Relazione 2

Stime della capacità a partire da misure di RTT

Mattia Chiarle 269969, Michele Ferrero 268542, Gabriele Ferro 268510

Gruppo 22 Anno 2021/2022

Introduzione

L'esperienza consiste nel raccogliere i dati necessari per valutare la capacità effettiva del canale tra due host sia con uno switch da intermediario sia senza. Per ottenere ciò viene utilizzato lo script mostrato, il quale permette di generare del traffico sulla rete tramite il comando ping, ottenendo i valori di RTT necessari per il calcolo della capacità. Lo script è suddiviso in 4 parti, ognuna delle quali si occupa di un determinato range di dimensioni del payload, da 16 byte a 10000 byte, con passo diversificato in base alle necessità di campionare maggiormente in alcuni range di dimensione per poter apprezzare meglio eventuali comportamenti attesi dalla teoria. Questa versione dello script è stata usata per ripetere i test a casa; quella precedente, sfruttata durante il laboratorio, si limitava a effettuare misure soltanto fino a 3000 byte.

Per ogni dimensione del payload si va a salvare, all'interno di un file, la dimensione del segmento seguita dai valori di RTT minimo, medio e massimo calcolati automaticamente dal comando ping, sui 5 ping inviati dall'istruzione. È stato deciso di effettuare 5 ping per ottenere i vari RTT in modo da minimizzare il più possibile variazioni dovute a fattori esterni, come collisioni o il carico della CPU.

I dati ottenuti vengono poi analizzati tramite i grafici RTT(D) e C(D), rispettivamente il Round Trip Time (RTT) in funzione della dimensione dei dati inviati (D, che include al suo interno anche i vari header) e la capacità in funzione sempre di D.

Lo scopo è quello di riuscire ad analizzare il comportamento dei grafici, ponendo particolare attenzione attorno alla soglia di frammentazione dei segmenti. Quest'ultima sarà presente in prima istanza per S=1472 B in quanto, in seguito all'aggiunta degli header ICMP e IP, avrà la dimensione della MTU del livello 2.

L'esperienza viene ripetuta tramite collegamento diretto e attraverso uno switch, utilizzando le diverse velocità offerte dalle schede di rete: 10/100/1000 Mb/s sia per il collegamento diretto sia per il collegamento tramite switch. Il setup prevede esclusivamente 2 PC dotati di presa ethernet ed eventualmente lo switch.

Per impostare la velocità del canale si utilizza il comando ethtool che permette di modificare le velocità che vengono offerte nel momento della negoziazione di rete nel collegamento ed il tipo di collegamento, prestando particolare attenzione alla scelta del duplex "full" per evitare ritardi dovuti al trasmettitore.

Generazione dei grafici

I grafici utilizzati nell'analisi vengono generati dallo script mostrato. Il comando multiplot iniziale permette di poter visualizzare entrambi i grafici ottenuti in parallelo; in particolare, 1,2 indica che verrà usata per la visualizzazione un matrice di una riga e due colonne, e rowsfirst indica che i plot verranno distribuiti prima lungo le righe. Lo stesso script viene utilizzato sia per le misure di ping dirette sia per quelle passanti attraverso lo switch: visto che i due casi richiedono delle costanti diverse è stato sfruttato un flag switch che, in base al suo valore, permette di implementare i calcoli corretti nella funzione D_sw(s). Infine, set title permette di assegnare un titolo ai grafici, mentre set xtics e set ystics permettono di impostare il font e la dimensione dei valori numerici dell'asse x, y che vengono visualizzati.

La prima parte dello script permette di realizzare il grafico del RTT in funzione di D: con xlabel e ylabel si rinominano gli assi, indicando sia la variabile corrispondente all'asse sia la sua unità di misura. Il comando plot permette infine di realizzare effettivamente il grafico.

La seconda parte dello script serve invece a realizzare il grafico della capacità in funzione di D.

In questo caso è tuttavia necessario differenziare il caso con lo switch rispetto a quello senza switch: nel primo, infatti, avremo

$$1)\ RTT = 2T_{tx} + \eta$$

con η trascurabile. Poiché Ttx=D/C, si ricava che C=2D/RTT.

Nel secondo caso invece, a causa del funzionamento store and forward dello switch, si verificano due comportamenti diversi in base al numero di frammenti. Con un solo frammento avremo infatti che il RTT sarà pari a 4Ttx, mentre con più frammenti avremo

$$2)\ RTT = 2T_{tx,\ tutti\ i\ frammenti} + 2T_{tx,\ primo\ frammento\ (MTU)} + \eta$$

con η sempre trascurabile. Per giungere a questi risultati sono stati trascurati i tempi di propagazione e si è supposto che i due canali (H1-switch e switch-H2) avessero la stessa capacità.
Una volta fatta questa introduzione teorica abbiamo quindi implementato quasi esattamente queste formule a livello di codice. Per leggibilità è stata definita come costante ip_eth, ovvero la dimensione degli header ip e ethernet sommati. Sono poi state definite le seguenti funzioni:

1. byte_framm(s): dato s, calcola il numero di byte totali dell'ultimo datagram IP. Per farlo ottiene il resto della divisione di s+8 per 1480 (è stato necessario definire s+8 come int(s+8) in quanto, anche se s è effettivamente un intero, senza l'inserimento di int gnuplot non lo riconosce come tale) e gli somma 20, ovvero la dimensione dell'header IP.
2. D(s): dato s (la dimensione del payload di livello 4) gli somma i vari header che verranno aggiunti procedendo nella pila protocollare. Sappiamo infatti che a ogni payload di livello 4 verranno aggiunti l'header ICMP (8 B), IP (20 B) e ethernet (38 B), oltre all'eventuale padding. In caso di frammentazione sappiamo inoltre che ad ogni frammento oltre al primo, già considerato precedentemente, verranno aggiunti tutti gli header elencati tranne l'header ICMP, in quanto la frammentazione avviene nel livello 3. Il numero di header IP e Ethernet (la cui dimensione totale è pari a 58 B, definita in ip_eth) da aggiungere viene quindi calcolato come floor((s+8-1)/1480). È stato necessario inserire floor in quanto senza si verifica un'approssimazione non desiderata, e questo comporta la possibilità errata di avere dei valori di D decimali. La sottrazione per 1 è necessaria in quanto altrimenti se ad esempio la dimensione di s fosse 1472 B, che insieme all'header IP e ICMP corrisponde alla MTU, si otterrebbe 1480/1480=1, che tenendo conto del primo frammento sempre presente porterebbe a sommare 2 ip_eth invece di uno erroneamente. Infine, il calcolo del padding si effettua ottenendo, grazie a byte_framm(s), il numero di byte dell'ultimo datagram IP tenuto conto della frammentazione. Se la sua dimensione è minore di 46 B verrà aggiunto del padding fino ad arrivare ai 46B minimi per il payload di Ethernet.
3. D_sw(s): dato s (sempre la dimensione del payload) per effettuare il calcolo di D viene prima analizzato il flag "switch" per capire se lo switch sia collegato o meno. Se questo è attivo si verifica l'eventuale presenza di frammentazione, ovvero se il payload del livello 4 ha una dimensione maggiore di 1472 B. In caso affermativo si somma al risultato di D(s) il valore 1538, ovvero una MTU con intestazione Ethernet, che sarebbe la dimensione del primo pacchetto. Infatti, il caso dello switch con frammentazione è equivalente a un ipotetico caso senza switch aggiungendo ai dati inviati una MTU iniziale a causa dello store and forward. Si moltiplica infine il tutto per due in modo da ottenere la stessa situazione presente nella formula [2].
   Se non è presente la frammentazione, invece, si moltiplica il risultato di D(s) per 4 sempre a causa dello store and forward. Come ultimo caso, se lo switch non è collegato, si moltiplica il risultato di D(s) per 2 per la formula [1].

Una volta definite queste funzioni di supporto si può quindi procedere alla realizzazione dei grafici. Ricordando quindi le formule per il calcolo della capacità ricavate precedentemente si ottiene

$$C = \frac{D_{sw}}{RTT}$$

In questo modo si otterrebbe però la capacità misurata in [B/ms]. Per ottenere come unità di misura [b/s], ovvero quella comunemente utilizzata, si moltiplica D_sw per 8 (per passare da B a b) e si divide RTT per 1000 (per passare da ms a s).

Analisi dei risultati

I grafici rappresentano il RTT e la capacità in funzione di D alle diverse velocità prese in esame.

