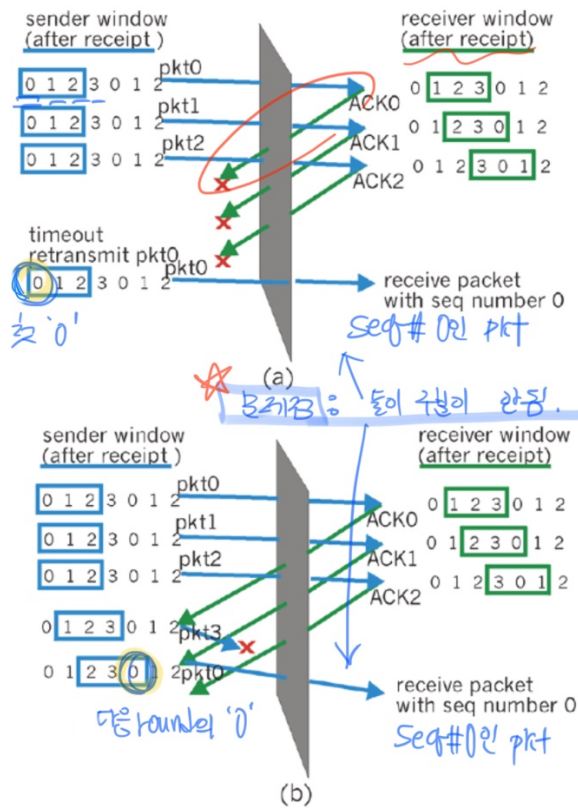


답안

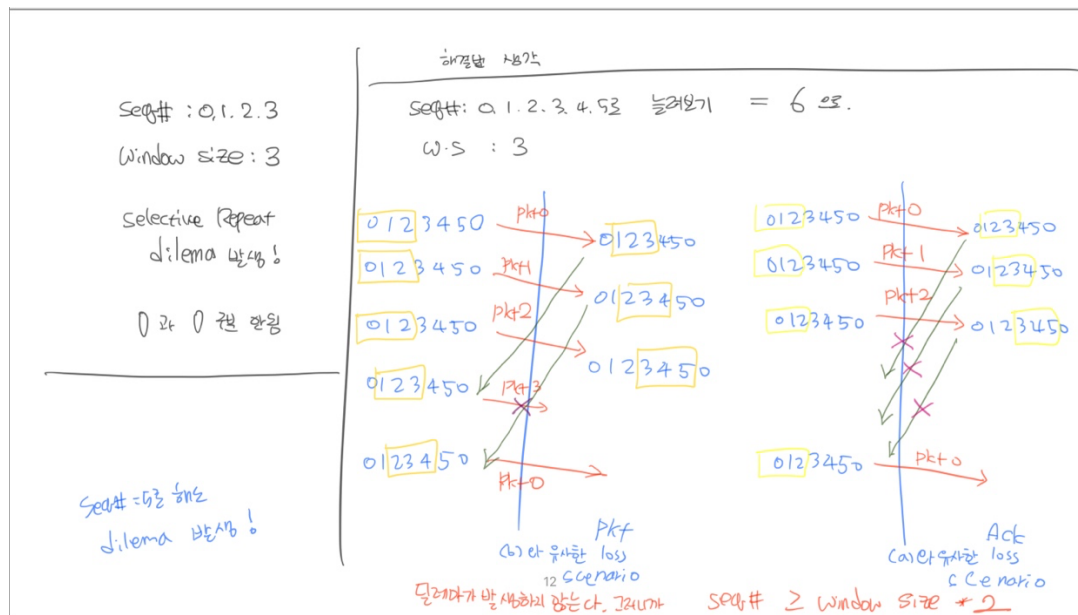
1. Sender는 처음 pkt 보낼 때 시작한 timer 가 time out 이 일어날 경우, 다시 pkt 를 보내게 되어있다. 때문에, receiver 가 pkt 을 정상적으로 받긴 했지만 ACK를 전송하기 이전에 이미 time out 이 발생해 무시당한다. 그래서 다시 retransmit 을 해주게 되는데, 이때 재전송으로 전송된 0이 그림 (a)의 receive packet with seq number 0 이다. '0'1230... 반면에 그림(b) 의 receive packet with seq number 0 의 경우, 재전송된 0이 아닌, 다음 round의 0 이다. 0123'0'... 결론적으로, 이 둘은 서로 다른 0이지만, 그림 (a) 에서 time out 으로 retransmit 된 seq#0의 pkt 과 그림 (b) 에서 정상적으로 transmit 된 next round 의 seq#0 의 pkt (receive packet with seq number 0)을 receiver가 구분할 수 없다는 문제가 발생하게 된다.

-incorrectly passes duplicate data as new in (a)

-receiver sees no difference in two scenarios



이러한 문제를 해결하기 위해서는, 애초에 window size를 seq#의 총 개수의 절반 이하로 설정해야 한다. 절반을 넘어가면 buffering 이 생겨 문제가 발생한다. 예를 들어, 주어진 상황과 같이 seq #'s : 0, 1, 2, 3 인 경우에는 window size가 2 이하여야 위와 같은 문제점이 발생하지 않는다. 윈도우 사이즈가 총 시퀀스 넘버 개수의 절반을 넘어가면, retransmit된 pkt과 slice 된 receiver의 window size 안의 pkt의 seq#가 중복되어, 구분되지 않는다. 또한 이미 수신된 pkt 을 엉뚱한 곳에 buffering 할 문제가 발생하기 때문에 seq#가 순환하지 않는다.



2. Sample RTT 가 1일 때, Sample RTT가 2가 된다고 가정을 하면, Estimated RTT는 1에서 12.5% 증가한 1.125 가 된다. Sample RTT의 경우, 1에서 2로 비교적 증가폭이 크지만, 그에 반해 Estimated RTT는 1에서 1.125로 증가폭이 비교적 작다. 때문에, Estimated RTT 가 smooth 하게 변경된다고 할 수 있다.