

Lezione 5 modulo 5

In questo video vediamo come migliorare le prestazioni del protocollo accesso multiplo casuale ALOHA, usando le tecniche che vanno sotto il nome di 'Listen Before Talk', ossia ascolta prima di parlare. Quando abbiamo studiato il protocollo ALOHA, abbiamo visto che sostanzialmente una stazione che parla ALOHA può trasmettere in maniera opportunistica sul canale tutte le volte che vuole, senza dover preventivamente controllare che il canale sia libero. Quindi da questo punto di vista, si dice che ALOHA è un protocollo maleducato nel senso che parla consente di parlare anche quando altre persone, altre stazioni stanno parlando sul canale, stanno utilizzando il canale. Un modo per rendere l'ALOHA più efficiente è quindi introdurre la cosiddetta funzionalità di ascolto del canale, che molto spesso va sotto il nome di carrier sensing, quindi di sensing del canale, del carrier, della portante, e evitare di trasmettere tutte le volte che, prima di decidere di trasmettere, il canale viene sentito come occupato, quindi se qualcun altro lo sta usando. Quindi usando tecniche di carrier sensing o Listen Before Talk, LBT, applicate all'ALOHA, potremmo evitare i casi in cui una stazione parta a trasmettere mentre qualcun altro sta già usando il canale, quindi evitando di trasmettere perché in questo istante, in cui la stazione B deve decidere se trasmettere o no, ascoltando preventivamente il canale, la stazione B può accorgersi che il canale è occupato e evitare di trasmettere e ad esempio allineare la propria trasmissione quando la stazione A ha terminato di trasmettere. In realtà, anche con la tecnica di ascolto del canale prima di trasmettere non riusciamo comunque ad eliminare in toto le collisioni perché comunque può sempre succedere che due stazioni che partecipano all'accesso multiplo casuale partano a trasmettere contemporaneamente. Quindi anche applicando la tecnica del carrier sensing, se due stazioni A e B sono esattamente sovrapposte nell'inizio delle loro trasmissioni, la collisione comunque può sempre esserci, perché al momento della trasmissione entrambe le stazioni sentono il canale libero, partono a trasmettere e la collisione avviene successivamente. In realtà, è possibile collidere nei meccanismi ALOHA con carrier sensing anche quando le trasmissioni non sono esattamente sincrone, ma quando le trasmissioni sono quasi sincrone. Le collisioni in questo caso sono dovute sostanzialmente alla differenza tra i ritardi di propagazione delle diverse trasmissioni concorrenti. Vediamo un esempio. Supponiamo ci siano due trasmissioni: trasmissione dalla stazione A e trasmissione dalla stazione B, supponiamo che queste due trasmissioni parlino Aloha con carrier sensing, quindi decidono di trasmettere solo se il canale è sentito libero prima della trasmissione stessa. La stazione A ha qualcosa da trasmettere in questo istante, applica carrier sensing, quindi ascolta il canale, il canale è libero in questo istante, quindi può partire a trasmettere e inizia la trasmissione del suo pacchetto verde. La stazione B ha qualcosa da trasmettere in questo istante, quindi in questo istante fa carrier sensing, quindi si mette in ascolto del canale e in questo istante tuttavia, dal momento che il primo bit della stazione A non è ancora arrivato alla stazione B, arriverà in questo istante, il canale è percepito come libero dalla stazione B stessa, quindi il carrier sensing, tra virgolette, fallisce, sente il canale libero quando in realtà non lo sarà. La stazione B parte a trasmettere e le due trasmissioni, quella verde e quella blu, collidono da qualche parte ad esempio a una stazione C, quindi si sovrappongono a una stazione C, generando una collisione. Quindi anche con il carrier sensing e anche con stazioni che non iniziano esattamente nello stesso istante a trasmettere, si può comunque avere collisioni. Il protocollo ALOHA più il carrier sensing, quindi più la feature di ascolto del canale prima di trasmettere, generalmente è noto con il nome di CSMA, carrier sensing multiple access: un protocollo, diciamo così, inventato e poi utilizzato nel 1975 in avanti. Le collisioni possono avvenire anche nel caso opposto rispetto all'esempio precedente, ossia nel caso in cui sia la stazione B che parte a trasmettere prima, quindi se la stazione B è quella che parte a trasmettere prima, sente il canale libero, parte a trasmettere, inizia la sua trasmissione, la stazione A inizia la sua trasmissione in questo istante, in questo istante il canale sentito è libero perché i bit della trasmissione di B non sono ancora arrivati fisicamente ad A, quindi il canale di fatto è libero, la stazione A trasmette e ancora una volta si ha collisione da qualche altra parte in una stazione C. Riassumendo quindi, per tecniche di CSMA, esiste un cosiddetto periodo di vulnerabilità, quindi un periodo di tempo a partire dall'istante di inizio di una certa trasmissione in cui cioè il carrier sensing è ceco, detto in altra maniera, se