Collegamento diretto degli host

1 - Capacità 10Mb/s

Il grafico 1.1 rappresenta i dati raccolti durante l'esperienza in laboratorio per quanto riguarda la velocità di 10Mb/s tramite collegamento diretto degli host, mentre il grafico 1.2 mostra i risultati ottenuti ripetendo il test a casa.
Quello che ci si aspetta è che nell'intorno di 1500 (per la precisione S=1472 e quindi D=1538 ovvero 1 MTU a livello 2+1 Ethernet header) si dovrebbe avere un RTT pari a 2.46ms; nella realtà però, per ritardi imputabili alle schede di rete utilizzate o ad altri fattori influenzati dal setup e dal ping, durante l'esperimento il ritardo totale per quel dato D è pari a circa 5ms. Questo problema si ripercuote anche nella capacità effettiva: infatti, essendo C=(D/RTT)*1000 e avendo un RTT maggiore rispetto a quanto ci si aspetta, la conseguenza è che anche C sia inferiore al valore atteso (in particolare visto che il RTT è circa il doppio del valore teorico la capacità viene dimezzata). Se facessimo tendere D ad un numero molto grande, ciò che ci si aspetta è che la capacità effettiva tenda a circa 10 Mb/s ed abbia delle cadute nei punti in cui si crea un nuovo frammento, portando a delle discontinuità nella crescita.
Un'ipotesi formulata per quanto riguarda questo strano comportamento è che, poiché all'aumentare di S la capacità converge verso un valore sempre più vicino ai 10 Mb/s imposti, i ritardi presenti nel RTT possono essere dovuti a ottimizzazioni interne della scheda di rete o del comando ping, che non mandano istantaneamente ogni pacchetto per non inondare la rete di tanti piccoli pacchetti. Di conseguenza, probabilmente eseguono una breve fase di pipelining per poi concentrare l'invio in un lasso di tempo più breve. Si può inoltre notare come questo comportamento si sia presentato soltanto nelle misure effettuate in laboratorio: infatti, in quelle fatte a casa la capacità tende già a 9 Mb/s alla prima frammentazione, facendoci quindi supporre che il rallentamento presente nei RTT fosse dovuto alle schede di rete dei computer usati durante la lezione.
La discontinuità attesa data dalla non esistenza di alcuni valori di D (per via della frammentazione) intorno al valore di circa 1538 non è molto evidente, poiché i valori raccolti durante la misura vengono poi interpolati linearmente portando a mostrare una linea retta tra i due valori di D reali. Al contrario il salto verticale presente nel grafico della capacità è molto evidente, coerentemente con quanto atteso.
Si può notare come nel punto 1622 (1538+x+20+(46-20-x)+38), ovvero la situazione presente con due o più frammenti e il padding, ci siano più valori validi. Questo è proprio dovuto al padding layer 2, che riempie il pacchetto in modo da arrivare ai 46 bytes necessari per avere la dimensione minima del payload ethernet. Nella foto 1.2bis è stato riportato uno zoom attorno alla discontinuità.
Questi ultimi due fenomeni sono presenti anche nei successivi grafici.

2 - Capacità 100 Mb/s

I grafici 2.1 e 2.2 rappresentano rispettivamente i dati raccolti durante i test svolti in laboratorio e a casa per quanto riguarda il collegamento diretto degli host con capacità di 100Mb/s.
In questo caso notiamo come nel grafico riferito al RTT ci sia molta più variazione tra minimo e massimo ed in generale sono presenti molte più oscillazioni durante le misurazioni: questo è dovuto ad una maggiore velocità del canale (il che contribuisce a rendere più evidenti anche ritardi precedentemente trascurabili) e ad una bassa precisione di misurazione del RTT fornita da ping.
Anche in questo caso si trova un ritardo superiore a quello atteso: infatti, per D=1538 il RTT dovrebbe essere pari a 0.246ms, mentre nella realtà è pari a circa 0.5-0.6 ms.Tra i 1500 ed i 2000 bytes si nota un andamento costante non preventivato, probabilmente dovuto ad una ottimizzazione non identificata svolta dal setup su cui si sta lavorando, e subito dopo un'oscillazione molto alta sui ritardi, arrivando circa a ciò che ci si aspetterebbe mantenendo un andamento lineare. Inoltre, si nota come anche nel migliore dei casi (il grafico 2.2) la capacità ci metta molto più tempo a raggiungere la saturazione attorno ai 100 Mb/s attesi; questo è dovuto al fatto che per dimensioni di D piccole, a causa della velocità aumentata, il tempo di elaborazione non è trascurabile (portando ad avere una capacità minore di quanto atteso), mentre per D crescenti il tempo di elaborazione diventa sempre più trascurabile e si converge quindi alla velocità teorica.

3 - Capacità 1000 Mb/s

I grafici 3.1 e 3.2 illustrano i risultati ottenuti nelle misurazioni svolte, rispettivamente in laboratorio e a casa, tramite collegamento diretto degli host alla velocità di 1000 Mb/s.
La misura effettuata nel laboratorio presenta notevole rumore e variazioni tra minimo e massimo. Il test è stato eseguito nuovamente a casa per verificare l'andamento, ottenendo un risultato molto simile (visibile nella foto 3.2). Si può quindi supporre che la bassa precisione di ping non ci permetta di avere una misura accurata a velocità elevate. Il risultato ottenuto, inoltre, supporta l'ipotesi che il ritardo presente in tutti gli esperimenti venga inserito a monte (quindi dettato dalle schede di rete e dallo strumento ping) e non è quindi legato al setup impiegato (in questo caso per D=1500 ci si aspetterebbe 0.025 ms mentre qui si nota un RTT pari a 0.2 ms). In questo caso il ritardo è ulteriormente amplificato dalla non trascurabilità di η: questo aspetto è confermato anche dal risultato ottenuto a casa, che, nonostante nei punti 1,2 risultasse molto più preciso e molto simile all'andamento teorico atteso, qui non solo presenta un andamento rumoroso ma non riesce neanche a convergere alla velocità teorica, fermandosi a circa 300 Mb/s. Probabilmente, visto che nel grafico 2.2 si è quasi raggiunta la velocità teorica per D=10.000 B si sarebbe potuto ottenere lo stesso risultato se il ping fosse arrivato a D=100.000 B: ciò è stato formulato in seguito all'osservazione dell'andamento del grafico 2.2 tra 0 B e 1000 B, che è molto simile a quello del 3.2 tra 0 B e 10.000 B.

Collegamento tramite switch

4 - Capacità 10 Mb/s

Il grafico 4.1 rappresenta il test effettuato in laboratorio alla velocità di canale di 10 Mb/s.
L'andamento risulta simile a quello in assenza di switch, presentando però un RTT leggermente superiore presumibilmente dovuto al tempo necessario allo switch per effettuare lo store & forward dei pacchetti.
Anche in questo caso il RTT risulta superiore rispetto alla stima prevista: infatti, usando D=1538 dovrebbe essere pari a circa 5 ms, mentre nella realtà risulta pari a circa 8 ms.
Il grafico 4.2 deriva dalle verifiche effettuate a casa, sempre tramite l'utilizzo di uno switch, ma valutando valori di D fino a 10000 B. Tramite questo ulteriore test è stato possibile apprezzare l'andamento asintotico della capacità del canale, il quale tende a raggiungere i 10 Mb/s effettivi nonostante la decrescita data dalle frammentazioni. La principale differenza tra i 2 grafici è data dalla pendenza della crescita della capacità (e quindi del RTT): infatti, il grafico interpolato con i dati ottenuti a casa raggiunge la capacità del canale in prossimità della prima frammentazione e tende ad essa asintoticamente, mentre il grafico ottenuto in laboratorio non arriva alla velocità teorica neanche subito prima della seconda frammentazione a causa dei probabili ritardi introdotti già analizzati nel punto 1.

