

TELECOMUNICAZIONI

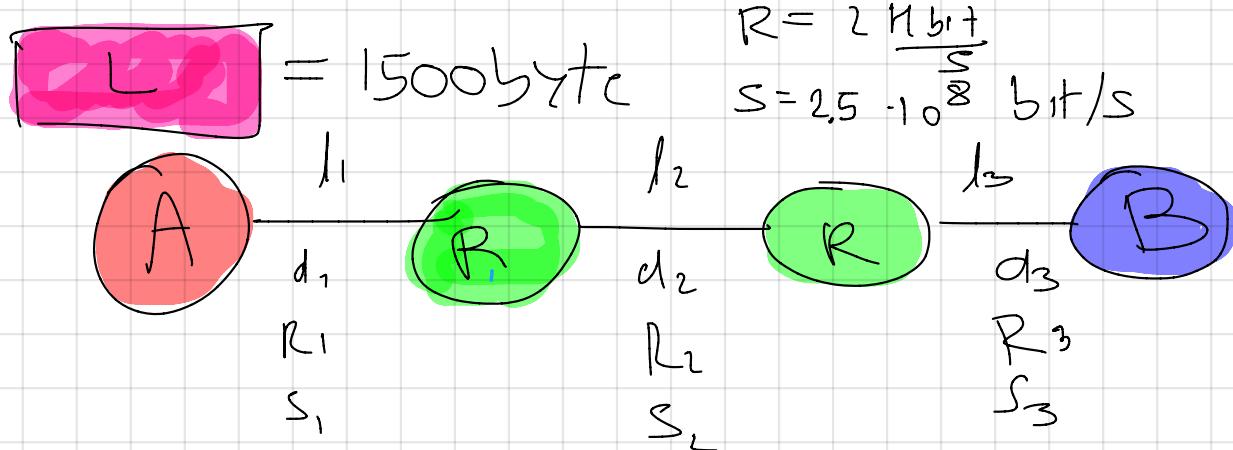
By Edoardo

RITARDI RETI PACKET SWITCHING (PS)

$$D = d_{elab} + d_{queue} + d_{transm} + d_{prop}$$

- d_{elab} = ritardo di elaborazione
- d_{queue} = ritardo di accodamento
- d_{transm} = ritardo di trasmissione = $\frac{L}{R} = \frac{\text{lunghezza pacchetto (bit)}}{\text{frequenza (bps)}}$
- d_{prop} = ritardo di propagazione = $\frac{d}{S} = \frac{\text{lung. fisica collegamento (m)}}{\text{velocità propagazione } (2 \cdot 10^8 \text{ m/s})}$

Pag 70 es 8.10 (Testo)



$$D_{\text{TOT}} = 3 \cdot d_{transm} + \frac{d_1}{S} + \frac{d_2}{S} + \frac{d_3}{S} + 2 \cdot d_{elab}$$

$$= 3 \cdot \frac{12000 \text{ b}}{2 \cdot 10^6 \text{ bps}} + \frac{5000 \text{ m}}{2 \cdot 10^8 \text{ m/s}} + \frac{4000 \text{ m}}{2 \cdot 10^8 \text{ m/s}} + \frac{1000 \text{ m}}{2 \cdot 10^8 \text{ m/s}} + 2 \cdot 3 \text{ ms} =$$

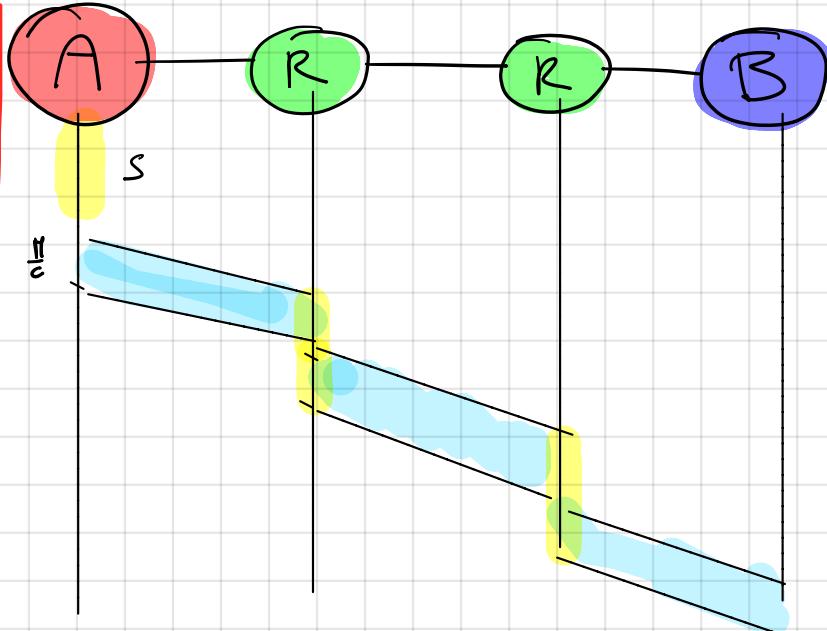
$$0.018 + 0.00025 + 0.0002 + 0.000005 + 0.006 = \sim 24 \text{ ms}$$

ES. 2

$$M = 560 \text{ bit}$$

$$\text{Protez. G2121ms} = 10 \text{ ms}$$

$$64 \text{ Kbit/s}$$

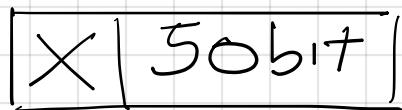


$$D = S + \frac{M}{c} + 10 \text{ ms} \cdot 3 + 0,5 \text{ ms}$$

Versione 2 Pacchetti

$$X = 10 \text{ bit}$$

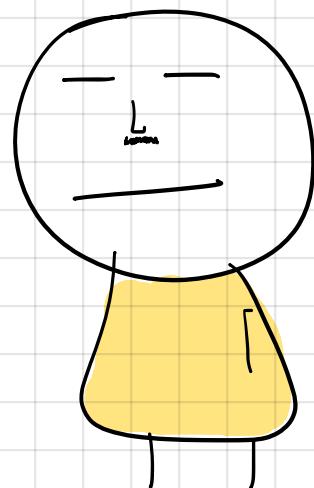
$$l = \max 50 \text{ bit}$$



$$N = \frac{M}{l} = \frac{560}{50} = 12 \text{ pacchetti}$$

Tempo orario 1° nodo
di 1 pacchetto

$$= \frac{60 \text{ bit}}{64 \text{ Kbit/s}}$$



A pena il primo pacchetto arriva al primo nodo, parte il secondo pacchetto

$$D = 3 \cdot 10 \text{ ms} + 3 \cdot \frac{60 \text{ bit}}{64 \text{ Kbit/s}} + 10 \cdot \frac{60}{64} + \frac{20}{60}$$

Pag 29 es.10 (Esercizio)

3) $L_{UOP} = 400 \text{ bit}$

$H_{UOP} = 25 \text{ bit}$

2) $L_{IP} = 420 \text{ bit}$

$H_{IP} = 40 \text{ bit}$

1) $L_C = 250 \text{ bit}$

$H_2 = 10 \text{ bit}$

$M = 3800 \text{ bit}$

IT = intervallo Temporale



TRONCO (Percorso in 10 ms)

$$C = \frac{260 \text{ bit}}{50 \text{ ms}} \quad \begin{array}{l} \text{conciati} \\ \text{sottoconcisi} \end{array}$$

1)



$$N (\# \text{ unità informative}) = \lceil \frac{3800}{400} \rceil = 10$$

a) $9N \rightarrow 420 + 1N \rightarrow 220$

b) $9N \rightarrow 460 + 1N \rightarrow 260$

c) $\lceil \frac{460}{250} \rceil \cdot 9 = 18 + \lceil \frac{260}{250} \rceil = 2 = 20$

N.B. I pacchetti da 460 non entrano nei 250
 $\Rightarrow \boxed{\lceil \frac{460}{250} \rceil} \rightarrow \frac{460}{2} = 230 + 10$

Vengono generati 2 pacchetti da 210 da ogni pacchetto da 420 e ad ognuno vengono aggiunti i 10 bit di H_{IP} . Poi dovranno costituire "pacchi" da 250:
 $\Rightarrow 230 + 10 = 260$ (18)

Il pacchetto da 260 genera un pacchetto da 250 (10) e ovviamente 10 bit a cui si aggiungono 10 bit di H_{UOP} e 200 di riempimento.

Ogni IT è utilizzata per trasportare un pacchetto da 260bit, dato che 1 IT per tronco è utilizzato per il trasferimento di informazioni di replicazione, dovranno necessarie 5 tronche per il trasferimento del messaggio.

2) Percentuale d'extra informazione:

$$20 \cdot 10 = 200 \text{ bit} +$$

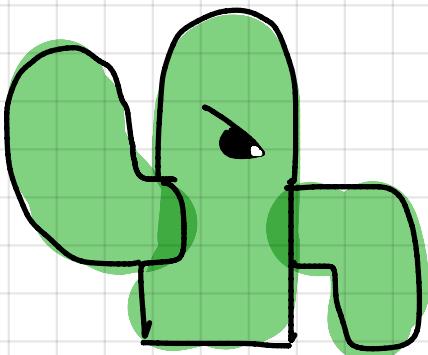
$$40 \cdot 10 = 400 \text{ bit} +$$

$$10 \cdot 20 = 200 \text{ bit} +$$

$$260 \cdot 5 = 1300 \text{ bit (Tronche)} = 2700$$

$$\text{Percentuale} = \frac{2700}{3800 + 2700} = 0,415$$

3) Il tempo minimo richiesto è dato dal tempo minimo necessario per percorrere le 5 tronche ovvero $5 \cdot 10 \text{ ms} = 50 \text{ ms}$



Pag 74 es P31 (Testo)

$$M = 8 \cdot 10^6 \text{ dimension pacchetto} \Rightarrow$$

(21) $\frac{8000000}{2000000} = 4 \text{ s}$

(22) $4 \cdot 3 = 12 \text{ s}$

(b) $\frac{10000}{2000000} = \frac{1}{200} = 5 \text{ ms} \quad (\text{Primo pacchetto})$

Il secondo pacchetto impiega 10 ms per essere ricevuto dal secondo commutatore

(c) $T = \frac{3P}{R} + 789 \cdot \frac{P}{R}$

ARCHITETTURA A STRATI

Strato Fisico

- Trasferisce bit

Si occupa del trasferimento dei bit sui mezzi di trasmissione

Strato Collegamento

- Trasferisce frame

Prende i bit dallo strato fisico, gli incapsula nei frame e controlla e corregge gli errori

Strato Rete

- Trasferisce pacchetti

Si occupa del trasferimento e intitolamento dei pacchetti e controllo gestione

Strato Transporto

- Trasferisce Segmenti

Segmenta e riassumba i pacchetti e gestisce i "port numbers"

Connection Setup, maintenance e release

Strato Applicazione

Ri elabora il messaggio ricevuto

Lo strato n fornisce servizi allo strato n+1.

I tratti dello stesso livello si combinano entità denominate PDU

Strati di livello diverso si combinano SDU = PDU + header. Tale scombinio è permesso da Service Access Point (SAP)

In caso uno strato abbia un limite di dimensione al dato emittente le PDUs vengono scomposte e riassummate al dato ricevente

Servizio CON connessione

I due terminali instaurano la connessione, stabilisce le regole di trasferimento, conferma ricezione e chiude la connessione

Servizio SEZIONE connessione

Nessuna connessione iniziale, ma solo trasferimento senza controllo di ricezione.

