CS3205: Introduction to Computer Networks

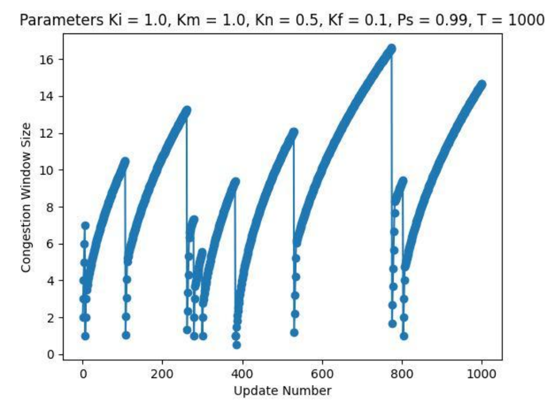
Assignment 3: Simulation of the TCP Congestion Control Algorithm

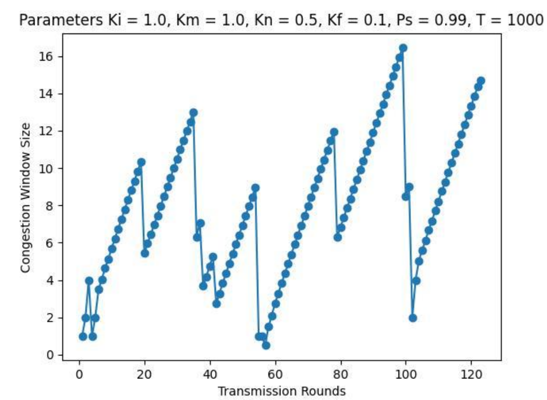
- Akshith Sriram Enadula, CS19B005

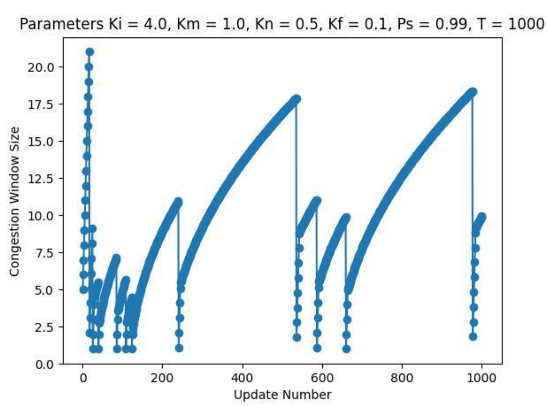
Simulation of the TCP Congestion Control Algorithm has been done for 32 different combinations of algorithm parameters as described:
$K_i \in \{1, 4\}$, $K_m \in \{1, 1.5\}$, $K_n \in \{0.5, 1\}$, $K_f \in \{0.1, 0.3\}$, $P_s \in \{0.99, 0.9999\}$

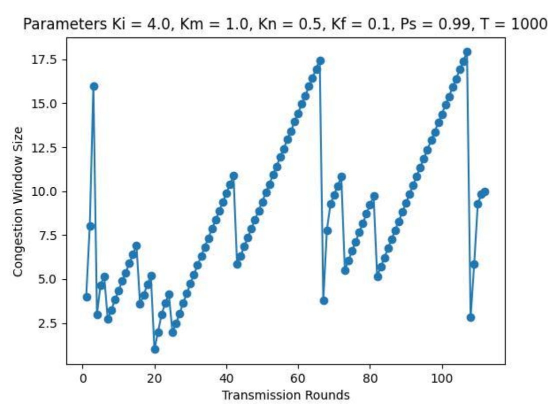
The instructions to execute the code are given in the README file of the submitted folder. For each set of parameters, two plots are shown. One plot depicts CW vs Update Number & another plot depicts CW vs Transmission round. An explanation for the influence of each of the parameters influences CW change is given below:

K_i , $1 \leq K_i \leq 4$ denotes the initial congestion window size (CW).