5 - Capacità 100 Mb/s

La figura 5.1 mostra l'andamento valutato in laboratorio per quanto riguarda la velocità di 100 Mb/s. Come è possibile apprezzare dal grafico, l'andamento del RTT resta qualitativamente costante tra i 1500 B e i 2000 B di dimensione dei dati inviati, per poi presentare un aumento quasi brusco del tempo impiegato a circa 2300 B. Questo comporta inoltre una diminuzione della capacità del canale nell'intorno dello stesso punto.
L'andamento qualitativo assomiglia comunque a quello atteso dalla teoria, ma risulta ovviamente sbagliato in quanto oltre al salto atteso attorno a 1600 B ne è presente un altro traslato di circa 800 B a destra. Questo comportamento è probabilmente dovuto sia alle alte velocità usate sia a eventuali ottimizzazioni o comportamenti interni della scheda di rete e/o dei tool usati.
La figura 5.2, come nel caso precedente, mostra l'esperienza sviluppata a casa con una dimensione dei dati fino a 10000 B. Le considerazioni restano le stesse del test svolto in laboratorio, mostrando in più l'andamento della capacità utilizzata del canale per valori più elevati di D, permettendo così di apprezzarne l'andamento asintotico. In generale l'andamento risulta molto meno rumoroso del test effettuato in classe e non presenta il salto intorno ai 2300 B precedentemente notato. Questo lo rende più coerente con quello teorico atteso, risultando in particolare qualitativamente simile al caso 2.2.

6 - Capacità 1000 Mb/s (Esperienza aggiuntiva)

Questo esperimento è stato svolto in seguito alla disponibilità hardware a casa, per coprire tutte le possibili casistiche con e senza switch.

Nel grafico 6 vengono visualizzate le solite informazioni per quanto riguarda una capacità di canale pari a 1000 Mb/s, in presenza di uno switch, per valori di D fino a 10000 byte.
L'andamento del RTT mantiene un andamento qualitativamente simile a quello dei casi precedenti pur presentando valori molto variabili del RTT massimo e minimo.
La capacità utilizzata del canale, d'altro canto, non raggiunge, nel caso valutato, il punto in cui il comportamento sarebbe dovuto essere asintotico. Questo è dovuto all'elevata capacità del canale considerato e ai valori elevati di η che non ne permettono una crescita rapida, in quanto in questo caso il tempo di elaborazione non risulta trascurabile.
Le considerazioni che si possono fare sono esattamente analoghe a quelle presenti nel punto 3.2: per poter avere dei tempi di elaborazione trascurabili e, di conseguenza, raggiungere una convergenza di C a 1000 Mb/s, sarebbe servito avere una dimensione di D superiore a 100.000 B.

Appendice

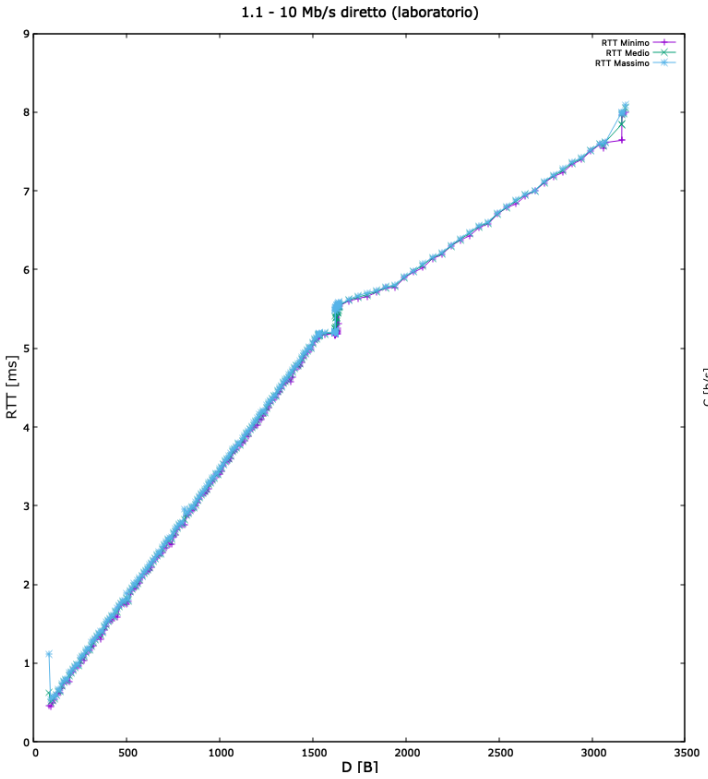
0.1:

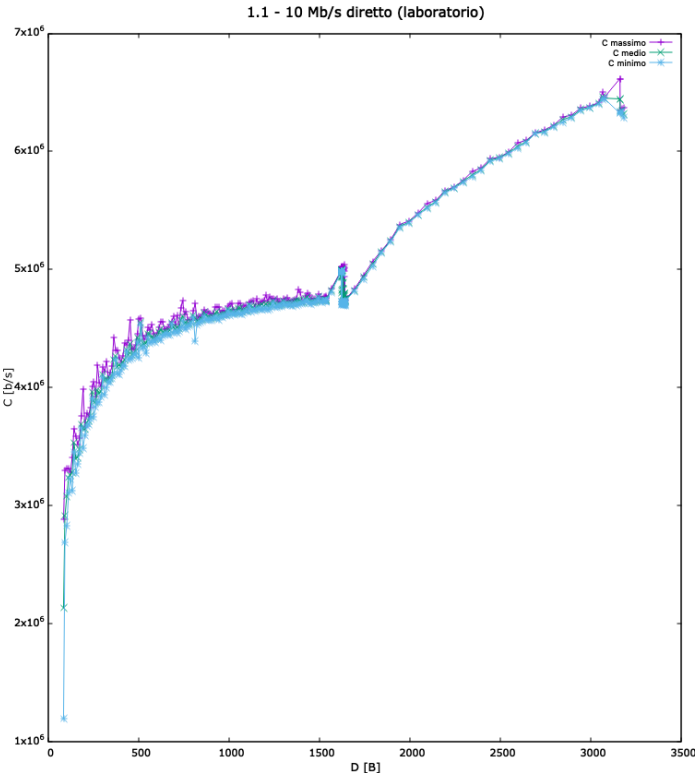
```
ip="172.16.22.1"
nome="tempi10fino10000"
for i in $(seq 16 10 1466)
do
        echo -n "$i "
        ping $ip -i 0,4 -s $i -c 5 | egrep -e "rtt" | tr -s " " | cut -d "=" -f 2,2 | cut -d "/" -f 1-3 | tr "/" " "
done>$nome.txt
for i in $(seq 1467 1 1550)
do
        echo -n "$i "
        ping $ip -i 0,4 -s $i -c 5 | egrep -e "rtt" | tr -s " " | cut -d "=" -f 2,2 | cut -d "/" -f 1-3 | tr "/" " "
done>>$nome.txt
for i in $(seq 1551 50 2939)
do
        echo -n "$i "
        ping $ip -i 0,4 -s $i -c 5 | egrep -e "rtt" | tr -s " " | cut -d "=" -f 2,2 | cut -d "/" -f 1-3 | tr "/" " "
done>>$nome.txt
for i in $(seq 2939 10 3000)
do
        echo -n "$i "
        ping $ip -i 0,4 -s $i -c 5 | egrep -e "rtt" | tr -s " " | cut -d "=" -f 2,2 | cut -d "/" -f 1-3 | tr "/" " "
done>>$nome.txt
for i in $(seq 3001 20 10000)
do
        echo -n "$i "
        ping $ip -i 0,4 -s $i -c 5 | egrep -e "rtt" | tr -s " " | cut -d "=" -f 2,2 | cut -d "/" -f 1-3 | tr "/" " "
done>>$nome.txt
```

0.2:

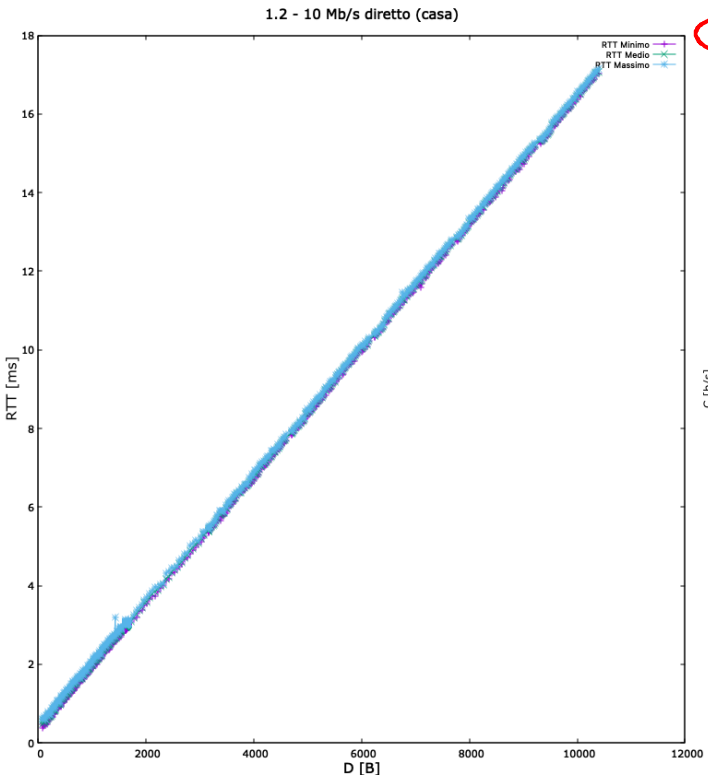
```
set multiplot layout 1,2 rowsfirst

set title "1.2 - 10 Mb/s diretto (casa)" font "Verdana,15"
show title

set xtics font "Verdana,12"
set ytics font "Verdana,12"

file="tempi10fino10000.txt"
switch=0

ip_eth=20+38
byte_framm(s)=int(s+8)%1480+20

D(s)=(s+8+ip_eth)+(ip_eth)*(floor((s+8-1)/1480))+(byte_framm(s)<46?46-byte_framm(s):0)

D_sw(s)=(switch?(s>1472?(D(s)+1538)*2:D(s)*4):D(s)*2)

set xlabel "D [B]" font "Verdana,15"
set ylabel "RTT [ms]" font "Verdana,15"

plot file using (D($1)):2 title "RTT Minimo" with linespoint, file using (D($1)):3 title "RTT Medio" with linespoint,
file using (D($1)):4 title "RTT Massimo" with linespoint

set xlabel "D [B]" font "Verdana,15"
set ylabel "C [b/s]" font "Verdana,15"

plot file using (D($1)):(D_sw($1)*8*1000/$2) title "C massimo" with linespoint, file using (D($1)):(D_sw($1)*8*1000/$3) title "C medio" with linespoint,
file using (D($1)):(D_sw($1)*8*1000/$4) title "C minimo" with linespoint
```

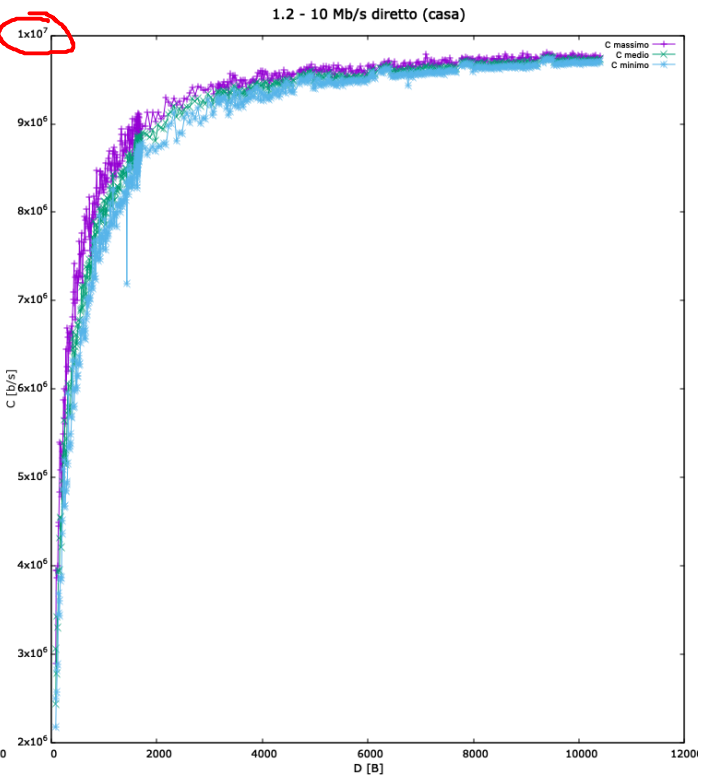
1.1: non satura



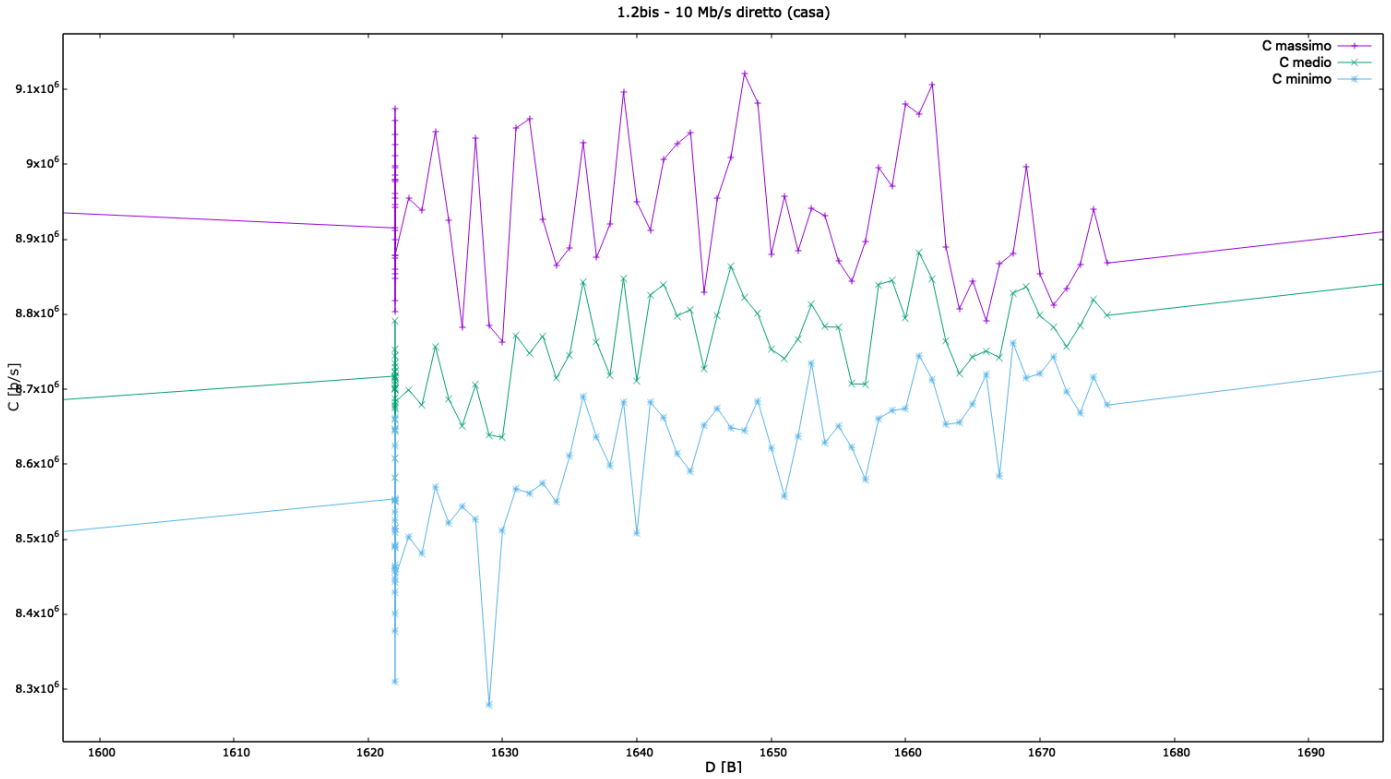


1.2:

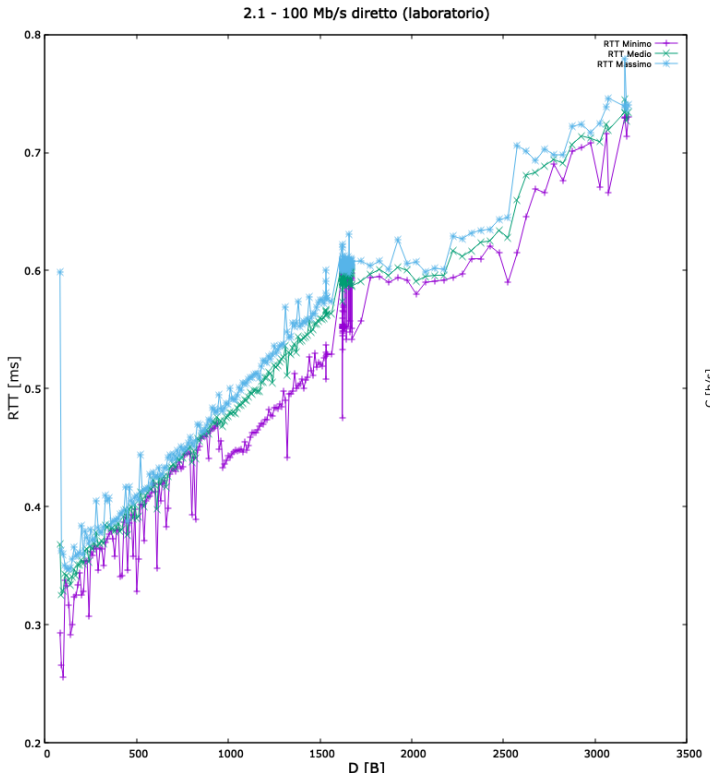


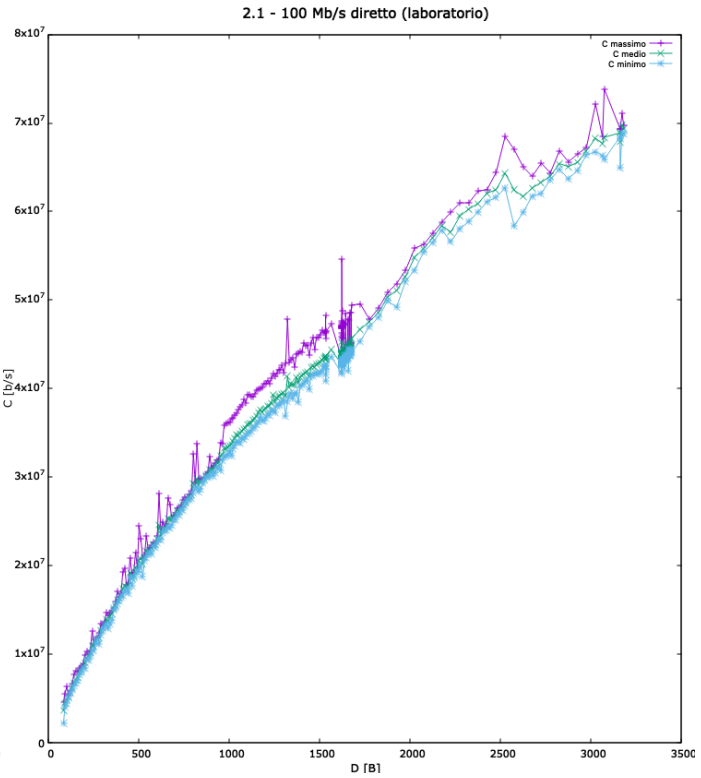


1.2bis:

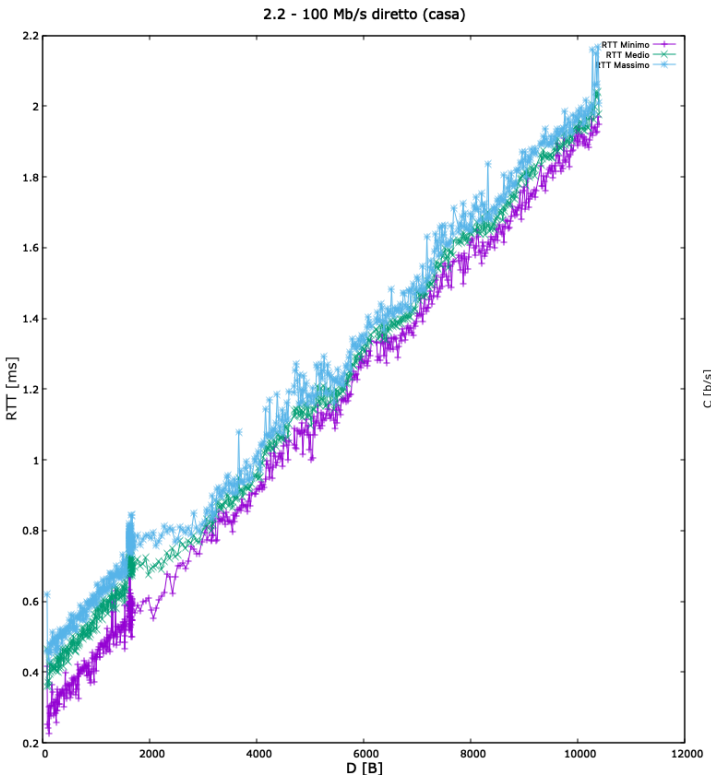

1.2bis - 10 Mb/s diretto (casa)
C massimo
C medio
C minimo
C [b/s]
D [B]

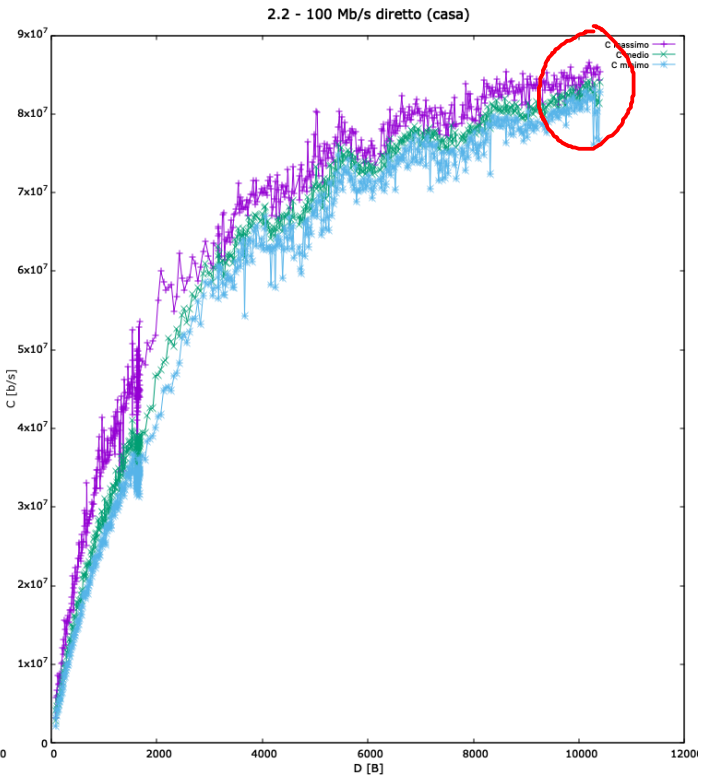
2.1:


2.1 - 100 Mb/s diretto (laboratorio)
RTT Minimo
RTT Medio
RTT Massimo
RTT [ms]
D [B]

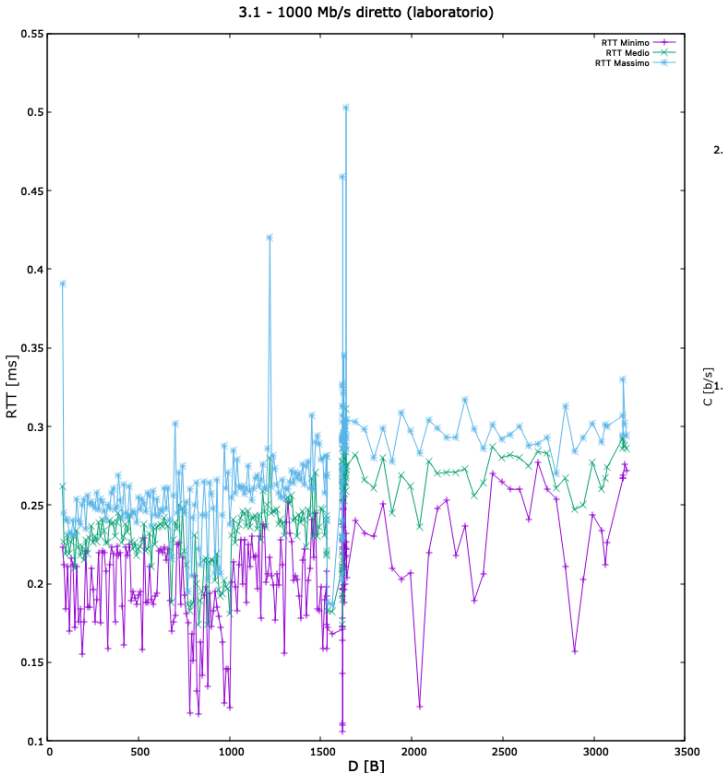

2.1 - 100 Mb/s diretto (laboratorio)
C massimo
C medio
C minimo
C [b/s]
D [B]