STRATO FISICO

- Informazione a blocchi: strutturata in blocchi indipendenti con un numero di bit fisica per blocco (size)
- Informazioni Stream: informazione prodotta e trasmessa in modo continuo
BIT RATE: quantità di bit prodotti in unità di tempo
Può essere Costante (CBR) o Variabile (VBR)
= frequenza bit / s

- Delay di trasferimento (blocchi)

$$- \text{Minc} = T_{\text{prop}} + \frac{L}{R} = \frac{d}{c} + \frac{L}{R}$$

Il ritardo può essere diminuito tramite tecniche di compressione (diminuzione di L) o di trasmissione (aumento di R)

Senza Perdita \rightarrow originale == inviata (compresso)

Con Perdita \rightarrow originale != inviata (compresso)

- Rapporto di compressione = $\frac{\text{Byte originale}}{\text{Byte compresso}}$ $R > 1$ per non perdere info

ES. trasmissione immagine $H \times W$ pixel

$$\begin{aligned} \text{Borigine} &= 3 \times H \times W \times B \text{ bit/pixel} = 3HWB \text{ bit} \\ G + \boxed{B} &= \boxed{R} + \boxed{G} + \boxed{B} \end{aligned}$$

Se informazioni di tipo stream vorremo DIGITALIZZARE tramite SAMPLING e CODIFICA: ogni istante prendo un campione del segnale diviso in livelli approssimandolo al più vicino; ogni livello ho un certo numero di bit

Il numero di bit dipende dal numero di livelli in cui divido il segnale ($3 \text{ bit} = 2^3 = 8 \text{ livelli}$)

$B_s = \text{bit/campione} * \text{campione/sec}$ e da esso dipenderà la qualità del segnale

Teorema del campionamento

Se campiono ogni $T_c = \frac{1}{F_c}$ dove $F_c = \text{frequenza di campionamento} = 2 \times W_s$
Se $F_c < 2 \times W_s$ non producirà una riproduzione fedele

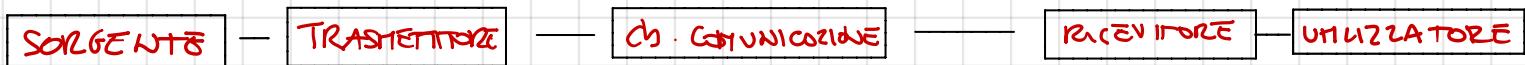
LARGHEZZA DI BANDA (BANDWIDTH) WS (Hz) indica quanto velocemente varia nel tempo il segnale, maggiore è la banda più frequenti sono i campioni

- Frequenza minima campionamento $F_c = 2 \times W_s$

Più la banda è grande più frequentemente dovrà compionere per preservare le informazioni.

Tempo di campionamento = $T_c = 1/F_c$ (ogni quanto compiona)

Schemi Sistemi Trasmissione



Trasmettore: converte il segnale dalla sorgente adattandolo al canale di comunicazione

Ricevitore: riceve il segnale dal canale e lo converte per l'utente finale

Tipologia Stream

Constant bit rate in cui la rete deve fornire un canale con banda almeno uguale al bit rate della sorgente

Variable bit rate in cui la rete deve supportare la variabilità del bit rate in modo efficiente

Nel constant bit rate si utilizza il coefficiente di attività per indicare quanto uno utente usa un canale ($\lambda = 0,2 \Rightarrow$ uso solo il 20% del canale)
Questo è detta MULITPLAZIONE STATISTICA

SEGNALE ANALOGICO: il ricevente deve ricostruire tutti i dettagli con elevata possibilità di errore, distorsione limitata

SEGNALE DIGITALE: il ricevente deve ricostruire solo i livelli del segnale permettendo comunicazioni a lungo distanza.

Segnale Numerico Binario

$$\text{bit rate} = 1 \text{ bit} / T \text{ sec}$$

Codifica: alle ordinate triono le ampiezze.

Per aumentare il bit rate si può diminuire il tempo es. $2T \text{ sec}$, ma ciò può portare ad errori:

- Interferenza Intersimbolica: due code di simboli possono interferire tra loro sommarsi creando curve diverse dall'originale.
- L'intervallo di decisione, cioè l'intento in cui viene valutato il segnale, può non avvenire perfettamente al centro di T e perciò l'overlapping del simbolo oltre lo spazio d'errore.

La frequenza massima degli impulsi per non avere interferenze tra loro è $F = 2 \times W_c$ dove W_c è la banda del canale (Frequenza di Nyquist)

Se la trasmissione è multilivello il bit rate R è dato da:

$$\text{bit rate} = m \text{ bit/impulso} \times 2 W_c \text{ impulsi/sec} = 2 m W_c \text{ bit/s}$$

m è dato dal numero di bit che ogni livello rappresenta

Ese. se l'ampiezza degli impulsi può assumere valori appartenenti all'insieme $\{-A, -A/3, A/3, A\}$ ogni impulso può rappresentare 2 bit \Rightarrow ampiezza $H \Rightarrow m = H/2$

Troppi livelli possono interfare tra loro generando rumore se sono a pochi distanze tra loro gli impulsi

$$\text{BER (bit error rate)} = \# \text{ bit giusti} / \# \text{ bit complessivi}$$

$$\text{SNR (signal to noise ratio)} = \text{rapporto segnale rumore} = \text{potenza media segnale} / \text{potenza media rumore}$$

$$\text{Per esprimere in dB} \rightarrow 10 \log (\text{SNR})$$

LIMITE DI SHANNON: esprime la massima velocità del canale

$$C = W_c \log_2 (1 + \text{SNR}) \text{ bit/s}$$

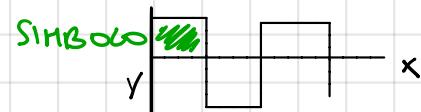
Se non c'è interferenza intersimbolica e non c'è rumore il massimo rate di trasmissione è $R_{\max} = 2W_c$

$$S(t) = \sin(2\pi W_c t) / 2\pi W_c t$$

FILTRO PASSO-BASSO: le frequenze $F > W_c$ sono bloccate. Le altre possono ma sono ritardate.

La larghezza di banda è l'ampiezza di tutte le frequenze che possono attraversare l'intervallo $0 - W_c$

FILTRO PASSA-BANDA: bloccano le basse e alte frequenze.



Formula Strato Fisico

$$\text{Rapporto Compressione} = \frac{\text{bit originali}}{\text{bit compressi}}$$

$$\text{Bit Rate} = \# \text{ bit ad ogni campione} \times \underbrace{\# \text{ campioni per secondo}}$$

$$\text{Frequenza Campionamento minima: } F_c = 2 \times W_s \quad W_s = \text{bandwidth}$$

$$\text{Rate Segnale Video} = M \text{ bit/pixel} \times (w \times h) \text{ pixel/frame} \times F \text{ frame/sec}$$

$$\alpha = \text{coefficiente attivit\~a} \leq 1 = \frac{R_{\text{medio}}}{R_{\text{picco}}}$$

nel caso ON/OFF $\alpha = \frac{T_{\text{on}}}{T_{\text{on}} + T_{\text{off}}}$

$$\text{Grado di Intermittenza} = G = \frac{1}{\alpha}$$

Trasloca banda: moltiplicare lo banda base $\times 2$. Altrimenti se si trasloca lo lascia così com'è.

STRATO DATA LINK

Si trasformano i blocchi di dati in pacchetti "parole di codice" (codeword): se il blocco ricevuto non è un codeword viene considerato errore.

Più le codeword sono "lontane" tra loro più è facile rilevare eventuali errori.

Ridondanza: aggiungono bit extra alla trasmissione per cercare eventuali errori. In particolare vengono aggiunti "bit di parità" per cercare eventuali modifiche nei bit.

Parità: si aggiunge un bit per rendere pari il numero dei bit a 1 nel caso della parità Pari, dispari nel caso di quella dispari. Se all'arrivo il numero dei bit a 1 è diverso dal previsto si rileva l'errore. Buone prestazioni perché non rileva un numero pari di errori. Per risolvere il problema si utilizzano codici bidimensionali.

TCP: transmission Control Protocol

È un protocollo di rete di livello trasporto. Presente solo negli host e non nei nodi (commutatori).

FUNZIONALITÀ:

- Instaura una connessione, tra mittente e destinazione, da rimanere attiva anche in assenza di scambio di dati e viene chiusa quando è esplicitamente chiesto. Grazie, mantiene e abbina una connessione.
- Garantisce la consegna dei dati attraverso il meccanismo degli ACKNOWLEDGEMENTS (notifiche) unica (at most once) senza eccezioni (checksum), controlla il flusso tra i termini in comunicazione e la congestione. Nella connessione offre un servizio di multiplexazione.

Se $n = \text{bit blocco da proteggere}$ e $n-k = \text{bit di controllo} \Rightarrow n = \text{bit codeword}$

La capacità del ricevente di rilevare e correggere l'errore è detta FORWARD ERROR CORRECTION (FEC) associata spesso a protocolli ARQ: in caso di errore rilevato il ricevente chiede di inviare il messaggio nuovamente.

\Rightarrow Alternative più usate: Internet checksum e Codici Polinomiali a ridondanza circolare (CRC)

Il checksum è una serie di bit di controllo per rilevare errori che viene inserito nell'header delle PDU. Se lo stringo da "poligono" è composto da parole di 16bit (bi). Il checksum si calcola così: ogni stringa è un intero $x = b_0 + b_1 + \dots + b_{L-1}$ modulo $2^{16}-1$

$\text{CHECKSUM} \Rightarrow b_L = -x \pmod{2^{16}-1}$ quindi il tutto deve soddisfare $\Rightarrow 0 = b_0 + b_1 + \dots + b_L \pmod{2^{16}-1}$

L'oggi più usati sono i codici a ridondanza ciclica in cui i dati sono visti in forma polinomiale. Supponendo che mittente e destinatario si sono accordati su uno stringa G generatore formata da $d+r$ bit. Al momento dell'inizio il mittente crea una stringa di $d+r$ bit divisibile per G. Alla ricezione se $(d+r)/G$ genera resto si è verificato un errore.

$d+r$ è dato dalla formula $D \times 2^r \text{ XOR } R$

$$\Rightarrow D \times 2^r \text{ XOR } R = nG$$

Applicando xor a entrambi i membri otteniamo $D \times 2^r = nG \text{ XOR } R$: dividendo $D \times 2^r$ per G si nota che $R = \text{resto di } [(D \times 2^r) / G]$

CRC può rilevare errori a burst fino a $r+1$ bit

FRAMMING

Lo studio di collegamento divide il flusso dati in "fronne" adatti ad una comunicazione a pacchetto. Questi pacchetti sono composti da un **Header** e un **Trailer** che variano al variare del protocollo usato.

Ogni fronna ha un inizio e uno fine contrassegnati da 2 flag che vengono letti quando il frame viene ricevuto per avere una lettura corretta del dato.

FLAG = 0111110 (6 uni)

In trasmissione, dopo cinque 1 consecutive viene sempre aggiunto uno 0, mentre in ricezione si esamina lo spazio dato cinque 1:

se è 1: lo segnala è un flag

se è 0: si elimina lo zero aggiunto prima.

Questa procedura è chiamata **Bit Stuffing**.

- **BYTE Stuffing**

Se in trasmissione il frame contiene la sequenza 011110 (escluso il flag)

o la sequenza 0111101, ad esse viene permessa la sequenza 0111101.

In ricezione:

se ricevo due sequenze 0111101 consecutive, una viene eliminata

se ricevo 0111101 seguita da 011110 la prima sequenza viene eliminata

se ricevo solo 011110 essa viene riconosciuta come flag

Protocolli ad Accesso Multiplo

Sono protocolli che hanno il compito di regolare il traffico simultaneo su un singolo collegamento BROADCAST avendo in collegamento in cui più nodi possono ricevere da trasmettere contemporaneamente. In questo tipo di collegamento può succedere che si ricevano più frame nello stesso momento causando la sovrapposizione degli stessi e, se ciò accade frequentemente, la corruzione del messaggio.

Per tale motivo questi protocolli utilizzano speciali regole per regolamentare il flusso e si dividono in:

- Protocolli a suddivisione del canale (Channel Partitioning Protocol)
- Protocolli ad accesso casuale (Random Access Protocol)
- Protocolli a rotazione (Token-Turn Protocol)

I **protocolli a suddivisione del canale** suddividono appunto il canale in intervalli temporali che vengono assegnati ai singoli nodi (nel caso TDMA) o dividono il canale in più canali con differenti frequenze (nel caso FDMA).

