

17° Lezione

giovedì 21 novembre 2019

11:19

Controllo di congestione TCP

- Regola il tasso di trasmissione dei segmenti al fine di utilizzare pienamente la banda disponibile, evitando collassi di rete
- La banda disponibile:
 - Non è nota a priori
 - Varia nel tempo
- Analogamente al controllo di flusso è definita una finestra di congestione (**cwnd**) per limitare la dimensione della finestra scorrevole
- $W \leftarrow \min(\text{cwnd}, \text{awnd})$
- Il valore di cwnd è dinamicamente modificato nel tempo per mezzo di un opportuno algoritmo di **controllo di congestione**

Quando parliamo di TCP dobbiamo associare anche un numero di sequenza dei pacchetti: per farlo usiamo il campo sequence number contenuto nel pacchetto TCP.

Tra gli endpoint viene definita la Advertised Window che è solo la dimensione del buffer del ricevente, ma, tra ricevente e trasmittente, c'è una intera rete da percorrere. Il TCP controlla in più la congestione della rete. Misura questo campo tramite la Congestion Window. La finestra di trasmissione W è data dal minimo tra il valore della Congestion Windows e dell'Advertised Window.

Per definire la dimensione della Congestion Window utilizzo un algoritmo di nome AIMD (Additive Increase Multiplicative Decrease).

Il paradigma Additive Increase Multiplicative Decrease (**AIMD**)

- Utilizza due fasi per la regolazione dinamica di cwnd
- La fase **Additive Increase** incrementa progressivamente cwnd sino a quando si verifica un **episodio di congestione**
- La fase **Multiplicative Decrease** riduce drasticamente cwnd in seguito ad un episodio di congestione per scongiurare il **collasso della rete**
- **Episodi di congestione:**
 - Ricezione di 3 ACK Duplicati
 - Scadere di un RTO

Incremento additivamente la congestion window (in modo lineare) e quando rileva la congestione di rete, diminuisce la finestra in modo moltiplicativo, ovvero la riduce drasticamente per scongiurare il collasso di rete.

Ho bisogno della Congestion Window perché si perde il vantaggio del concetto della Sliding Window se la rete è congestionata e/o collassata.

Controllo continuamente la cwnd e fin quando è al di sotto di un concetto di soglia che è stato fissato empiricamente (controllando quella data rete so quando potrebbe essere congestionata) che indica il limite oltre il quale è molto probabile che si trovi congestione, allora la incremento additivamente. Se si verifica congestione (3 ack duplicati o allo scadere di un RTO) allora ci sono una serie di protocolli che consentono di gestire lo stato di congestione tra cui gli algoritmi **TCP Tahoe** e **TCP Reno**.

TCP Tahoe

- **Ricezione ACK:**
 - If ($cwnd < ssthresh$) then $cwnd = cwnd + 1$ (**slow start**)
 - else $Cwnd = cwnd + 1 / cwnd$ (**congestion avoidance**)
- **Ricezione 3 DUPACK:**
 - $Ssthresh = cwnd / 2$
 - $Cwnd = 1$
- **Timeout:**
 - $Ssthresh = cwnd / 2$
 - $Cwnd = 1$

TCP Reno

- **Ricezione ACK:**
 - If ($cwnd < ssthresh$) then $cwnd = cwnd + 1$ (**slow start**)
 - else $Cwnd = cwnd + 1 / cwnd$ (**congestion avoidance**)
- **Ricezione 3 DUPACK (FAST RECOVERY):**
 - $Ssthresh = cwnd / 2$
 - $Cwnd = Ssthresh$
- **Timeout:**
 - $Ssthresh = cwnd / 2$
 - $Cwnd = 1$

Es.

$W = \min(awnd, cwnd)$

Abbiamo un buffer di dimensione 6.

$awnd = 6$; $cwnd = 1$;

W sarà quindi uguale a 1.

Il threshold è uguale a 8. Il threshold ($ssthresh$) è il valore che ho citato sopra che si valuta empiricamente in base alla rete.

A questo punto incremento di una la $cwnd$ e arrivo a questa situazione:

$awnd = 6$; $cwnd = 2$;

W sarà quindi 2.

e così via incrementando $cwnd$ fino ad arrivare con $W = 6$ (rimarrà tale perché il minimo è $awnd$) e la $cwnd$ continuerà a crescere fino a quando non avverrà una congestione e, a quel punto, verrà ridotta drasticamente tramite uno dei due algoritmi (TCP Reno o Tahoe).

DOMANDA ESAME: DESCRIVERE IL CONTROLLO DI CONGESTIONE NEL TCP E COME VIENE GESTITO DA TCP TAHOE E TCP RENO.

Quando si parla di controllo di congestione si capisce finalmente come viene gestita la sliding window.

La sliding window è il rate della trasmissione ovvero il numero di pacchetti che posso inviare senza aspettare riscontro. La sliding window è dinamica ed è data dal minimo tra l' $awnd$ e la $cwnd$.

Il TCP Tahoe non distingue Timeout da

.

TCP New Reno

- Nel TCP Reno l'uscita dalla *fast recovery* avviene a ricezione dell'ACK relativo al segmento che ha causato l'ingresso nella *fast recovery* → basso throughput nel caso di perdite multiple nella medesima finestra
- Nel TCP New Reno l'uscita dalla *fast recovery* avviene quando viene riscontrata l'intera finestra di segmenti outstanding (in volo) all'istante dell'ingresso nella *fast recovery*

Reno TCP: evoluzione temporale della *cwnd*