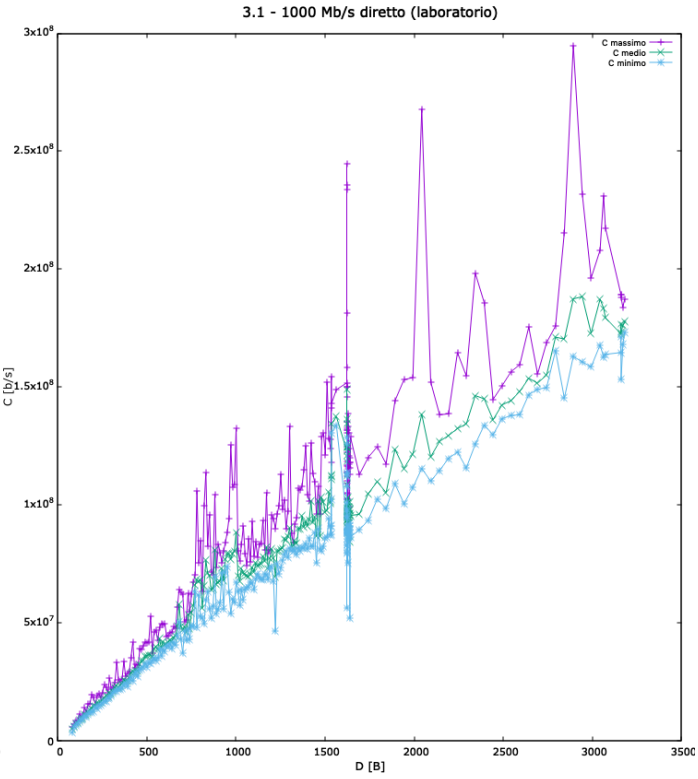
2.2:


2.2 - 100 Mb/s diretto (casa)
RTT Minimo
RTT Medio
RTT Massimo
RTT [ms]
D [B]

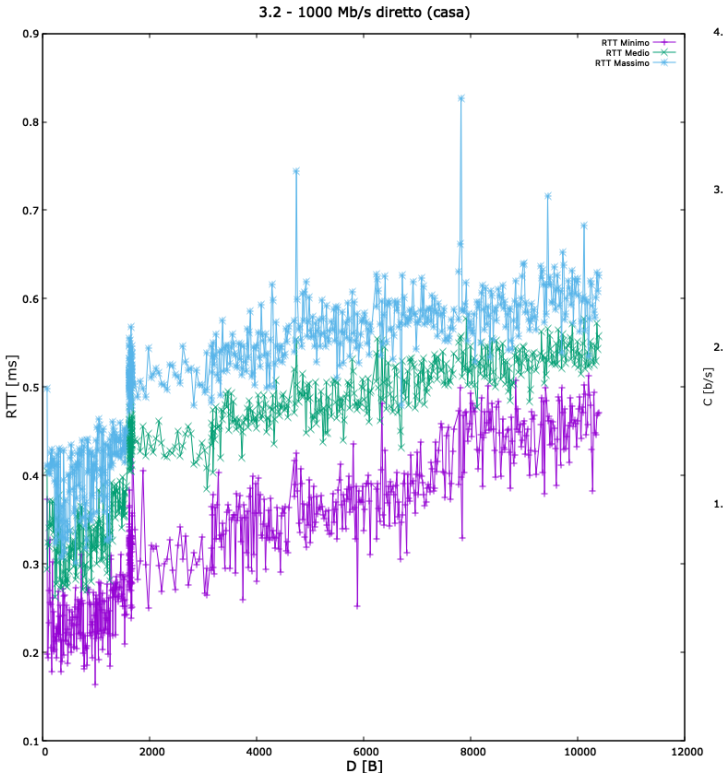

2.2 - 100 Mb/s diretto (casa)
C massimo
C medio
C minimo
C [b/s]
D [B]

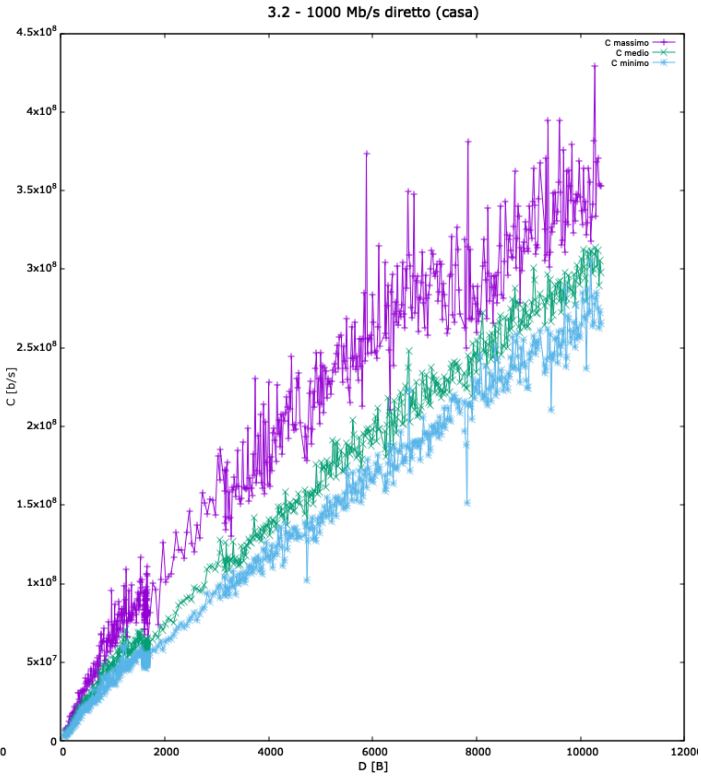
3.1:


3.1 - 1000 Mb/s diretto (laboratorio)
RTT Minimo
RTT Medio
RTT Massimo
RTT [ms]
D [B]

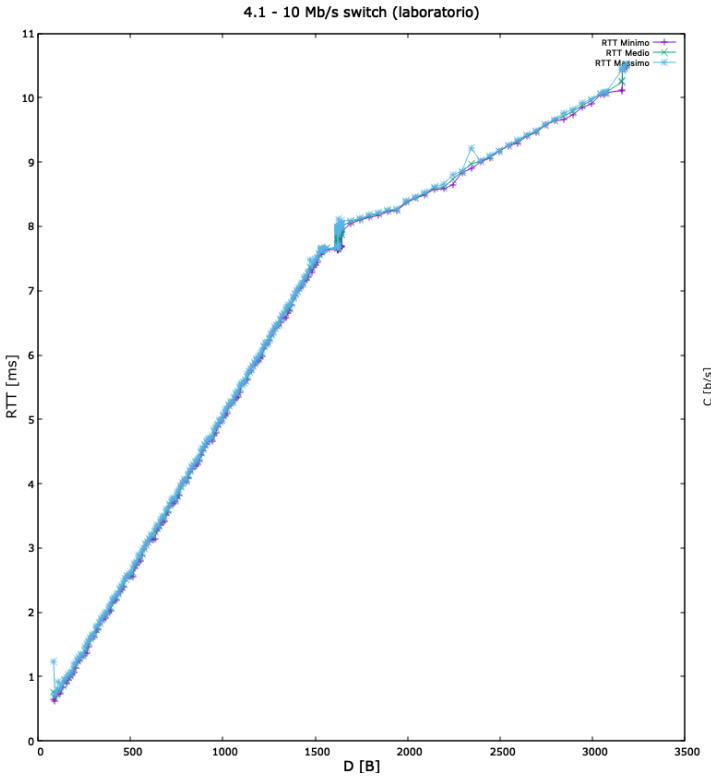

3.1 - 1000 Mb/s diretto (laboratorio)
C massimo
C medio
C minimo
C [b/s]
D [B]