Entrambi i casi evitano collisioni tra frame, ma nel primo caso un nodo deve attendere il suo turno anche se i precedenti non hanno finito di inviare e nel secondo lo frequenta è limitato anche se è solo un nodo a trasmettere.

Un terzo caso è quello CDMA (Code Division Multiple Access) in cui ogni utente decodifica la trasmissione con un proprio codice personalizzato per evitare collisioni, ma è molto dipendente dalla scelta dei codici.

I **Protocolli ad accesso casuale** si dividono in: Aloha, Slotted Aloha e CSMA.

Aloha non prevede vincoli sull'invio di dati: quando un nodo ha dati da trasmettere li trasmette. Il successo dell'invio dei dati sta nel fatto che ci sia una collisione o meno. In caso affermativo i frame vengono eliminati indipendentemente dall'entità dell'errore e reinvia dopo un'attesa CASUALE. Questa attesa "casuale" è comunque vincolata dal fatto che bisogna evitare di avvenire di nuovo le collisioni. Per fare ciò si utilizza un meccanismo di Back-off secondo il quale l'attesa varia da 0 a $2^{(k-1)} \times T$ dove T = tempo di trasmissione del messaggio e k può dipendere dal numero di collisioni già avvenute.

Slotted Aloha aggiunge al precedente il vincolo della suddivisione del tempo in slot: ogni nodo (quando pronto a trasmettere) è obbligato ad iniziare la trasmissione precisamente all'inizio dello slot. Così facendo 0 due o più frame colidono perfettamente in uno slot o non colidono affatto.

CSMA (Carrier Sense Multiple Access = protocollo ad accesso multiplo con rilevamento dello portante) implementa lo obiettivo: "Ascolta prima di trasmettere". Se trovi il canale occupato aspetta e riprova più tardi secondo uno modulato di ritrasmissione stabilita".

A sua volta il protocollo CSMA si divide in 3 varianti a seconda della metodologia di attesa

- CSMA Per sistente controlla continuamente se il canale è libero: se libero trasmette, se c'è una collisione attende un intervallo casuale
- CSMA Non per sistente controlla inizialmente se il canale è libero, se non lo è attende un intervallo casuale di tempo e poi controlla di nuovo (Back off)
- CSMA P-Per sistente attende se il canale si libera e poi il nodo trasmette con probabilità P . In seguito con probabilità $1-P$ attende un breve periodo e riesegue il carrier sensing

Il CSMA può avere lo funzione Collision Detection (CSMA/CD) che rileva le collisioni durante i trasferimenti. In caso di collisione arresta il trasferimento di tutte le "stazioni" e gli applica un ritardo che aumenta esponenzialmente (Back off). Il CD riduce le durate delle collisioni e quindi aumenta l'efficienza del trasferimento.

Un esempio di CSMA/CD è la commissione Ethernet LAN.

1-persistent CSMA, $R = 10 \text{ kbit/s}$, $T_{\text{prop}} = 51,2 \mu\text{s}$ ($512 \text{ bit} = 64 \text{ byte slot}$, distanza massima $2,5 \text{ Km} + n$ ripetitori), TRUNCATED BINARY EXPONENTIAL BACKOFF: dopo l' n -esima collisione, il tempo di backoff è moltiplicato n volte ($0,1,2^k-1$) dove $k = \min(n, 10)$

I protocolli a rotazione sono detti anche **ad accesso controllato** e si dividono in: protocolli Polling e protocolli token passing.

POLLING: un nodo master da il permesso di trasmettere a turno ai nodi. Ogni nodo a rotazione invia un numero fisso di frame: fatto ciò si ferma e il nodo master da il via al successivo nodo. Anche se capita 6 collisioni, questo protocollo non è esente dai difetti: prima tra tutti il fatto che se si guasta il modo principale tutto il sistema si ferma. Il secondo problema è che dare il permesso di inviare ad un nodo genera ritardo (ritardo di polling): anche se solo un nodo è utilizzato, ci sarà comunque un ritardo generato dal nodo master che comunica con i nodi multipli.

TOKEN PASSING: un nodo può inviare solo se in possesso di un token, ovvero in posizione di controllo che posso da un nodo ad un altro con un ordine fissato.

Quando un nodo finisce di inviare passa il token al nodo successivo fino a ricominciare il giro. Anche questo protocollo seppur molto efficiente può avere problemi: se si guasta un nodo il sistema si ferma e se capita che un nodo si "dimentica" di inviare il token sono necessarie operazioni di recupero.

AUTOMATIC REPEAT QUESTION (ARQ)

L'ARQ è un metodo di controllo di errori basato sul reinvio automatico del dato (frame) in caso di ricezione d'errore.

Se il ricevente nota un errore invia una richiesta al mittente di invio di nuovo il pacchetto. Le sue varianti sono: Stop-and-wait ARQ, Go-back-N ARQ e Selective Repeat ARQ.

- **Stop-and-wait ARQ** è un semplice metodo stop-and-wait in cui cioè il mittente smette di inviare finché il destinatario non invia a sua volta un Acknowledgment (ACK) in caso di messaggio ricevuto correttamente o un Negative Acknowledge (NAK) se il messaggio ha errori o non è arrivato effettivo. Se il mittente non riceve l'ACK entro un certo periodo (Timeout), invia il messaggio di nuovo. Questo può creare duplicati poiché gli ACK possono subire ritardi. Per risolvere il problema viene usato un bit di sequenza nell'header della frame. Ricevuta la frame, l'ACK conterrà anche esso il bit di sequenza per evitare ACK duplicati che potrebbero generare errori. Tale metodo non è efficiente su collegamenti ad alta velocità o con elevati ritardi di propagazione.

- **Go-Back N ARQ** opposta delle piccole modifiche al precedente metodo: invece di inviare un frame alla volta, ne invia N (window size) a rotta.

In caso di duplicati e/o errori analogamente al precedente viene reinviato l'intera sequenza di frame. Anche qui gli header contengono i bit di sequenza da cui ACK corrispondono all'ultima frame ricevuta con successo. La miglioria rispetto allo Stop-and-wait è che il tempo da prima veniva usato per "aspettare" ora è usato per inviare altri frame. Il contro di ciò è che in caso di errore bisogna inviare di nuovo molti più frame impiegando più tempo.

- **Selective Repeat ARQ** risolve il problema del Go-Back N: invia solo le frame corrette o non ricevute.

Il procedimento è semplice: il mittente inizia ad inviare la sua finestra di frame ricevendo un ACK per ogni frame correttamente inviata. Nel momento in cui uno frame presenterà un errore, il mittente riceverà un NAK con il bit di sequenza di tale frame nell'header. Il mittente continuerà ad inviare le frame successive e salverà nel buffer le frame fino allo fine della finestra. Terminata la finestra, verranno reinviate le frame nel buffer dopo aver rimandato la frame mancante e poi si continuerà con la finestra successiva.

FLOW CONTROL

Ha lo scopo di prevenire gli overflow del buffer di ricezione regolando il tasso di emissione delle frame da parte del trasmettitore

Protocollo PPP

C'è solo un mittente e un destinatario

FUNZIONI:

- framing dei pacchetti
- Rilevazione degli errori
- Disponibilità della connessione

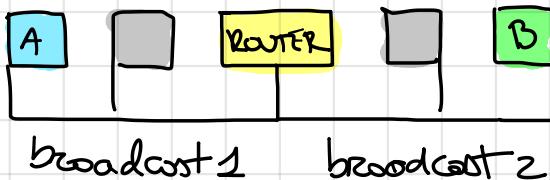
INDIRIZZI MAC

Indirizzi IP (32bit): livello di rete analogo all'indirizzo di cosa che combia quando si cambia residenza (rete)

Diversi sono gli indirizzi MAC che risultano simili al codice fiscale (48 bit) e quindi immutabile. Ogni scheda di rete (nel mondo) ha un indirizzo MAC differente composto da 6 byte in esadecimale. Quando viene costruita una scheda di rete la società produttrice compone un blocco di indirizzi univoci per montarne tali.

ADDRESS RESOLUTION Protocol: utilizzato per associare ad ogni IP il proprio indirizzo MAC. Il mittente invia un pacchetto ARP con un IP a tutti i nodi nel broadcast. Se uno di essi si riconosce nell'IP risponde con il MAC. Lo stesso ARP avendo lo stesso in cui si associano indirizzi IP e MAC è "plug and play" ovvero una volta inizializzata si aggiornano automaticamente senza l'urto dell'amministratore.

N.B. Non è possibile inviare un messaggio ARP da un nodo A in un canale broadcast ad un nodo B situato in un altro canale broadcast collegato da un router senza sapere l'indirizzo IP di B. Per fare ciò è necessario usare il router da intermediazione: A → ARP → Router ⇒ Router → ARP → B



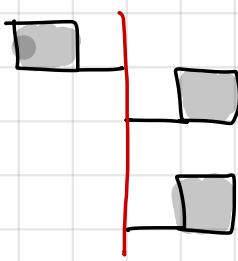
Tramite IP con un messaggio ARP scopre il MAC del Router che mi garantisce poi di mandare tale messaggio ARP attraverso il Router a B

ETHERNET (802.3 IEEE standard)

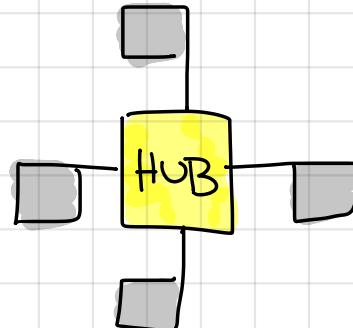
802.15 = BLUETOOTH

802.11 = WIFI

TIPOLOGIA BUS



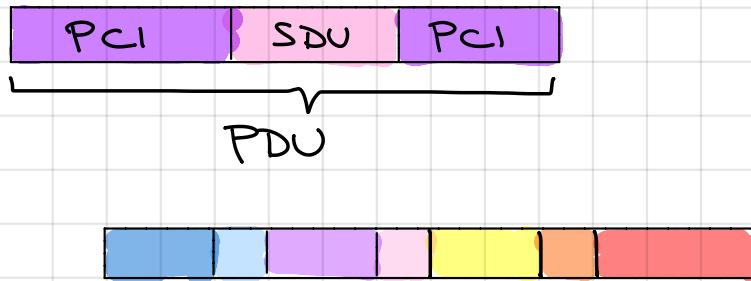
TIPOLOGIA STELLA



Ethernet usa CSMA/CD.

È necessario avere un messaggio minimo di 64 byte e massimo 1500 m di lunghezza di rete altrimenti non potrebbe funzionare il collision detection.

FRAME ETHERNET



Preamble: serve per attivare la rete dei riceventi e sincronizzare i loro clock con il mittente.

Start Delimiter: indica l'inizio del frame.

Indirizzo di destinazione: il modo che si riconosce nel Mac copia il frame nel buffer, gli altri lo ignorano.

Length: indica il #di byte del campo informativo (Frame max 1518 byte, campo informativo max 1500 byte).

Dato

PAD: garantisce che la frame sia minima di 64 byte.

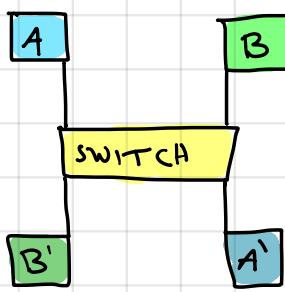
CRC

HUB (Architettura a stella)

Il livello Logico è equivalente alla precedente architettura, ma il livello fisico è più semplice ed elimina problemi come l'interruzione di retransmissione in caso di esaurimento del collegamento. I problemi di collisione persistono.

SWITCH (BRIDGE)

Permette di aumentare il limite di 500m del collegamento: invece di mandare un messaggio a tutte le porte, lo manda solo a quella in cui è collocato il destinatario.