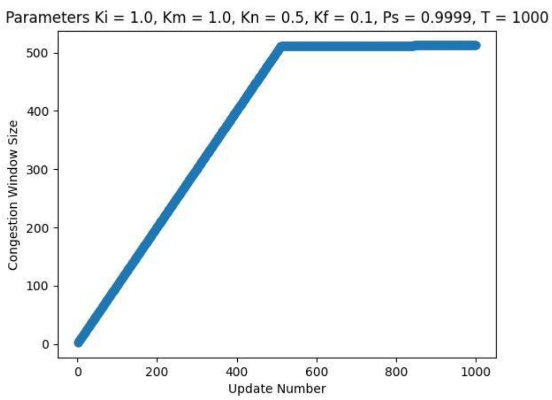


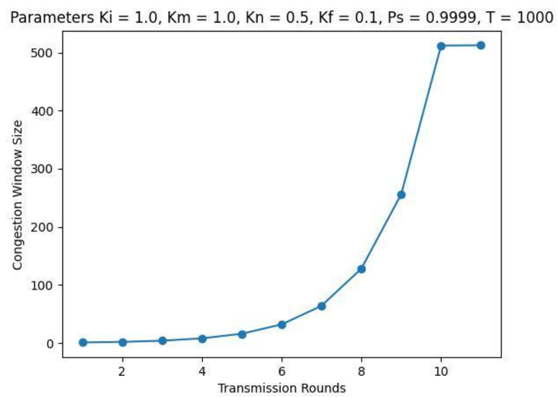
From the above plots, it can be seen that for different Ki (keeping other parameters the same), the initial CW value is higher for Ki = 4 than for Ki = 1.

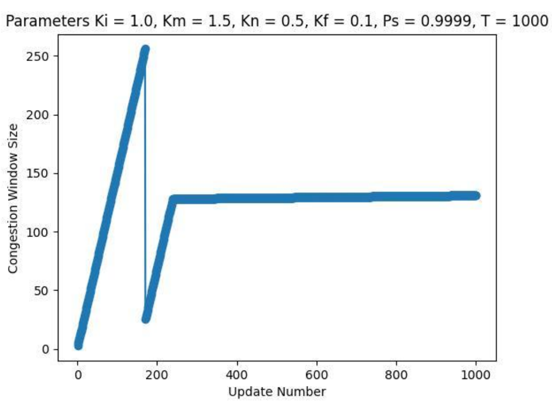
K_m , 0.5 ≤ K_m ≤ *2* denotes the multiplier of the Congestion Window, during the exponential growth phase.

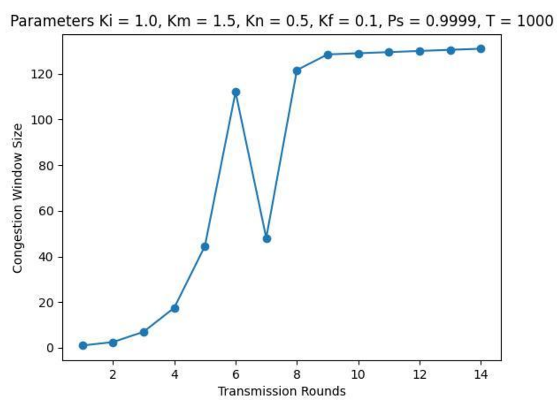
For a given value of Km, CW is calculated using the expression

$$CW_{new} = \texttt{min}(CW_{old} + K_m*MSS, RWS)$$









From the above plots, it can be seen that the growth in the CW (during the exponential/slow-start phase) is faster for a higher value of Km

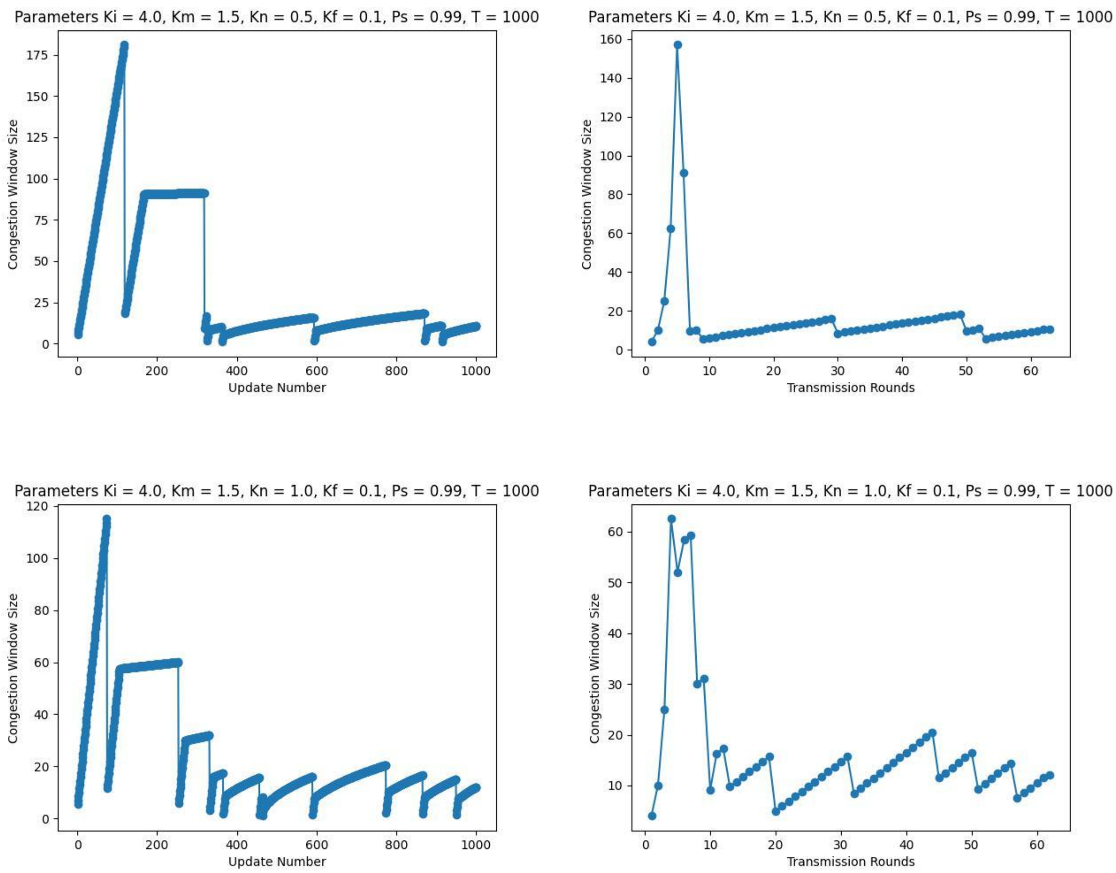
When Km = 1.0 (first two plots), the value of CW increased to 100 after 8 transmission rounds; whereas when Km = 1.5, the CW value increased to 100 within 6 transmission rounds itself.