una stazione inizia a trasmettere all'interno del periodo di vulnerabilità, allora il carrier sensing fallisce, quindi il canale viene sentito come libero quando in realtà non lo sarà, quindi quando in realtà si potrà andare a collidere con la stazione che ha iniziato o inizierà la propria trasmissione nell'istante 0. Il periodo di vulnerabilità dipende ovviamente dal ritardo di propagazione e in particolare il periodo di vulnerabilità è definito come due volte il ritardo di propagazione che esiste tra le due stazioni di cui stiamo parlando. È per questo che l'efficienza del meccanismo CSMA dipende fortemente dal ritardo di propagazione o meglio dipende dal rapporto che esiste tra il ritardo di propagazione e il tempo di trasmissione dell'informazione, rapporto che generalmente si indica con la lettera a . Un sistema CSMA ben dimensionato deve essere tale per cui a è minore di 1 quindi il ritardo di propagazione è minore del tempo di trasmissione dell'informazione e il sistema CSMA funziona tanto meglio quanto più piccolo è il parametro a quindi quanto più piccolo è il tempo di propagazione. Verosimilmente, il sistema CSMA ha efficienza massima quando il ritardo di propagazione è nullo, ossia quando la velocità di propagazione dell'informazione del mezzo è infinita. Viceversa, se il parametro a è maggiore di 1, quindi se il ritardo di propagazione è maggiore del tempo di trasmissione, allora il CSMA è totalmente inutile, quindi la feature di ascolto del canale è totalmente inutile perché il primo bit della trasmissione arriva a chi sta sentendo il canale quando in realtà la trasmissione è già finita, quindi il canale viene sentito occupato quando in realtà è libero. Dal punto di vista delle prestazioni, è possibile dimensionare le prestazioni del sistema CSMA usando un modello simile a quello visto nel video sul protocollo ALOHA. Non vediamo qui la dimostrazione delle prestazioni del sistema CSMA, ma analizziamo semplicemente la formula finale, formula finale che è data dall'equazione della slide, che lega, come per l'ALOHA, il throughput al traffico offerto G . Questa formula, rispetto all'ALOHA, dipende anche ovviamente dal parametro a appena definito, quindi il rapporto tra τ e T , ritardo di propagazione e tempo di trasmissione. Nella figura, nel grafico è mostrato appunto il throughput sul traffico per diverse configurazioni di sistema CSMA con il parametro a che via via decresce, quindi da a uguale a 0.1, a uguale a 0.01, 0.001, fino ad a uguale a 0. a uguale a 0, abbiamo detto, è il caso in cui il ritardo di propagazione è nullo, quindi velocità di propagazione infinita. In questo caso, come si vede, il CSMA ha le prestazioni migliori, ottime e non presenta instabilità, quindi al crescere del traffico, il throughput del CSMA arriva ad essere uguale a 1, quindi di fatto passa tutto quello che viene generato, o meglio, i singoli slot del sistema vengono usati appieno con efficienza pari a 1. Viceversa, scendendo in questa direzione, quindi aumentando il parametro a , aumentando il ritardo di propagazione rispetto al tempo di trasferimento dell'informazione, le prestazioni del CSMA scendono dal punto di vista del throughput massimo e si ritorna ad avere instabilità, ossia si ritorna ad avere un comportamento bistabile del sistema, per cui, dopo un certo traffico, le prestazioni del CSMA tendono a scendere esattamente come succedeva per l'ALOHA. Quindi asintoticamente tendono a throughput uguale a zero. Per migliorare un attimo le prestazioni del CSMA, in alcune tecnologie di livello fisico, di livello di collegamento, è possibile usare la cosiddetta caratteristica o funzionalità di collision detect. si parla quindi di CSMA con collision detect. In buona sostanza, la tecnologia di collision detect è quella per cui due stazioni che stanno trasmettendo e che si accorgono di una collisione evitano di continuare a trasmettere, ad esempio quello della slide in cui ho due stazioni A e la stazione dalla stazione B, le due stazioni parlano carrier sensing quindi iniziano a trasmettere in due istanti diversi, le due stazioni iniziano a trasmettere nel periodo di vulnerabilità, quindi il carrier sensing è ceco, non si vedono una con l'altra, e a un certo punto quindi collidono, quindi la stazione A inizia a ricevere i bit della stazione B in questo istante e la stazione B inizia a ricevere i bit della stazione A in questo istante. Se le due stazioni usano la funzione di collision detect, sono in grado di riconoscere la collisione mentre stanno trasmettendo. Quindi la stazione A si accorgerà di collidere con la stazione B in questo istante e la stazione B si accorgerà di collidere con la stazione A in questo istante. Se si accorgono di stare collidendo, possono evitare di continuare a trasmettere, quindi possono bloccare la trasmissione nella stazione A in questo istante e nella stazione B in questo istante, evitando cioè di occupare il canale con bit inutili perché sicuramente collisi e quindi riducendo di fatto il traffico sul canale stesso. Nella pratica, la funzione di collision detect, ad esempio usata nello standard Ethernet che vedremo in video futuri, non blocca la trasmissione esattamente quando la