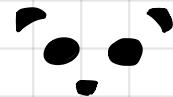


Lo switch elimina i domini di collisione e permette il collegamento simultaneo tra A e A' e B e B'. Anche lo switch ha la sua tabella (switch table) in cui vengono associate le porte con i relativi indirizzi MAC. Lo switch è "self learning" ovvero si aggiorna da solo: se non trova la porta (può non presente in tabella) manda il messaggio a tutti, ma lo imposta porta e MAC del mittente riempiendo piano la tabella.

DESTINAZIONE GLOBALE = FLOOD DESTINAZIONE NOTA = SELECTIVE SEND

Se lo switch rileva un Mac già presente in tabella, ma su un altro porto
la tabella viene aggiornata allo "stato" più recente.

RIASSUNTO STRATO COLLEGAMENTO



Invia Frame

FRAMING: con bit stuffing e byte stuffing

RIVELAZIONE ERRORE: bit parita (e bidimensionale), internet checksum e codice a ridondanza ciclica (composizione con polinomio generatore)

CORREZIONE ERRORE: FEC o ARQ

Protocolli Accesso Multiplo (MAC): suddivisione canale (TDMA, FDMA, CDMA),

Accesso Comune: canale non diviso per tempo o frequenza, ma condiviso \Rightarrow collisioni (Aloha, slotted Aloha, CSMA-CD)

Accesso Controllato \Rightarrow A turni (Token Passing, Polling)

Controllo Errore Hop by Hop (ogni passaggio) o end-to-end (solo alla fine)

STOP-AND-WAIT: Se ricevo Ack positivo invio il frame seguente, se NAK reinvio lo stesso.

GO BACK-N: Come stop-and-wait, ma divide il tutto in finestre temporali in caso di NAK. Ricomincia dall'inizio della finestra. Ritardo maggiore di NAK, minore di ACK.

Selective Repeat: in caso di NAK reinvio solo quelli errati e solo gli altri in buffer nell'attesa.

INDIRIZZO MAC: univoco per ogni scheda rete, salvati nella tabella ARP in cui c'è la corrispondenza tra IP e MAC

Tabella plug & play autoaggiornante

All'invio di un frame se non c'è corrispondenza in tabella mondo ma richiesta del IP a tutti e aspetto risposta

Formule Accesso Comune

Prodotto Bonda Ritardo = R (banda canale bits) \times d (tempo propagazione s)

Tempo Propagazione = $\frac{\text{distanza}}{\text{velocità mezzo fisico}}$

Intervallo Vulnerabilità = 2 t. propagazione

Prodotto Bonda ritardo Normalizzato = $\frac{T_{prop}}{\frac{L}{R}}$ L = lunghezza frame
 R = banda canale

Efficienza = $\frac{1}{1+2\alpha} = \frac{L}{L+2t_{prop}} \cdot R$ α = prodotto banda ritardo normalizzato

Throughput massimo = $\frac{1}{1+2\alpha} R = \frac{L}{\frac{L}{R} + 2t_{prop}}$

Formule Stop&Wait

Temp transazioni Frame = $2t_{prop} + 2t_{proc} + \frac{n_f}{R} + \frac{n_a}{R} = T_0$ n_f = lunghezza frame
 n_a = lunghezza ACK

Rate Transmission Efficace = $\frac{\# \text{bit informativi consegnati}}{\text{tempo trasferimento}} = \frac{n_f - n_a}{T_0}$

Efficienza = $\frac{\text{Rate T. Efficace}}{R}$

$$\text{Capacità base subcomiale: } C_S = \frac{N_{\text{bit IT}}}{T_{\text{trama}}} = \frac{\# \text{ bit per IT}}{\text{durata trama}} \quad \text{NB. Capacità Totale} = C \cdot \# \text{ IT}$$

Proteggiere in base banda di picco di una linea di capacità C

$$\sum_{i=1}^N F_{p,i} \leq C \quad N = \text{sorgenti multiplete} \quad F_p = \text{ritmo binario picco}$$

Proteggiere in base banda media di una linea di capacità C

$$\sum_{i=1}^N F_{m,i} \leq UC \quad N = \text{sorgenti multiplete} \quad F_m = \text{ritmo binario medio} \\ U < 1 = \text{rendimento utilizzazione massimo linea multiplete}$$

$$\text{Capacità Comiale Multitrama} \quad C_M = \frac{N_{\text{bit IT}}}{M \cdot T_{\text{trama}}} \quad M = \# \text{ trame per multitrama}$$

$$\text{Ritardo trasferimento Pacchetto} \quad D_{PAC} = \sum_{i=1}^N \left(\frac{H+L}{C_i} + d_{\text{prop},i} \right)$$

N = # interfacce (zomi) C_i = ritmo trasmissione del zomo

d_{prop} = ritardo propagazione = $\frac{\text{distanza}}{\text{Velocità prop.}}$ H = bit di Header L = bit di dati

$$\text{Ritardo Messaggio: } D_{MES} = d_{\text{prop},AB} + \frac{H+L}{C} \cdot (N-1) + \underbrace{\frac{1}{C} \left\{ \lceil \frac{X}{L} \rceil \cdot (H+L) \right\}}_{\# \text{ pacchetti} \cdot \text{dim pacchetto}}$$

N = # collegamenti / interfacce

$$\text{Ritardo Messaggio (L variabile)} \quad D_{MES} = d_{\text{prop},AB} + \frac{H+L_{\text{MAX}}}{C} \cdot (N-1) + \frac{X + \lceil \frac{X}{L_{\text{MAX}}} \rceil \cdot H}{C}$$

$$L_{\text{MAX}} = \sqrt{\frac{H \cdot X}{N-1}}$$

$$\text{Ritardo flusso continuo} \quad D_{Bn} = \frac{L}{R} + (L+H) \sum_{i=1}^N \frac{1}{C_i}$$

$\frac{L}{R}$ = ritardo
pacchettizzazione

Ritardo trasferimento (Rete Circuito)

$$D_{BIT} = N \cdot d_{\text{prop}} + (N-1) d_{\text{trama}} \quad d_{\text{trama}} = \text{ritardo attraversamento nodo a circuito}$$

$$D_{MES} = D_{BIT} + \frac{X}{C}$$

$$\# \text{IT} = \left\lceil \frac{\text{ritmo binario}}{\text{capacità subcomiale}} \right\rceil$$

$$\# \text{frame} = \left\lfloor \frac{\text{capacità subcomiale}}{\text{ritmo binario}} \right\rfloor$$

$$\# \text{bit} = \left\lceil \frac{\text{ritmo binario}}{\text{capacità subcomiale}} \right\rceil \cdot \text{lunghezza IT}$$

$$\text{Lunghezza Finestra} = \underbrace{z_{\text{prop}} + t_{\text{trasm frame}} + t_{\text{host Acq}} + t_{\text{elaborazione}}}_{\frac{L}{R}}$$

$$\text{Efficienza} = \frac{p \cdot t_{host}}{p \cdot t_{transf} + 2t_{prop}} = \frac{d_{max}}{\text{Vel. prop}}$$

Differenza tra hub e switch

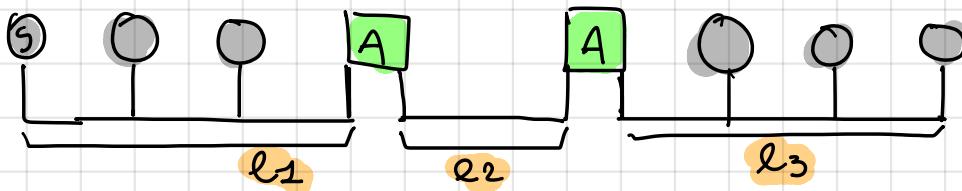
d_{max} si riferisce alla distanza massima tra i domini di collisione

1 solo dominio = somma distanze contenute

più domini = punto lo più grande

ESERCIZI

- CSMA / CD



$$P_{\text{success}} = \frac{1}{e} e^{-2,71}$$

DATI

$$F = 128 \text{ byte (Tutte)}$$

$$R = 10 \text{ Mbit/s}$$

$$l_1 = l_3 = 100 \text{ m}$$

$$l_2 = 300$$

$$c = 10 \cdot 10^6 \text{ m/s}$$

a) Che succede se A sono ripetitori?

Possono esserci collisioni tra tutte le stazioni.

b) Se sono bridge?

Si denotano 3 domini di collisione e quindi possono esserci collisioni solo nello stesso dominio di collisione.

$$\overline{\text{EFFICIENZA}} = \frac{1}{1 + 2t_{\text{prop}}} = \frac{P \cdot t_{\text{tx}}}{P \cdot t_{\text{tx}} + 2t_{\text{prop}}}$$

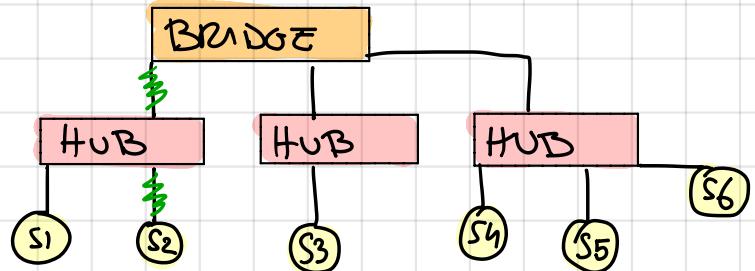
t_{tx} = tempo di trasmissione Pacetto
 t_{prop} = tempo di propagazione

$$t_{\text{tx}} = \frac{F}{R} \quad t_{\text{prop}} = \frac{\text{distanza max}}{c} (= l_1 + l_2 + l_3)$$

$$\begin{aligned} E &= \frac{1}{1 + 2 \left(\frac{l_1 + l_2 + l_3}{c} \right)} \\ &= \frac{1}{1 + \frac{1000}{10 \cdot 10^6}} = 0,27 \end{aligned}$$

moltiplico per 8 per trasformare da byte a bit

E = valore minimo tra le efficienze dei 3 blocchi e dato che (bridge) l'efficienza è inversamente proporzionale alla distanza basta calcolare l'efficienza del blocco più grande.



$$\begin{aligned}
 F &= 128 \text{ byte} \\
 R &= 10 \text{ Mbit/s} \\
 L &= 50 \text{ m} \\
 C &= 10 \cdot 10^8 \text{ m/s} \\
 \text{Processo} &= \gamma_e
 \end{aligned}$$

Efficienza trasferimento da S1 a S2

$$E = \frac{1}{1 + \frac{2 \cdot 2L \cdot R}{C \cdot F \cdot P}} = \frac{1}{L \cdot \frac{200 \cdot 10 \cdot 10^6}{10 \cdot 10^8 \cdot 128} \cdot 8 \cdot \frac{1}{e}}$$

N.B. 2L perché l'hub invia a tutte le porte a sua disposizione

STRATO DI RETE

Permette di trovare il percorso ottimale da percorrere per inviare dati da un terminale ad un altro

- FORWARDING (INOLTRI) funzione attuativa
- ROUTING (INSTRUZIONAMENTO) funzione decisionale
- SWITCHING



La funzione di routing ha come side-effect la creazione di tabelle di forwarding: quanti sono i nodi che compongono la rete (simile alla tabella ARP). Dopo la fase di routing inizia la fase attuativa di forwarding che regole le istruzioni date dalla fase di routing (decisione del cammino migliore)

TIPOLOGIE DI SERVIZI DI RETE

- CONNECTIONLESS (reti a datagramma)
- CONNECTION-ORIENTED

Le sostanziali differenze stanno nel fatto che nel primo i pacchetti sono inviati senza "ovisori" destinatario della rete. I pacchetti sono trattati dalla rete e quindi da ciascun nodo come entità indipendenti e il loro percorso è scelto pacchetto per pacchetto (router con funzionamento STATELESS).

I servizi con connessione costituiscono reti a "circuito virtuale" (V.C.): prima di ogni invio di pacchetti viene instaurata una connessione di rete con un percorso deciso al momento di tale instaurazione e i router (STATEFUL) in questo caso mantengono i dettagli dello stato della connessione.