With higher values of Km, the rate at which the no. of packets in flight (network traffic) increases (before reaching the threshold/timeout) is high.

K_n , $0.5 \le K_n \le 2$ denotes the multiplier of the Congestion Window, during the linear growth phase.

When an ACK packet for a sent segment is received successfully, CW is updated using the following expression:

$$CW_{new} = \min\left(CW_{old} + K_n * \frac{MSS*MSS}{CW_{old}}, RWS\right)$$



From the above plots, it can be seen that the growth in the CW (during the linear/Congestion Avoidance phase) is faster for a higher value of Kn

The effect of Kn can be seen in the slopes of the linear parts of the above plots.

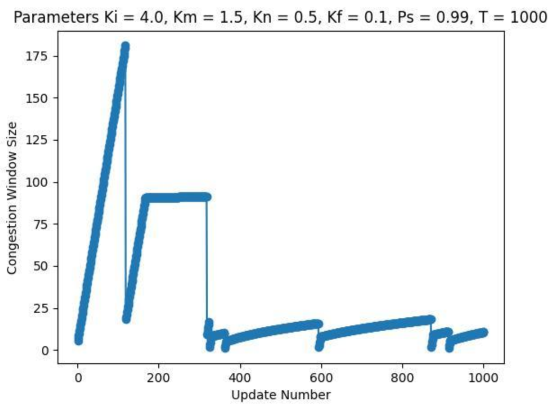
The slope of the linear portions of the plots where Kn = 0.5 (first two plots) is smaller than that of the plots where Kn = 1 (bottom two plots).

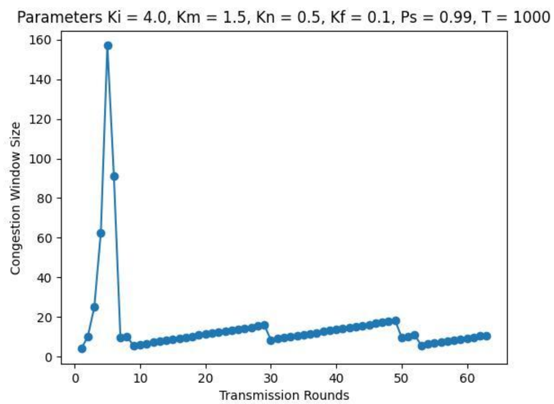
Kn indicates the rate at which the CW changes after it reaches the threshold.

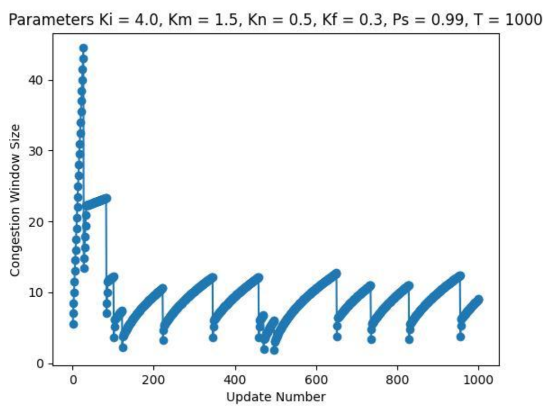
K_f **, 0.1 ≤** K_f **≤ 0.5** denotes the multiplier when a timeout occurs.

