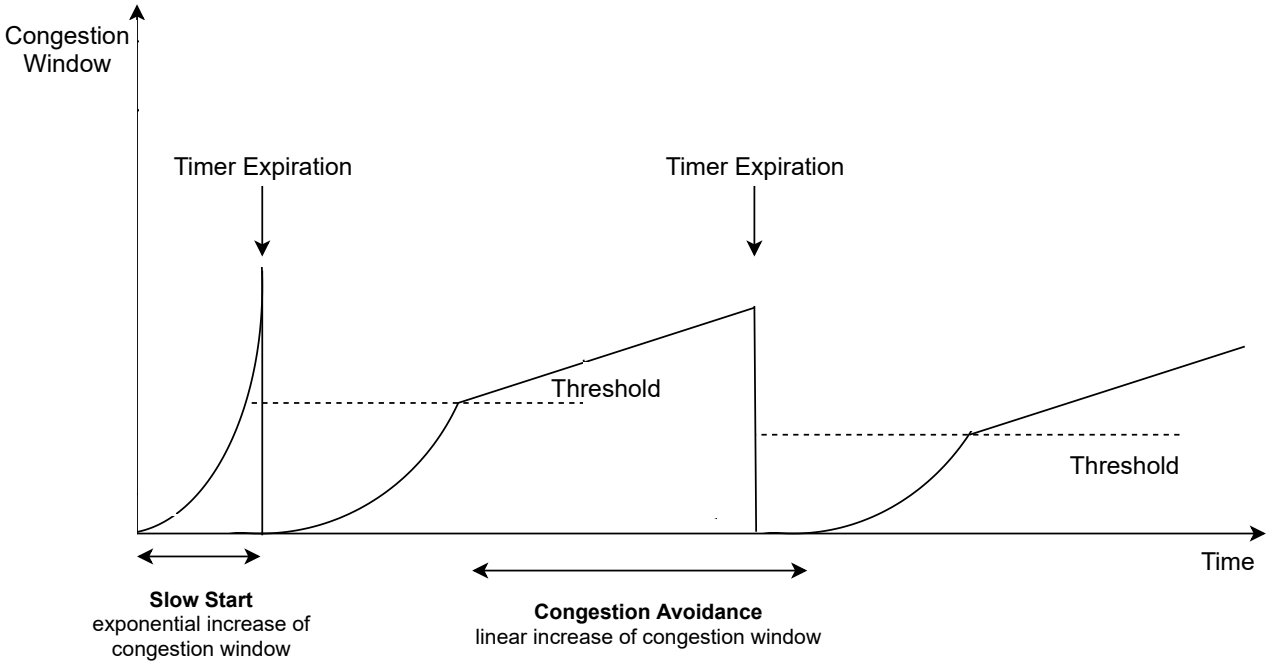
TCP considers that the network is severely congested when its retransmission timer expires. The following process is followed accordingly:

1. The sender performs slow-start until the first segments are lost and the retransmission timer expires.
2. At this time, TCP retransmits the first segment and the slow start threshold is set to half of the current congestion window. Then the congestion window is reset at one segment.
3. The lost segments are retransmitted as the sender again performs slow

3. The lost segments are retransmitted as the sender again performs slow-start until the congestion window reaches the slow start threshold.
4. It then switches to congestion avoidance and the congestion window increases linearly until segments are lost and the retransmission timer expires.

Figure

The figure below illustrates the evolution of the congestion window when there is severe congestion:

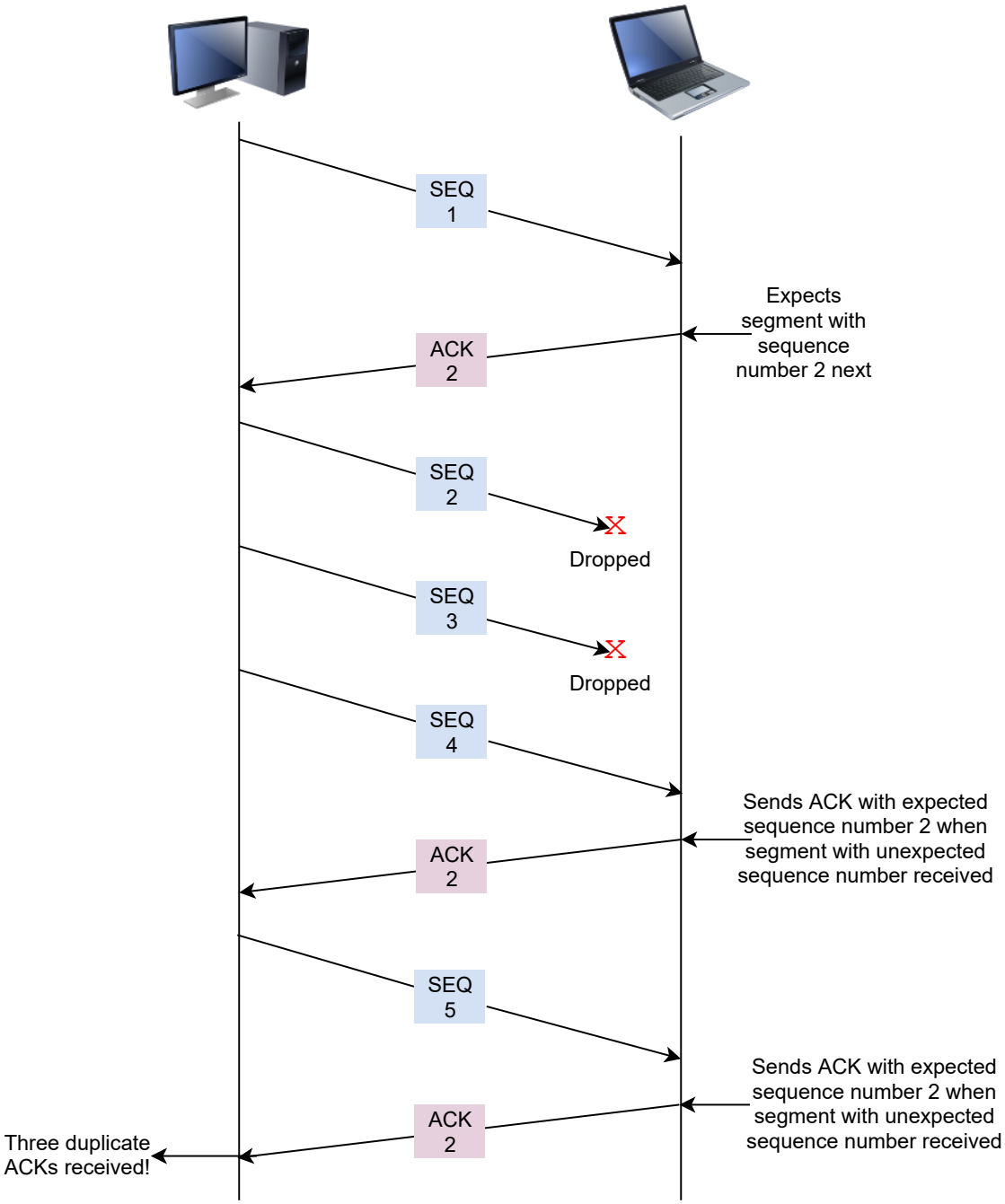


Evolution of the TCP congestion window with severe congestion

Mild Congestion

TCP considers that the network is lightly congested if it receives three duplicate acknowledgments.

1. The sender begins with a slow-start.
2. If 3 duplicate ACKs arrive, the sender performs a **fast retransmit** (retransmits without waiting for the retransmission timer to expire).
    - Have a look at the following slides to see when 3 duplicate acknowledgments could arrive and when a fast retransmit happens.



1 of 2

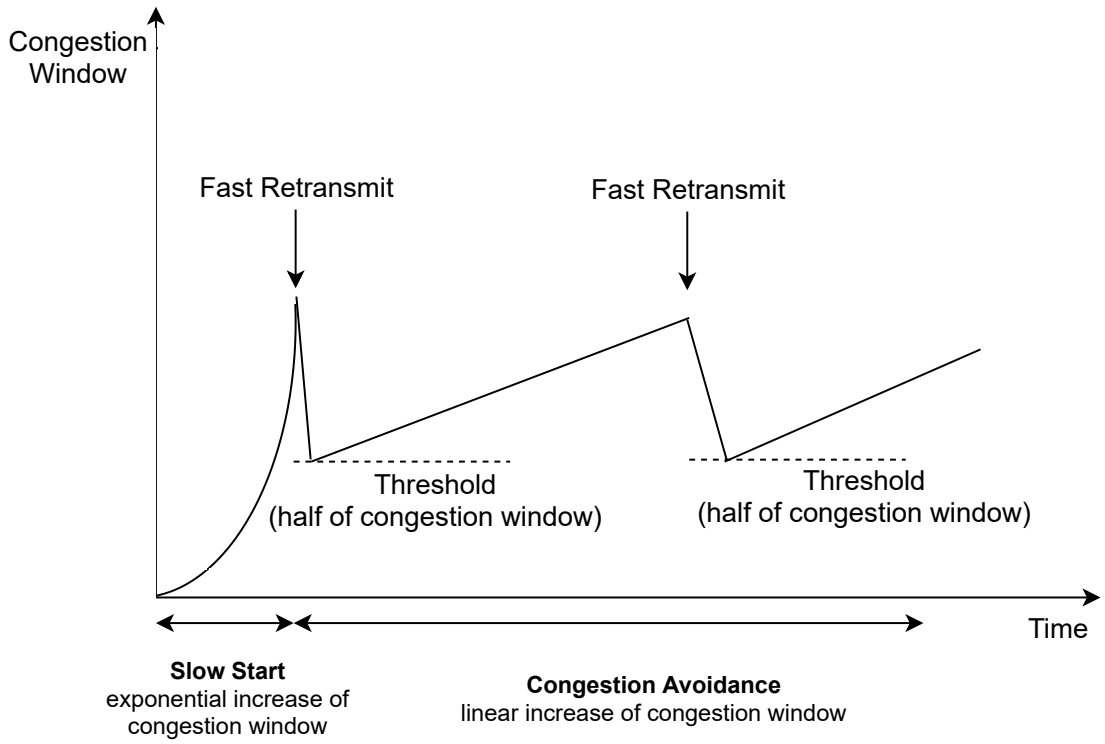
3. If the fast retransmit is successful, this implies that only one segment has been lost.
    - In this case, TCP performs multiplicative decrease and the congestion window is divided by 2.

- The slow-start threshold is set to the new value of the congestion window.

4. The sender immediately enters congestion avoidance as this was mild congestion.

Figure

The figure below illustrates the evolution of the congestion window when the network is lightly congested and all lost segments can be retransmitted using fast retransmit.



Evolution of the TCP congestion window when the network is lightly congested

Quick Quiz!

1

Slow start increases the congestion window size exponentially, whereas congestion avoidance increases the congestion window size linearly.

A)

True

B) False

COMPLETED 0%

1 of 3

That’s it for the transport layer! Let’s look at socket programming in Python next!