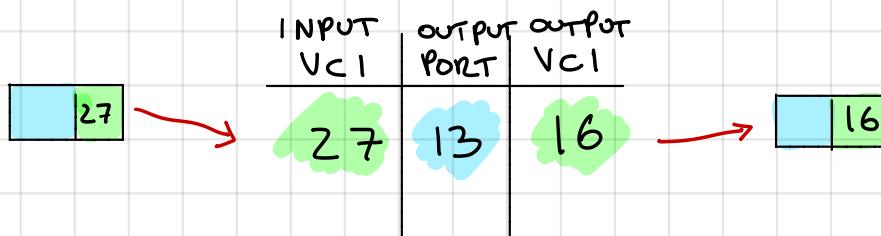
CONNECTION SETUP (VC)

Vengono trasmessi messaggi di negoziazione lungo il percorso scelto e ogni nodo lo trasmette al modo successivo innanzitutto la tabella di forwarding e completandolo una volta arrivato a destinazione.

Ogni pacchetto che utilizza questa connessione ha lo stesso VIRTUAL CIRCUIT IDENTIFIER (VCI) nell'header, tag locale che identifica lo "strada" da seguire.

Il connection setup ha un ritardo (C.S. DELAY) che si somma al ritardo di transito dei pacchetti: tollerabile se minore del tempo di trasferimento dei pacchetti, inaccettabile se sono trasferiti pochi pacchetti.

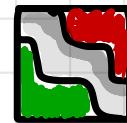
Ad ogni nodo (switch) il pacchetto cambia VCI



Nelle reti a datagramma i router utilizzano gli indirizzi di destinazione per effettuare il forwarding: ogni pacchetto può uscire per percorsi diversi e non è garantita la consegna in sequenza.

Nella tabella di routing le uscite vengono definite per "prefisso": tutti gli indirizzi con gli stessi n bit iniziali comuni ridono lo stesso porta di uscita. Non utilizzando però prefissi con la stessa dimensione possono verificarsi casi in cui lo stesso indirizzo può avere "più uscite".

Esempio:
11001000 00010111 000111000
11001000 00010111 00011



In questo caso viene "usato" quello di dimensione maggiore.

Protocollo IP (RFC 791, 919, 922, 950, 1349)

Il protocollo IP è un protocollo di livello di rete che opera con modalità di trasferimento CONNECTIONLESS.

Determina il formato dei pacchetti (max 65536 byte), lo schema di indirizzamento, la modalità di indirizzamento ed esegue, se necessario, la frammentazione e ri-assemblaggio delle unità dati.

Un pacchetto IP è diviso in:

Vers: la versione del protocollo (anche più insieme, 4 bit)

Header Length (HLEN): ovvero la lunghezza dell'intestazione in parole a 32 bit, con una parte fissa (2 byte) e una opzionale, valore massimo, 6 byte (4 bit)

Total Length: lunghezza totale in byte di intestazione + payload, massimo 65536 byte (16 bit)

Service Type: specifica i parametri di qualità di servizio richiesti dall'utente per il pacchetto

Precedence: indica il livello di priorità del pacchetto (3 bit)

Type of Service (TOS): indica quale servizio è richiesto per il pacchetto e secondo di quale posizione occupa 1 nei 4 bit

1000: Minimize Delay

0100: Maximize Throughput

0010: Maximize Reliability

0001: Minimize Monetary Cost

0000: Normal Service (Default)

Ogni rete fisica ha un valore massimo di lunghezza della propria unità informativa (MTU). Se l'MTU dello sottorete è inferiore alla lunghezza del datagramma il pacchetto viene frammentato dal router/host prima del rilascio nella sottorete e ricomposto nell'host di destinazione.

Identification: numero identificativo del pacchetto da frammentare assegnato dal processo sorgente (16 bit)

Flags: X = non uscito e posto a zero

DF = Don't Fragment (0: frammentazione permessa; 1: vietata)

MF = More Fragment (0: ultimo frammento del datagramma; 1: non ultimo) (3 bit)

Fragment Offset: posizione del frammento all'interno del pacchetto (in unità di 8 byte), consente di voler fare l'interpretazione del pacchetto.

Time To Live (TTL): indica il numero massimo di router che il pacchetto può attraversare, inizializzato dallo sorgente e decrementato ad ogni router. Se arriva a zero, il pacchetto è scartato e viene inviato un messaggio ICMP di notifica verso lo sorgente (8 bit)

Protocol: indica a quale protocollo dello strato superiore deve essere trasferito il contenuto informativo del pacchetto (es. TCP=6, 8 bit)

Header Checksum: protegge solo l'intestazione del pacchetto, in caso di errore lo scarta (16 bit)

Source Address (32 bit) e Destination Address (32 bit)

Options: Record Route Option (RRO): lista che si riempie di indirizzi IP ogni volta che attraversa un router

Timestamp Option: come RRO, ma aggiunge anche l'istante

Loose Source Routing Option (LSRO): specifica la lista di router che il pacchetto deve attraversare

Strict Source Route Option (SSRO): specifica tutti i router che il pacchetto deve attraversare. Si differenzia dal precedente perché in questo caso la lista è compilata al momento dell'invio e non "step-by-step" (multiplo di 8 bit)

Padding: rende l'intestazione multiplo di 32 bit inserendo zeri.

INTERNET CONTROL MESSAGE PROTOCOL: consente ai router di inviare all'host sorgente informazioni riguardanti anomalie nel processamento di un pacchetto. ICMP è parte integrante di IP e i suoi messaggi sono incapsulati nella parte dati dei pacchetti IP.

I router intermedi non elaborano questi messaggi e non vengono generati messaggi di errore riguardanti anomalie dei messaggi di errore stessi.

INDIRIZZAMENTO IP

Un indirizzo IP identifica un'interfaccia di rete. Ogni router avrà un IP per ogni rete a cui è connesso e ogni router avrà un IP per ogni rete che gestisce.

Un IP pubblico è unico in tutta la rete

NOTAZIONE NUMERICA (32 bit) es. 10010111 01100100 00001000 00010010
NOTAZIONE DOT-DEC 151.100.8.18

Uno sottorete è una rete isolata i cui punti terminali sono collegati all'interfaccia di un host o di un router ed è anche detta **rete IP**

ES. Un link diretto tra due router è uno sottorete con due interfacce

Un indirizzo IP è formato da due parti:

Net-ID cioè l'identificativo di sottorete (prefixo) e l'Host-ID cioè l'identificativo di Host all'interno della sottorete

$$IP = \text{Net-ID}.\text{Host-ID}$$

Gli IP delle interfacce della stessa sottorete condizionano il Net-ID

Dal 1984 parte del campo usato dall'Host-ID è usato da Subnet-ID identificato da una maschera denominata **Subnet Mask** ovvero un parolo di 32 bit in cui i bit segnali a 1 identificano i bit del Net-ID e Subnet-ID e quelli segnali a 0 identificano i bit dell'Host-ID.

SUBNETTING STATICO: tutte le subnet hanno la stessa mascherata

$$\# \text{ max Subet} = 2^{18-2} = 262.142$$

$$\# \max \text{host per substrate} = 2^6 - 2 = 62$$

SUBNETTING A LUNGHEZZA VARIABILE: le sottoreti hanno maschere diverse.
Così corrente di gestire reti di dimensioni diverse

ARCHITETTURE ROUTER

Un router è composto da porta d'ingresso, switch fabbric e porte d'usata. La porta d'ingresso è composta da **line termination** che riceve i bit a livello fisico, **DATA LINK PROCESSING** che specifica il protocollo e dai **buffer** (che determinano anche la porta d'usata).

Le switch fabbrica porta il pacchetto alla porta d'uscita specifica.
Le porte d'uscita hanno in più il **packet scheduling** che manda i pacchetti dal buffer e li trasmette nel giusto ordine.

HÉTÓN DI INDIRIZZAMENTO

- shared memory \rightarrow ho il difetto di dipendere dalla lentezza della CPU
 - bus \rightarrow dipende dalla velocità del bus stesso
 - crossover

ROUTING IN RETI IP

Sia gli host che i router hanno la stessa tabella di routing.

HOST

Se lo destinazione è nello stesso rete, il pacchetto è inviato usando direttamente l'interfaccia di rete, altrimenti è inviato al router di default

ROUTER

Legge l'IP di destinazione nel pacchetto entrante: se lo destinazione è nello rete collegata al router il pacchetto è inviato usando l'interfaccia di rete, altrimenti si accede alla routing table per decidere il passaggio successivo. (NEXT-HOP)

CLASSLESS INTER DOMAIN ROUTING CIDR

È utilizzato per utilizzare lo spazio di indirizzamento IP con più efficienza e diminuire la complessità della tabella di indirizzamento

Ogni rete avrà un certo numero di blocchi contigui di indirizzi (Super Netting) e sarà corrispondente da un prefisso (bit più significativi)

Un insieme di reti corrispondente da blocchi contigui di indirizzi sarà identificato da un unico prefisso.

Gli indirizzi di classi A e B sono assegnati solo in caso di necessità.

Sono assegnati blocchi consecutivi di classe C

da metà inferiore degli indirizzi di classe C e assegnato su base geografica

ADDRESS ALLOCATION Policy

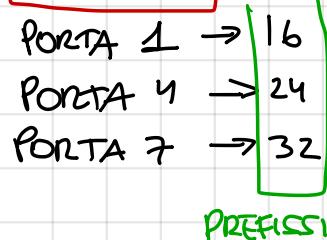
ESE. ISP più grande (CONTINENTAL) divide i suoi blocchi in "sottoblocchi" contigui che vengono assegnati a ISP più piccoli (STATAL).

LONGEST PREFIX Matching

In una routing table è possibile avere lo stesso prefisso più volte: in questo caso viene scelto il percorso con il prefisso di matching più lungo

ESE.

198.15.7.3



DYNAMIC HOST CONFIGURATION PROTOCOL

È un protocollo che consente di configurare l'host (IP, Subnet, DNS ecc) e gli permette di ottenere dinamicamente il suo indirizzo IP dal server di rete.

FUNZIONAMENTO:

L'host invia un messaggio broadcast DHCP discover

Il server DHCP invia l'indirizzo con il messaggio DHCP offer

L'host richiede la configurazione con il messaggio DHCP request

Il server DHCP invia la configurazione con il messaggio DHCP ack

L'indirizzo IP può essere assegnato in 3 modi diversi:

DINAMICAMENTE: assegnato ad un host per un certo periodo di tempo

AUTOMATICAMENTE: assegnato permanentemente dal server

MANUALMENTE: assegnato dall'amministratore di rete

NETWORK ADDRESS TRANSLATOR (NAT)

- Riduce l'utilizzo dello spazio di indirizzi IP.
- È utilizzato nelle INTRANET (indirizzi privati che non possono essere utilizzati per fare routing)
- Assegna un indirizzo pubblico ad un host solo quando deve comunicare con l'esterno (traslazione indirizzo privato con uno pubblico)

Il NAT nasconde i dettagli della intranet al mondo esterno come dare anche la sicurezza.

Quando invia un pacchetto il NAT usa un numero di porta e lo sostituisce al numero di porta dell'host (interno) e sostituisce il suo IP a quello privato di origine. (Viceversa in ricezione)

ROUTING

Il routing (e la creazione dello tabella stessa) si divide in più categorie

CENTRALIZZATO: i percorsi sono determinati da un modo centralizzato e tutte le informazioni sono inviate ad esso. Difficile adattamento alle variazioni e di scarsa affidabilità.

DI STABILITO: i cammini sono decisi dai router tramite algoritmo distribuito e le informazioni sono scambiate tra i router. Maggiore adattabilità e alta scalabilità.

STATICO: cammini configurati manualmente che non variano, adatto a reti semplici e con percorsi prevedibili. Usato per impostare cammini specifici e per fornire un indirizzamento di default (default router)

DINAMICO: adatto a sostenere variazioni nel tempo coelalo automaticamente i cammini.

I percorsi sono decisi in base allo stato della rete ricevuto per mezzo di un protocollo di indirizzamento. Usato dalle reti IP per internet non stabile o possibili guasti nei nodi.

In ognuna esistono due tipi di router in base alla "gerarchia" dei router che compongono la rete.