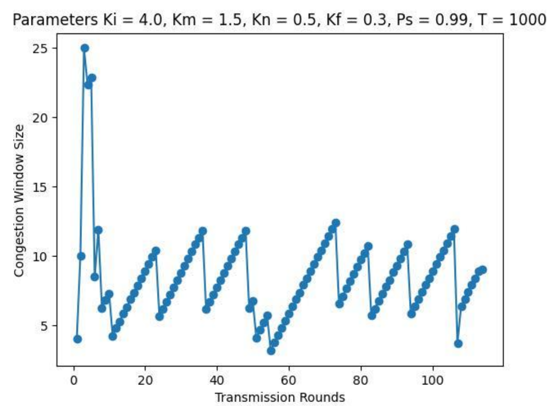
For a given value of Kf, the new value of CW is calculated (when a timeout occurs with probability Ps) as follows:

$$CW_{new} = \max(1, K_f * CW_{old})$$





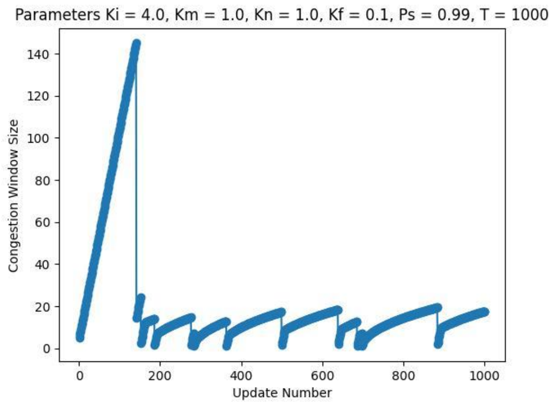


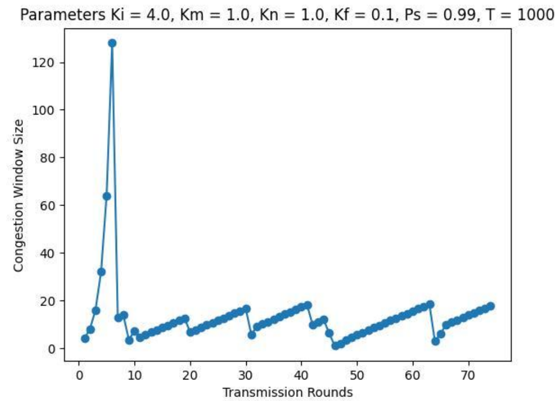


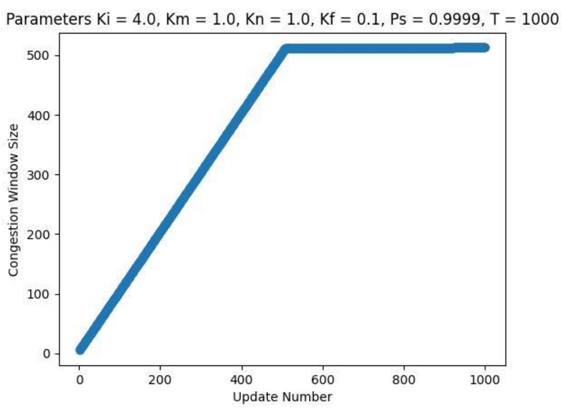
For Kf = 0.1 (first two plots), when a timeout occurs, the value of CW drops drastically to 10% of the current value (on a timeout)
In the last two plots, Kf = 0.3, therefore the value of CW drops to 30% of the current value (on a timeout)

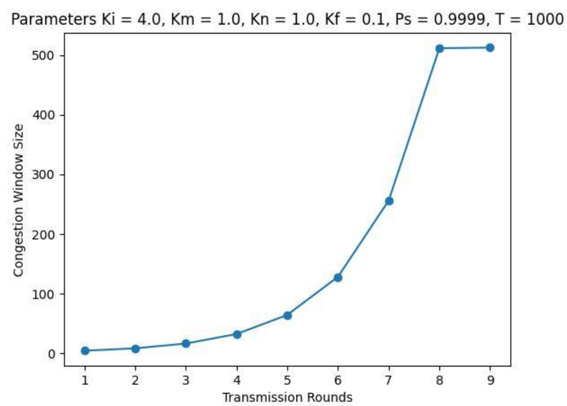
Therefore, as Kf increases, the drop in CW decreases. (when all other parameters remain the same)

P_s , $0 < P_s < 1$, denotes the probability of receiving the ACK packet for a given segment before its timeout occurs.









Since an ACK is received with a probability Ps, the probability of a timeout is (1 - Ps).

For Ps = 0.99 (top two plots), there is a 1% chance of a timeout. When 1000 segments are sent, approximately 10 packets will receive a timeout. This can be seen in the plots.

For Ps = 0.9999, the chance is 0.01%, which is very low. A timeout occurs very rarely. Even for 1000 packets sent, not even a single packet experiences a timeout. This can be seen in the bottom two plots, where there are no timeouts.

As the value of Ps increases, no. of timeouts decreases. (Keeping other parameters same)