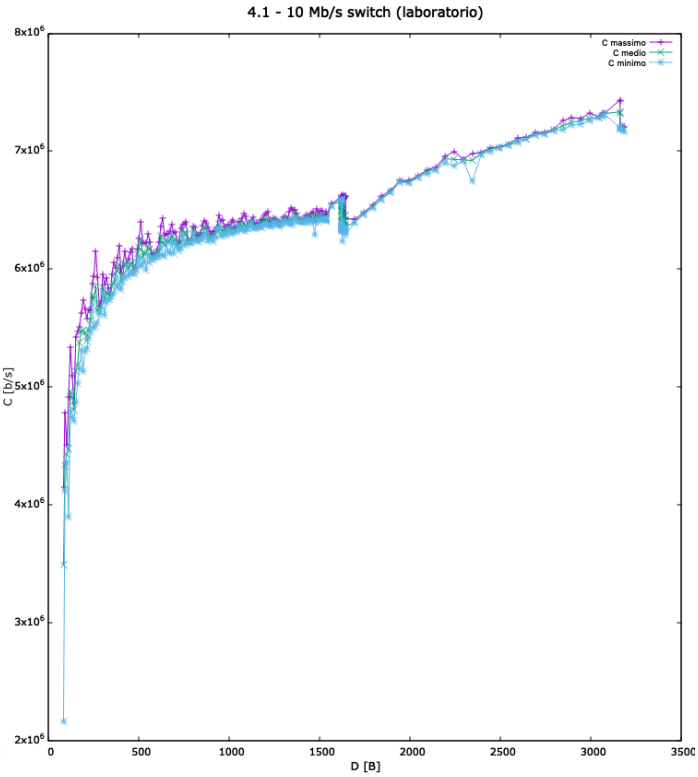
3.2:


3.2 - 1000 Mb/s diretto (casa)
RTT Minimo
RTT Medio
RTT Massimo
RTT [ms]
D [B]

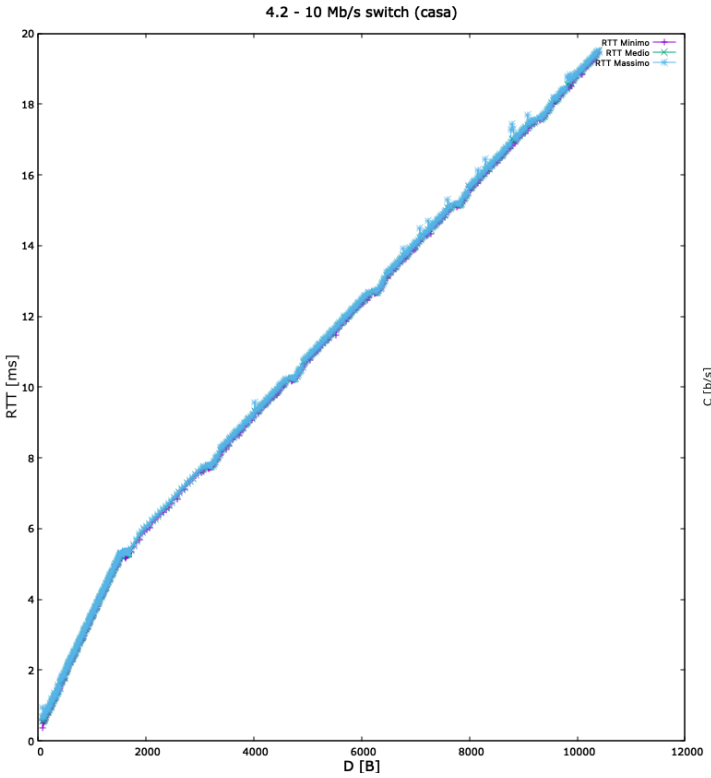

3.2 - 1000 Mb/s diretto (casa)
C massimo
C medio
C minimo
C [b/s]
D [B]

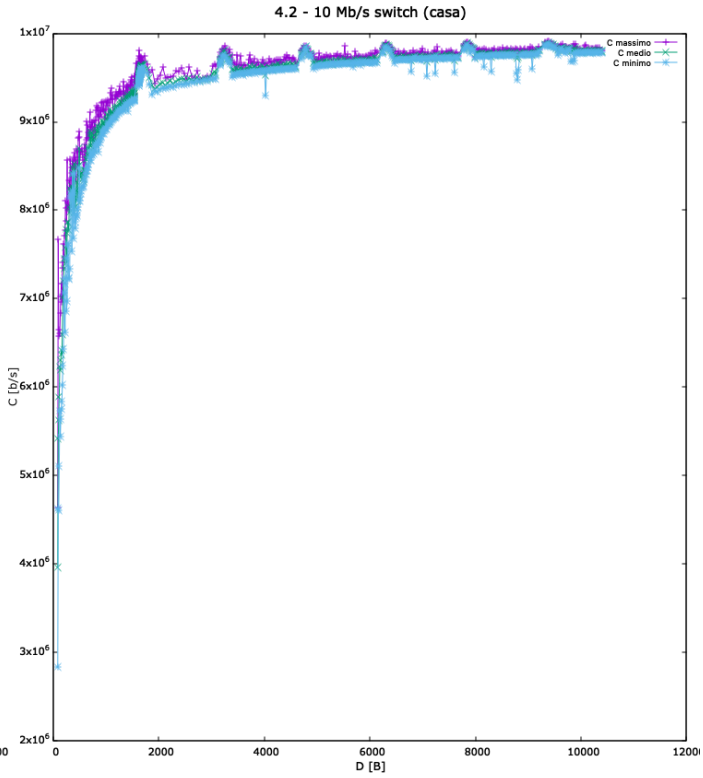
4.1:


4.1 - 10 Mb/s switch (laboratorio)
RTT Minimo
RTT Medio
RTT Massimo
RTT [ms]
D [B]

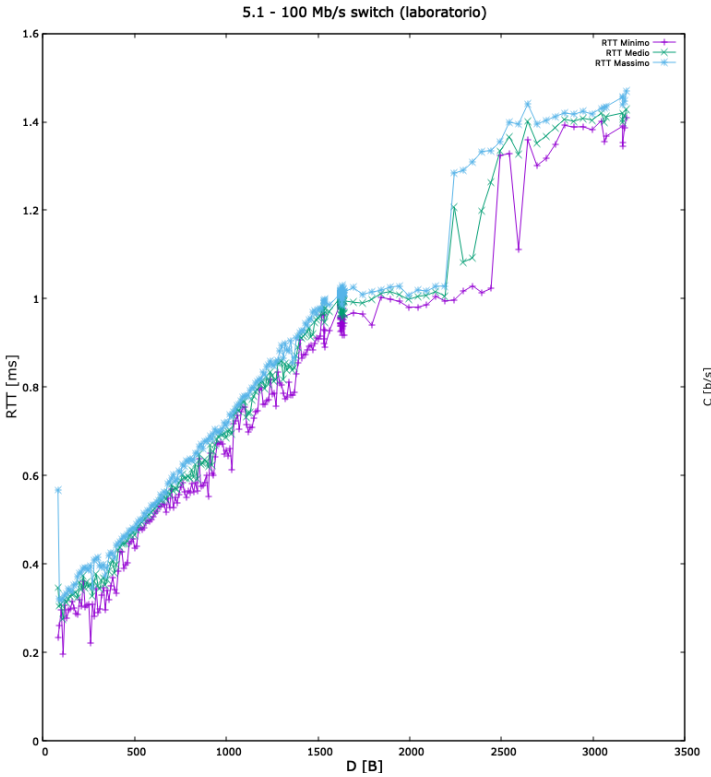

4.1 - 10 Mb/s switch (laboratorio)
C massimo
C medio
C minimo
C [b/s]
D [B]

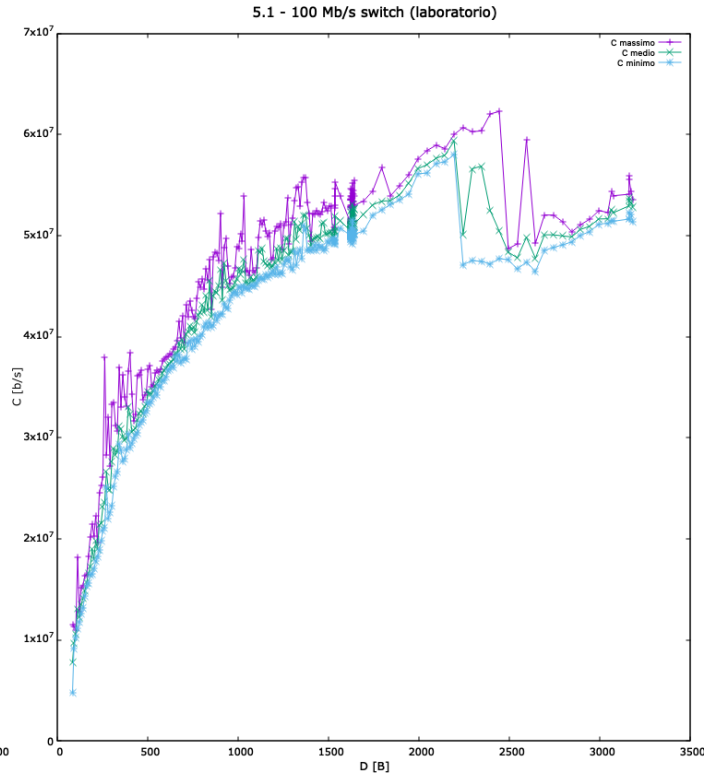
4.2:


4.2 - 10 Mb/s switch (casa)
RTT Minimo
RTT Medio
RTT Massimo
RTT [ms]
D [B]

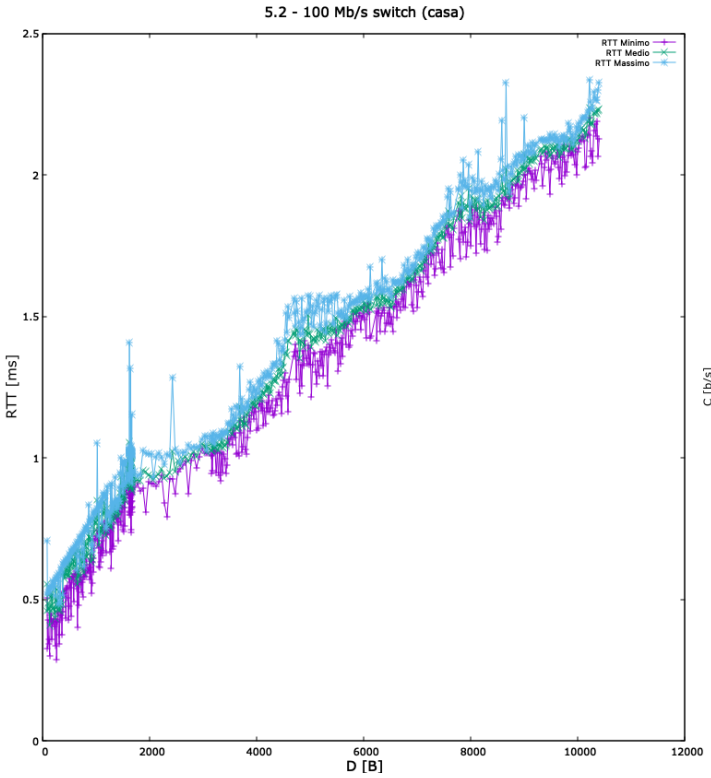

4.2 - 10 Mb/s switch (casa)
C massimo
C medio
C minimo
C [b/s]
D [B]

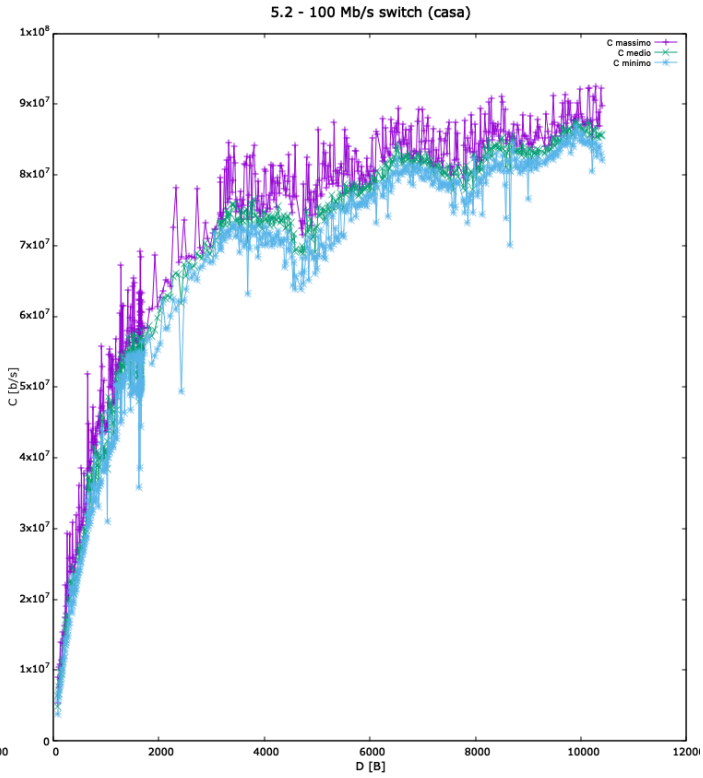
5.1:


5.1 - 100 Mb/s switch (laboratorio)
RTT Minimo
RTT Medio
RTT Massimo
RTT [ms]
D [B]

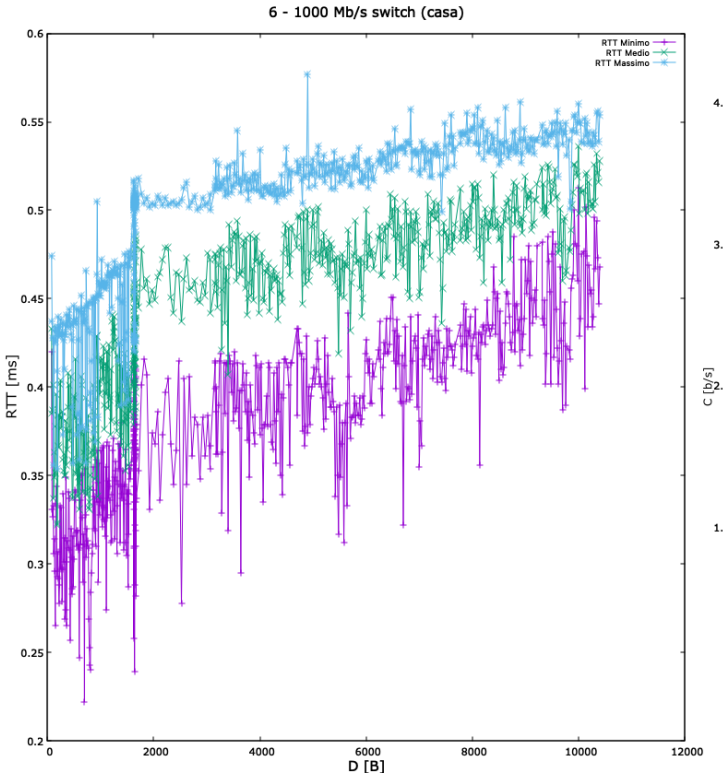

5.1 - 100 Mb/s switch (laboratorio)
C massimo
C medio
C minimo
C [b/s]
D [B]

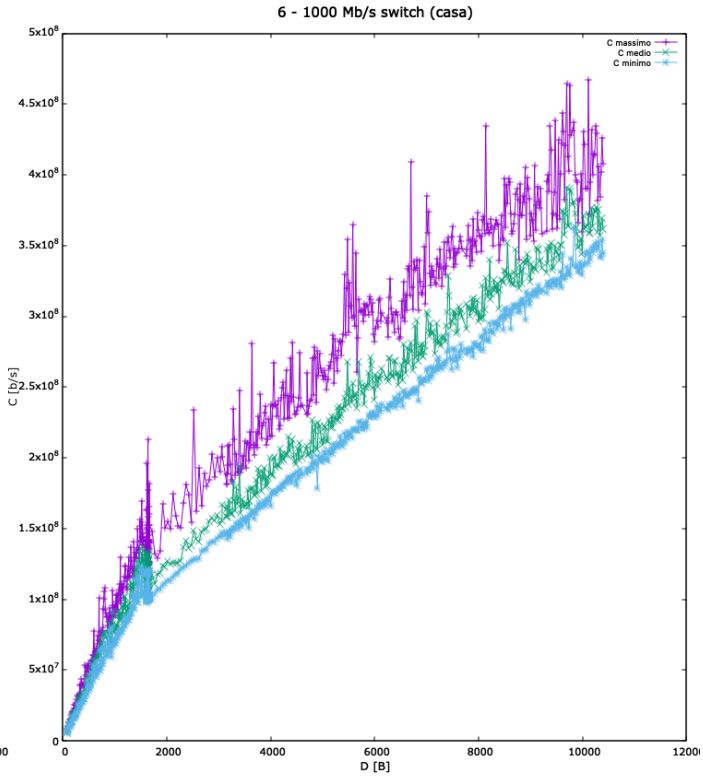
5.2:





6:





Completa e ben fatta

10/10