FAT: tutti i router sono sullo stesso livello (peer), scarsa scalabilità

GERARCHICO: la rete è divisa in domini, sistemi autonomi ecc., alcuni router fanno parte dello backbone e altri comunicano solo con un router della stessa area. Soluzione efficiente e scalabile.

SISTEMI AUTONOMI: sono insiemi di host e router controllati da una singola autorità amministrativa (es. ISP) (AS). La backbone di internet è costituita da un particolare AS detto CORE AS i cui router sono chiamati CORE ROUTER, gli altri AS sono detti STUB AS.

Ogni AS ha il proprio protocollo di instroadamento.

Ogni stub AS deve avere almeno un router connesso ad un core router (**EXTERIOR ROUTER**). I router interni sono detti **INTERIOR ROUTER**.

I protocolli di instroadamento in un AS si chiamano **INTERIOR GATEWAY Protocol (IGP)**, se coinvolgono più AS sono detti **EXTERIOR GATEWAY PROTOCOL (EGP)**

Un EGP svolge 3 funzioni:

- 1) individuazione dei router adiacenti con cui scambiare le informazioni di instroadamento
- 2) verifica continua delle funzionalità dei router interlocutori
- 3) scambio periodico delle informazioni di instroadamento, queste riguardano la sola raggiungibilità delle reti, non la distanza.

Il protocollo di instroadamento mette in grado ogni router di determinare il modello a grafo della rete (collegamenti tra router e relativi costi) mentre l'algoritmo di instroadamento determina il cammino a costo minimo tra due nodi della rete. Le metriche misurano la "qualità" di un link o di un comminno:

costo basso se il link è ad alto qualità e do includere nel comminno se possibile;

costo elevato se il link è di bassa qualità e do evitare se possibile.

Le metriche si differenziano in base a ciò che misurano: # di Hop, affidabilità, ritardo, bandwidth e carico.

Per trovare il comminno minimo si usano due metodi:

1) DISTANCE VECTOR Protocol

I nodi adiacenti si scambiano la lista delle distanze verso le destinazioni e viene determinato il **Next-Hop** migliore per ogni destinazione tramite l'algoritmo di Bellman-Ford.

2) LINK STATE Protocol

Le informazioni sullo stato del link (costi) sono diffuse in rete tramite **flooding** per permettere a tutti i router di conoscere l'intera topologia della rete: ogni router calcola il comminno minimo e il next-hop verso ogni destinazione tramite l'algoritmo di **DJIKSTRA**.

INSTRADAMENTO GERARCHICO

PROBLEMI:

- 1) Ogni ISP deve poter amministrare la sua rete e allo stesso tempo avere la possibilità di collegarsi alle altre
- 2) Troppo connessioni richiedono molta memoria

Si organizzano i Router in sistemi autonomi (AS)

Ogni AS è collegato con gli altri tramite router definiti GATEWAY

In caso di connessioni verso l'esterno il pacchetto è inviato direttamente al gateway specifico per la destinazione

I protocolli d'indirizzamento IANA-AS sono noti come Interior Gateway Protocol.

RIP: Routing Information Protocol

OSPF: Open Shortest Path First

IGRP: I. G. Routing Protocol (Cisco)

R.I.P.

Protocollo Distance Vector: conteggio degli hop come metrica di costo ($\max = 15$, $16 = \infty$)

Applicato su reti piccole / medie, utilizza il protocollo UDP

REQUEST chiede ai vicini il distance vector

RESPONSE annuncia il distance vector

Ogni 30 sec. i router adiacenti si scambiano gli aggiornamenti d'indirizzamento: ogni messaggio contiene fino a 25 sottoreti e la distanza del mittente con esse

Se un router non riceve messaggi da un vicino entro 180 sec., quel router è considerato guasto o spento e tale informazione viene propagata in tutta la rete nello stesso modo di prima, aggiornando le tabelle di routing di tutti i router. Combinando i percorsi che utilizzano tali router.

O.S.P.F.

Protocollo Link State: utilizza le informazioni di stato dei link (Link State Advertisement - LSA) e l'algoritmo di Dijkstra

Ad ogni cambiamento di stato di link, il router emette un LSA a tutti gli altri tramite tecnica di flooding, ovvero direttamente da IP con un protocollo di livello superiore (rapida convergenza)

FLOODING: un LSA è rilasciato da un router su tutte le sue interfacce tranne quella da cui è stato ricevuto

Gli LSA contengono dei TIME STAMP / NUMERO DI SEQUENZA per evitare LSA "doppi" ricevuti da più router

Lo tecnico di flooding assicura che tutti i router abbiano le stesse informazioni sullo stato di link

Se da un solo fornisce affidabilità di ricezione, dall'altro può occupare molta banda.

Il protocollo OSPF risolve anche il problema della memoria per reti di grandi dimensioni: la rete è suddivisa in aree interconnesse ad un'area di BACKBONE indipendenti.

Ogni area ovviamente ha un AREA BORDER ROUTER - ABR che appartiene a più aree e contiene i database di esse e che emettono dei messaggi (summary records) contenenti la lista delle sottoreti raggiungibili tramite le aree a cui appartengono.

I border router forniscono l'external records ovvero il commesso minimo per raggiungere l'esterno.

B.G.P.

Permette a tutti gli AS di ottenere informazioni sulla raggiungibilità delle sottoreti di AS controllati, proporle a tutti i router interni e determinare i percorsi migliori in base a tali informazioni.

Consente alle sottoreti di farsi identificare da internet.

Un router che supporta BGP viene chiamato BGP speaker

NUERO SOTTORETI = 2^N dove N = # uni del numero in
binario dell'host nello maschera
es. 255.255.255.224 (classe c rete. rete. rete. host)
 $224 \rightarrow 11100000 \Rightarrow 2^3 = 8$ sottoreti

classe A (IP) = 1-127 (primo ottetto)

classe B (IP) = 128-191

classe C (IP) = 192-223

classe D (IP) = 224-239

classe E (IP) = 240-255

classe A (MASK) = 255.0.0.0

classe B (MASK) = 255.255.0.0

classe C (MASK) = 255.255.255.0

RIASSUNTO STRATO DI RETE

Forwarding e Routing

Switching Circuit: affidabili ma lente

Packet Switching: veloci ma non affidabili

Virtual Circuit: una via di mezzo tra le due

Router: sono tabella di rete per il routing. Input line card, output line card e switching fabric che fa forwarding tra input e output line card.

3 tipi di commutazione: shared memory, bus, crossbar

Protocollo IP: senza conversione. Stabilisce formato, indirizzamento e instradamento dei pacchetti e se necessario multiplexing e demultiplexing.

- Protocollo ICMP: risolve e comunica errori.

IP identifica un'interfaccia di rete (4 byte - dotted 1 bit ciascuno - max 255.255.255.255)

Classfull: indirizzi divisi in classi (net id + host id)

A: 0 7_net_id 24_hostid : 1 - 126

B: 10 14_net_id 16_hostid : 128 - (9)

C: 110 21_net_id 8_host_id : 192 - 223

Subnetting → Net id + (host id + subnet-id) → MASCHERA

Statico: tutte le subnet sono uguali.

Dinamico: diverse lunghezze di subnet mod a secondo di quante sottorete mi servono

Routing: se sto nello stesso rete: mac → mac

se sto su reti diverse: mac → default router → IP destinazione

Se l'IP destinazione è una sottorete della rete a cui è connesso il router il pacchetto viene inviato verso il mac, altrimenti routing table → next hop.

Routing table: Dest IP, IP next-hop, porta, m/lo varie.

Supernetting: opposto subnetting. Unisce più indirizzi in un'unica rete individuata dallo stesso prefisso

Instradamento: centralizzato → I comandi sono gestiti da un unico router
che se si rompe oscura tutto

Distribuito: tutti i router si scambiano info sulle tabella di routing

Routing flat → tutti router uguali

Routing gerarchico → divisione in domini, sistemi auto nomi ecc.

Instradamento statico → Deciso all'inizio manualmente

Instradamento Dinamico → Deciso tramite algoritmi

Protocolli Instradamento

IGP → Interior gateway P.

EGP → Exterior Gateway P.

IGP < Routing Information P.

○ SPF.

RIP → Bellman Ford

Request → Distancia Vettore → Response.

IP dest. + mask + next hop + metrica = Request / Response

OSPF: Dijkstra → Link State P.

In flooding in maneggio LSA viene inviato con le info dei router vicini

DHCP: dynamic Host Configuration P. = configurazione automatica della rete.

HOST → DHCP discover (Broadcast)

ROUTER → DHCP offer (Broadcast) invia l'IP

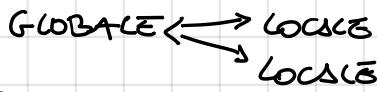
HOST → DHCP request (Broadcast) host richiede la configurazione

SERVER → DHCP ACK (Broadcast) invia la conf.

Conf. Manuale, Dinamico e Automatico.

NAT: dà un'indirizzo pubblico agli indirizzi privati solo quando devono

comunicare con l'esterno



Incompatibilità con ICMP e P2P.

STRATO DI TRASPORTO

Le entità che offrono i servizi dello strato di trasporto si trovano solamente nella sorgente e nella destinazione e non nei nodi tra esse.

DEMULTIPLEXING (in entrata) e **MULTIPLEXING** (in uscita) ovvero assemblamento e disassemblamento di tutti i dati dalle e verso le rispettive porte

# porta origine	# porta dest.
IP origine	IP destinazione
DATO	

Struttura segmento TCP/UDP

Demultiplexing SENZA commessione (UDP)

(# porta origine, # porta dest.) ← STRUTTURA SOCKET

Quando l'host riceve il segmento UDP controlla il # di porta e invia il segmento allo socket corrispondente

Pacchetti IP con indirizzi IP e/o # di porta di origine diversi sono inviati allo stesso socket.

Demultiplexing CON commessione (TCP)

(IP origine, # porta origine, IP dest., # porta dest.) ← Struttura Socket

La differenza con UDP è che in questo caso la commessione garantisce il controllo di errori, gestione di flusso, controllo sui pacchetti ecc.

INDIRIZZAMENTO

- **STATICO**: le applicazioni più diffusi hanno delle porte specifiche (Well Known port numbers), da 0 a 1023. Elenco gestito da Iana.org in tempo reale
- **DINAMICO**: identificativi assegnati al momento della commessione del sistema operativo, maggiori di 1023.

USER DATAGRAM PROTOCOL (UDP)

È un protocollo di trasporto semplice che non ha controlli sui dati inviati e/o ricevuti e per questo possono essere persi o consegnati fuori sequenza.

È un protocollo senza commessione, quindi senza HANDSHAKING tra mittente e destinatario, e ogni segmento UDP è gestito indipendentemente dagli altri.

È utilizzato spesso nelle applicazioni multimediali che prevedono piccole perdite.

TRANSPORT CONTROL PROTOCOL (TCP)

FUNZIONI:

indirizzamento
controllo di sequenza
controllo e recupero di errore
controllo di flusso
controllo di congestione

Interpreta il flusso di dati proveniente dallo strato applicativo come sequenza di oggetti

È un protocollo con connessione. Nella fase di instaurazione della connessione le due entità TCP remote si sincronizzano scambiandosi: i dettagli dello socket (port, IP address), il proprio numero di sequenza iniziale che rappresenta il numero a partire dal quale tutti gli oggetti trasmessi saranno sequenzialmente numerati, il valore iniziale dello finestra di ricezione.

HANDSHAKE a 3 vie

- 1) A invia un segmento **SYN** a B che specifica il numero di sequenza iniziale e nessun dato.
- 2) B riceve SYN e risponde con **SYN ACK** allocando i buffer e specificando il numero di sequenza iniziale del retour.
- 3) A riceve SYN ACK e risponde con **ACK** che può contenere dati.

N.B. Con lo stesso SYN è possibile indicare anche la massima dimensione del campo dei dati utenti (Maximum Segment Size - MSS) a cui l'entità ricevente risponde a sua volta: in casi di connessioni bidirezionali le dimensioni possono essere differenti.