collisione viene scoperta, ma la blocca dopo un certo ritardo δ . Quindi, se questo è l'istante in cui la stazione A scopre la collisione con la stazione B, la trasmissione non viene bloccata seduta stante, ma aspetta un certo ritardo δ , ritardo δ che serve sostanzialmente per far sì che l'informazione sulla collisione si propaghi ancora un po' all'interno della rete locale, all'interno del canale broadcast, in maniera tale che anche altre stazioni possano in qualche modo evitare di collidere, interrompendo le rispettive trasmissioni. Esattamente come per l'ALOHA, e per il carrier sensing base, è possibile derivare un modello di prestazione del carrier sensing con collision detect che lega ancora una volta il throughput S con il traffico G e, essendo un meccanismo basato sul carrier sensing, il periodo di vulnerabilità. Nella slide, ancora una volta è mostrato il throughput rispetto al traffico delle due versioni di carrier sensing, quindi quella base senza collision detect e quella con collision detect, al variare ancora una volta del parametro a . Esattamente come per la versione di carrier sensing base, anche la versione di carrier sensing un collision detect ha delle prestazioni che decrescono all'aumentare del parametro a e, dalla figura, è facile evincere come introdurre la funzione di collision detect, quindi le curve rosse, consenta di avere delle prestazioni che sono superiori rispetto alle curve blu che rappresentano invece il carrier sensing base. Ad esempio, nel caso in cui il parametro a sia uguale a 0.01, usare la funzione di collision detect, quindi la curva rossa qui, consente di avere prestazioni migliori rispetto a non usare la curva blu e in più consente di stabilizzare il funzionamento del sistema, quindi consente di evitare che il throughput vada a zero quando il traffico sale sopra un certo valore. La funzione di collision detect, oltre a migliorare le prestazioni in assoluto del CSMA, ha anche un vantaggio ulteriore, vantaggio che riguarda il fatto che se due stazioni o se n stazioni implementano funzioni di collision detect, di fatto non è più richiesto il riscontro di trasmissioni, non è più richiesto banalmente perché una stazione è in grado di capire localmente se la sua trasmissione è collisa oppure no, semplicemente osservando il canale, senza dover cioè aspettare un riscontro dal destinatario della sua trasmissione stessa. Ovviamente se il riscontro non è necessario, si evita di mandarlo e questo consente di ridurre ulteriormente il traffico sui canali perché i riscontri, come abbiamo detto tante volte, usano le stesse risorse usate da trasmissioni dati. Purtroppo però la funzione di collision detect non è sempre usabile, quindi non tutte le tecnologie sono in grado di usare la funzione di collision detect. Il motivo è fisico, nel senso che la funzione di collision detect si basa, come dicevamo prima, sulla capacità di un trasmettitore che sta usando un'interfaccia di rete per trasmettere informazioni, di capire che, sulla stessa interfaccia di rete, c'è anche un'altra trasmissione. Quindi la funzione di collision detect è usabile in generale in tutti quei casi in cui la potenza del segnale trasmesso, quindi la trasmissione che si sta effettuando, è in qualche modo comparabile con la potenza del segnale che si sta ricevendo. Questo è vero ad esempio in tutti i casi di reti locali in cui cioè il trasmettitore e il ricevitore sono, tra virgolette, vicini e/o sono direttamente collegati su un mezzo fisico cablato di un qualche tipo. Viceversa, la funzione di collision detect non è usabile in tutti quei casi in cui la potenza di segnale trasmesso è ordini di grandezza superiore rispetto alla potenza del segnale ricevuto, la trasmissione che sta collidendo, ossia la funzione di collision detect non è usabile quasi mai nel caso di collegamenti broadcast basati su reti radio, su reti wireless, perché in reti wireless la potenza del segnale trasmesso è ordini di grandezza superiore rispetto alla potenza del segnale che si riceve da ogni altra possibile trasmissione concorrente.