In modo simile funziona la chiusura della connessione tra i due host:

A invia un segmento **FIN** a B che risponde con ACK, chiude la connessione e invia un ulteriore FIN. A ricevere il FIN e risponde con ACK ritrovando un timer. Rispondendo con ACK ai FIN ricevuti. B riceve ACK e la connessione viene chiusa.

CONTROLLO DI SEQUENZA E DI ERRORE

La gestione dei segmenti fuori sequenza non è specificata dallo standard e dipende dall'implementazione.

Il controllo di errore invece si basa su 3 protocolli: codifica a rivelazione d'errore, riscontri positivi e Retransmission Timeout (RTO).

La codifica è effettuata dall'entità TCP emittente e il risultato è inserito nel checksum e poi utilizzata dall'entità TCP ricevente per trovare eventuali errori.

I riscontri positivi detti ACK possono essere molti da dall'entità TCP ricevente con segmenti vuoti (senza dati) [**IMMEDIATE**], in modalità **piggybacking** ovvero il ricevente aspetta di avere un messaggio da inviare e lo invia con esso, ma per evitare ritardi troppo alti attiva un timer al cui scadere invia l'ACK ufficialmente [**COMPLETINO**].

L'RTO è un timer attivato dall'entità emettente che viene attivato quando un segmento è inviato su una connessione ascendente, disattivato quando si riceve un ACK relativo al segmento corrispondente e se la ricezione avviene prima che l'RTO stesso si esaurisca. L'RTO è dinamico e portando dal suo valore massimo viene poi ricalcolato dinamicamente dopo ogni invio per fornire una media aggiungendo un margine d'errore.

Gli ACK possono essere emessi secondo due modalità:

- IMMEDIATA: appena vengono accettati i dati emette immediatamente un segmento vuoto che contiene l'appropriato numero di riscontro
- CUMULATIVA: appena vengono accettati i dati tiene in memoria il numero di riscontro e lo invia insieme al prossimo segmento in uscita. In questo caso viene attivato un timer: se scade, il riscontro viene inviato anche da solo.

CONTROLLO DI FLUSSO

Questo controllo ha lo scopo di limitare il ritmo di emissione dei dati da parte di un host per evitare la saturazione dello spazio del buffer di ricezione. TCP in particolare utilizza un controllo di flusso basato su una finestra scorrevole di larghezza variabile (scorrimento e larghezza controllati dall'entità ricevente). Tale controllo opera a livello di oggetti (byte) numerati in sequenza a partire dal numero scelto durante la procedura di instaurazione della connessione.

SN = Sequence Number si riferisce al primo oggetto contenuto nel segmento

ACKN = ACK Number si riferisce al prossimo oggetto de l'entità ricevente sta aspettando

RECWINDOW = Window si riferisce al numero massimo di oggetti che l'entità emittente può inviare consecutivamente senza ricevere riscontro per alcuno di questi.

Un riscontro (ACKN = x e RECWINDOW = w) vuol dire che sono rincontrati tutti gli oggetti ricevuti fino a quello numerato con $x-1$ e l'entità TCP emittente può inviare altri w oggetti fino a un totale di $x+w-1$

CONTROLLO DI CONGESTIONE

A differenza del controllo di flusso da riguardo il ricevente, il controllo di congestione riguarda la rete (pacchetti persi, elevati ritardi)

C.G. PUNTO-PUNTO: nessun supporto esplicito dalla rete e la congestione è decelta osservando le perdite e i ritardi nei sistemi terminali (Utilizzato da TCP)

C.G. ASSISTITO DALLA RETE: i router forniscono in feedback ai sistemi terminali tramite un singolo bit per indicare la congestione e comunicare in modo esplicito al mittente la frequenza trasmissiva.

Il TCP utilizza 3 meccanismi:

- 1) l'esaurimento dell'RTO come sintomo di congestione
- 2) la finestra di congestione
- 3) la soglia

la finestra di congestione (CONGESTION WINDOW - COGWIN) è utilizzata insieme alla finestra di ricezione e aggiunge un'ulteriore limitazione alla quantità di traffico che un host può inviare in una connessione

La soglia detta THRESHOLD è un valore pari alla metà del valore di Cogwin al momento in cui viene rilevata una perdita: all'inizio della connessione (slow start) la soglia è impostata a ∞

L'entità emittente determina nel tempo il valore della finestra disponibile (Available Window - Awnd) che è uguale al numero di segmenti di lunghezza massima (MSS) che possono essere inviati senza riscontro. Tale valore non deve superare il minimo tra le lunghezze Cogwin e lo RecWindow

$$Awnd = \min \{ Cogwin, RecWindow \}$$

Cogwin e RecWindow sono quantità espresse in numero di segmenti MSS.

RecWindow è la lunghezza comunicata nell'ultimo ACK ricevuto · ottenuto dall'entità TCP emittente dividendo il numero contenuto nel campo Window di questo ACK per il numero di octetti che compongono una MSS

ADDITIVE-INCREASE (Incremento additivo): aumenta il valore di Cogwin di 1 MSS a ogni RTT in assenza di eventi di perdita

MULTIPLICATIVE-DECREASE (Decremento moltiplicativo): dimezza il Cogwin dopo un evento di perdita

Approssimativamente il rate di emissione dei segmenti è dato da:

$$\text{FREQUENZA D'INVIO} = \frac{\text{COGWIN}}{\text{RTT}} \text{ byte/sec}$$

L'entità TCP emittente percepisce la congestione se si esaurisce il timeout o riceve 3 ACK duplicati. Riduce la frequenza d'invio (Cogwin) dopo un evento di perdita.

Per evitare la congestione l'emettitore TCP segue una **procedura ciclica** in due fasi:

1) SLOW START

Quando si stabilisce la connessione avremo COGWIN = 1 MSS e SOGLIA = ∞ . La frequenza, probabilmente molto minore della banda disponibile, aumenta in modo esponenziale finché non si verifica un evento di perdita. Quando ciò accade si pone COGWIN(NEW) = 1 MSS e SOGLIA = COGWIN(OLD)/2

2) CONGESTION AVOIDANCE

Se l'aumento che si ha nella fase precedente raggiunge e supera il valore di soglia, e cioè se $\text{COGWIN} \geq \text{SOGLIA}$, l'incremento di COGWIN diventa **lineare** al crescere di RTT (1 MSS per ogni RTT)

In conclusione, se si trova la fase di Slow Start, un'entità TCP emittente

- 1) incrementa di 1 segmento MSS per ogni RTT il Cwdn quando il percorso di rete non è congestionato
- 2) se invece è congestionato lo diminuisce di un fattore 2 per ogni RTT

RiASSUNTO Strato Transporto

Multiplexing da lato emittente: divide il dato in vari pacchetti IP

Demultiplexing da lato ricevente: invia i vari pacchetti IP alle varie socket

Senza connessione

Pacchetto UDP = Porta origine + Porta destinazione

Con connessione

Pacchetto UDP = indirizzo IP + Porta (origine e destinazione)

Un host dovrà però supportare più socket TCP contemporaneamente.

I servizi web hanno socket differenti per ogni connessione client.

Le opp più diffuse hanno sempre le stesse porte (indirizzamento statico) altre hanno porte assegnate dal sistema (indirizzamento dinamico)

Trasporto SEGMENTI che possono essere persi o corrotti fuori sequenza (UDP)
Non utilizza connessione o controllo gestione

TCP

Protocollo con connessione; indirizzamento, controllo di sequenza, controllo e recupero d'errore, controllo di flusso, controllo congestione.

Interpreta i dati come sequenza di octetti

ordine 1° byte

Socket origine, Socket destinazione, Numero sequenza, Numero Ack, Flag
di sequenza da si aspetta di ricevere

Flag = ACK: 1 re Ack Number valido

SYN: 1 solo nel primo segmento

FIN: 1 quando sono finiti i dati, do trasmettere

Handshake 3 vie: Host \rightarrow SYN \rightarrow Host B

Host B \rightarrow SYN ACK \rightarrow Host A

Host A \rightarrow ACK (può contenere dati) \rightarrow Host B

Chiusura Connessione: Host A \rightarrow FIN \rightarrow Host B

Host B \rightarrow ACK \rightarrow Host A

Host B chiude la connessione \rightarrow FIN \rightarrow Host A

Host A \rightarrow ACK per ogni FIN \rightarrow Host B

Controllo Errore: Il ricevente fa il controllo sul checksum e risponde con un ACK
o in piggybacking se invia anche dei dati insieme all'ACK
Per evitare ulteriori ritardi il mittente attiva un timer al cui termine
reinvia i dati se non ha ricevuto l'ACK

Gli ACK possono essere inviati in modalità IMMEDIATA o CUMULATIVA (piggybacking)

IMMEDIATA:

Controllo di Flusso: ho l'obiettivo di non sovraccaricare il ricevente che comminca la disponibilità tramite REC window (max dati possibili da ricevere)

Vengono inviati un numero di ottetti pari alla larghezza della finestra.

Controllo Correttore: si occupa di non sovraccaricare la rete.

Vengono associate un'ulteriore finestra relativa alla rete + timer + soglia (metà CompWin al momento della perdita)

ESERCIZI

Pag 10 n° 2

$$D_{prop} = 5 \mu\text{s}/\text{km}$$

$$D_{centro} = 50 \text{ s}$$

$$d = 2 \text{ km}$$

Così b:

$$[2000 \cdot 5 \mu\text{s}/\text{km}] \cdot 2 + 50 \text{ s} = 70 \mu\text{s}$$

Esercizi es 1

$$\text{SNR} = 24 \text{ dB}$$

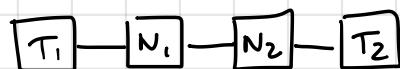
$$\text{Bandiera} = 3 \text{ - } 4 \text{ MHz}$$

$$C = 4 \log_2 (1 + \text{SNR})$$

↑ convertito da $10 \log_{10}(\text{SNR})$

Esercizi es 5

$$L = 100 \text{ byte} = 800 \text{ bit}$$



$$\text{Pacchetto} = 400 \text{ bit} + 100 \text{ bit} = 500 \text{ bit}$$

$$9,5 \text{ ms} \geq 3 \cdot 0,5 \text{ ms} + \frac{800}{x} \Rightarrow 9,5 \times \geq 1,5 \times + 800 \Rightarrow 7 \times \geq 800 \Rightarrow x \geq \frac{800}{7} = 114 \text{ bit/s}$$

Provare intermedia 1 es 1

$$M = F \times L \text{ bit}$$



$$\text{Velocità} = 12 \text{ kbit/s}$$

Store-and-forward

$$D_A = 5 \cdot \frac{F \cdot L}{R}$$

Così b

$$\# \text{ pacchetti} = \frac{M}{L} = F$$

Ogni pacchetto "peso" $L+H$

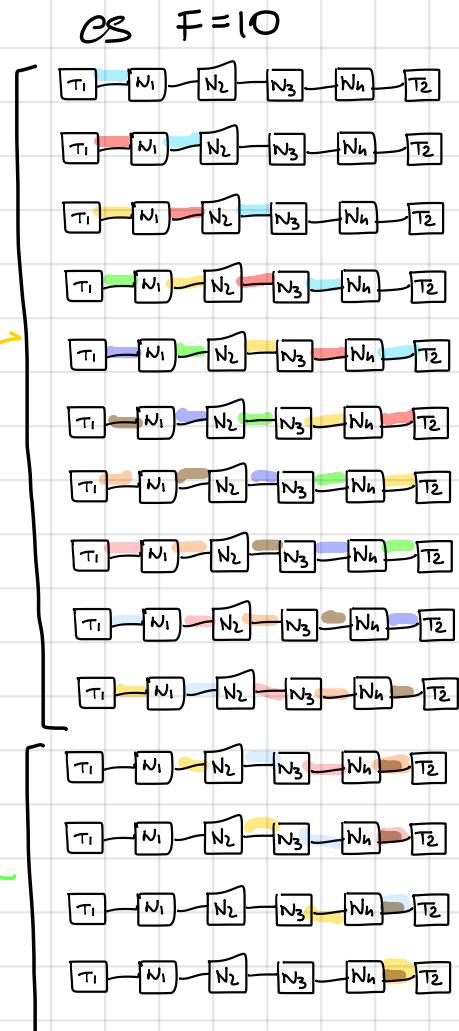
$$F \cdot \frac{L+H}{R}$$

$$4 \cdot \frac{L+H}{R}$$

$$4 = \# \text{ collegamenti} - 1$$

$$\frac{50L}{20} > 10 \frac{(L+10)}{20} + \frac{4(L+10)}{20}$$

$$50L > 14L + 140 \Rightarrow 3,8 > L$$



Libro Esercizi #15

#IT=8 IT=10bit durata trama = 0,5 ms = 0,000005 s

$$1) \text{Capacità complessiva} = \frac{8 \cdot 10}{0,000005} = 160\ 000\ 000 \text{ bit/s} = 160 \text{ Mbit/s}$$

Libro Esercizi #61



$$M = H_p + D_{ip}$$

$$UI = L + H$$

$$D_{ip} + H_p = L \Rightarrow \frac{D}{n} + H_{ip} = L \Rightarrow \text{numero frammenti} = \frac{D}{L - H_{ip}} = 3$$

NB
secondo
di intero
superiore

\hookrightarrow troppo grande
va diviso!

I primi 2 pacetti sono grandi $L - H_{ip} + H_{ip}$ byte = 1960 byte + H_{ip} + H
l'ultimo è quello che rimane e sarà più piccolo \Rightarrow campo utile totale (D) modulo dimensione altri frammenti = 1960 \Rightarrow 5600 mod (1960) = 1280 byte (+ H + H_{ip})

#62

$$PDU1 = H_1(100) + L (\leq 480) \quad] + H_{ip}(20)$$

$$PDU2 = H_2(40) + L_2(60)$$

Dato gommata = 460 (direzione 1); = 100 (direzione 2)

\rightarrow (R) NO FRAGMENTAZIONE perché in direzione 1 si utilizzano 460 byte di gommata e lo PDU1 può arrivare anche a 480 solo di campo utile.

Del router allo switch 2 combina lo PDU e il campo utile al massimo 60

$$\Rightarrow \frac{460 - H_{ip}}{60 - H_{ip}} = \frac{460}{60} = 11 \text{ frammenti}$$

2) efficienza di trasferimento (min[efficienza S1; efficienza S2]) è data da

$$\text{campo utile (460 - } H_{ip}) / \text{informazione totale trasmessa (11} \cdot [H_2 + L_2]) = \boxed{0,4}$$

per PDU2

$\frac{100 - H_{ip}}{60 - H_{ip}} = 2$ se non ricevitore d'intermezzo frammentazione in S1
e IP ha il riemanneggiamento solo a destinazione!

efficienza = $\frac{100 - H_{ip}}{2(100 + 60)} = 0,25$ WB. È stato utilizzato $L_2(60)$ poiché L_1 è variabile e in questi casi si lascia quella "di sicurezza".

#65

$$\begin{aligned} \text{Subnet Mask } (/20) &= \underset{\text{IP}}{1111111.1111111.11100000000000000} \\ &= \frac{10010101.10100000.01001011.00000111}{10010101.10100000.01000000.00000000} \\ &\quad |45 \quad 80 \quad 64 \quad 0 \end{aligned}$$

n indica che sono utilizzati n bit a 1 nella subnet mask. Per "calcolare" indirizzo di rete si esegue l' ANDing tra subnet mask e IP entrambi in binario

In questo caso si è andati a tassazioni fra /20, /24, /23.

Esercizi #10

$$H_{IP} + D_{IP} \rightarrow L] + H$$

$$H_{IP} = 40$$

$$D_{IP} = 8000$$

$$L = 1500$$

$$H = 60$$

$$\frac{D}{n} + H_{IP} = L \Rightarrow n = \frac{D}{L - H_{IP}} = \frac{8000}{1500 - 40} = 6$$

D/n	H _{IP}	H
-----	-----------------	---

$$L - H_{IP} + H_{IP} + H \Rightarrow 1560 \text{ dimensione compreso } H$$

L'ultimo pacchetto compreso di entrambi gli header pesa 800bit
 efficienza = $\frac{\text{comp. dati}}{\text{dati complessivi}} = \frac{8000}{8000 + 6(H_{IP} + H)} = \frac{8000}{8600} = 0,9$

Esercizi #12



FILE = 24000 bit

$$1500 = 01 \text{ 20 bit} = H$$

$$\frac{24000}{1500-20} = 17 \text{ pacchetti}$$

$$H_{tot} = 17 \cdot 20 = 340$$

$$\therefore H = \frac{34000}{24340} = 1.40\%$$

$$\left[\frac{16 \cdot 1500}{R} \right] + \frac{340}{R} + 17 = 240 + 3,4 + 17 = 260 \text{ s}$$

NUERO SOTTORETI = 2^N dove N = # uni del numero in binario dell'host nello maschera
es. 255.255.255.224 (classe C rete. rete. rete. host)
 $224 \rightarrow 11100000 \Rightarrow 2^3 = 8$ sottoreti

classe A (IP) = 1-127 (primo ottetto)

classe B (IP) = 128-191

classe C (IP) = 192-223

classe D (IP) = 224-239

classe E (IP) = 240-255

classe A (MASK) = 255.0.0.0

classe B (MASK) = 255.255.0.0

classe C (MASK) = 255.255.255.0

EFFAERA: compo dati
dati inviati complessivi (compresi Header)

host = 2^{32-N} dove N è il numero di zero nello subnetmask

NB. 2 sono rinvolti e sono spesso eliminati (dipende dai casi)

RITMO BINARIO MEDIO

$$RBM = \frac{\text{bit totali in un dato periodo}}{\text{periodo}}$$

GRADO DI INTEGRITENZA

$$G = \frac{\text{ritmo binario di picco}}{\text{ritmo binario medio}}$$

RITMO BINARIO PICCO = massimo bit rate nell'intervallo

Esame 12/01/16

2) capacità disponibile \times % utilizzo = capacità massima
 $2150 \cdot 65\% = 1397,5 \text{ kbit/s}$

$$\text{Coefficiente utilizzo} = \frac{\text{rete reale}}{\text{rete di picco}}$$

flussi \times rete reale = efficienza \times capacità disponibile = capacità massima

$$\text{Coefficiente utilizzazione} = \frac{\# \text{ flussi trasferiti}}{\# \text{ flussi da trasferire}}$$

indirizzo di rete = xxx.xxx.xxx.

$$\text{es. } 12 \text{ host} \Rightarrow 2^x \geq 12$$

indirizzo = xxx.xxx.xxx.0

indirizzo = xxx.xxx.xxx.2^x

(1)
2

$$\text{maschera} = /32 - x$$

$$\text{efficienza} = \frac{\text{host}}{2^x - 1}$$

Esercizio 12/01/16

#2

SOTTORETE	INDIRIZZO RETE	MASCHERA (1...)	EFFICIENZA
NET - 1	192.70.20.0	128	0,75
NET - 2	192.70.20.16	125	0,78
NET - 3	192.70.20.144	127	0,93
NET - 4	192.70.20.176	129	0,62

Maschera net-1: host = 12 $\Rightarrow 2^4 = 16 > 12 \Rightarrow$ maschera /32 - 4 = /28

$$\text{efficienza net-1} = \frac{12}{16-1} = \frac{12}{15} = 0,8$$

NB Tengo un host dedicato al broadcast

indirizzo net-2 = indirizzo net-1 " + 16 host " = 192.70.20.16

Maschera net-2 : host = 100 $\Rightarrow 2^7 = 128 > 100 \Rightarrow$ maschera /32 - 7 = /25

$$\text{efficienza net-2} = \frac{100}{127} = 0,78$$

• • • •

LINK	INDIRIZZO LINK	MASCHERA
A - B		
B - C		
C - D		
D - A		

Esercizio #13 (Cuomo)

Trama = 125 μs divisa in 24 IT, 23 utilizzabili

1IT = 8 bit (7 dati + 1 segnalazione)

$$\text{CAPACITÀ COMPRESSIVA} = \frac{\text{bit trasmessi}}{\text{durata trama}} = \frac{24 \cdot 8}{125 \cdot 10^{-6}} = \frac{192}{0,000125} = 1536 \text{ Kbit/s}$$

$$\text{CAPACITÀ SUB-CANALE} = \frac{8}{0,000125} = 64 \text{ Kbit/s}$$

$$\hookrightarrow \text{Segnalazione} = \frac{1}{0,000125} = 8 \text{ bit/s}$$

$$\hookrightarrow \text{Simocomm.} \uparrow$$

Esercizio #14 (Cuomo)

$$\text{Capacità subcanale} = \frac{19}{0,5 \mu} = 28 \text{ Kbit/s}$$

$$\hookrightarrow \text{Solo voce} = \frac{8}{0,5 \mu} = 16 \text{ Kbit/s}$$

$$\text{Capacità Segnalazione} = \frac{6}{0,5 \mu \cdot 8} = 1,5 \text{ Kbit/s}$$

$$\text{Capacità dati} = 1,5 \text{ Kbit/s} \cdot 7 = 10,5 \text{ Kbit/s}$$

$$\text{Capacità Totale} = \frac{(14 \cdot 24) + 8}{0,5 \mu} = 688 \text{ Kbit/s}$$

Esercizio #15 (Cuomo)

$$\text{Capacità complessiva} = \frac{8 \cdot 10}{0,5 \cdot 10^{-6}} = 160000 \text{ Kbit/s} = 160 \text{ Mbit/s}$$

$$\text{SOVRACCUPAZIONE} = \left\lceil \frac{\text{ritmo binario sorgente}}{\text{capacità subcanale}} \right\rceil = \# \text{ IT per sovramultiplicazione}$$

$$\Rightarrow \left\lceil \frac{35}{\frac{10}{0,5 \mu}} \right\rceil = 2$$

$$\text{SOTTOCUPAZIONE} = \left\lfloor \frac{\text{capacità subcanale}}{\text{ritmo binario sorgente}} \right\rfloor = \# \text{ terme per sottomultiplicazione}$$

$$\Rightarrow \left\lfloor \frac{\frac{10}{0,5 \mu}}{35} \right\rfloor = 3$$

Esercizio #66 (Giorno)

$$110 + 21 \text{ ret} 10 + 8 \text{ host id}$$

$$7 \text{ sottoreti} \rightarrow 2^3 = 8$$

$$\text{host id } 8 - 3 = 5 \text{ bit}$$

host per sottoret = $2^5 - 2 = 30$ host Not bootano per pc-net

Dopo fare subnetting a lunghezza variabile!

NURE RETE	INDIRIZZO RETE	MASCHERA	BROADCAST	RANGE	HOST
PC - Net	195.168.1.0	$32 - 7 = 25$ ↳ hosts $\leq 2^7$	195.168.1.127	1 - 126	126
Si parte dalla più piccola			2^{7-1}		
WS - Net	195.168.1.128	$32 - 5 = 27$	195.168.1.159	129 - 158	30
X - net	195.168.1.160	27	195.168.1.191	161 - 190	30
X ₂ - net	195.168.1.192	$32 - 4 = 28$	195.168.1.207	193 - 206	14
LINK 1	195.168.1.208	$32 - 2 = 30$	195.168.1.211	209 - 210	2
LINK 2	195.168.1.212	30	195.168.1.215	213 - 214	2
LINK 3	195.168.1.216	30	196.168.1.219	217 - 218	2

Esercizio #67 (Giorno)

$$6 \text{ sottoreti} = 2^3 = 8$$

$$193.212.100.0$$

$$255.255.255.0$$

$$8 - 3 = 5 \text{ zeri} \rightarrow 255.255.255.\underbrace{111\ 00000}_{224}$$

NURE RETE	INDIRIZZO RETE	MASCHERA	BROADCAST	RANGE	HOST
rete 1	193.212.100.0	\27	193.212.100.31	1 - 30	30
rete 2	193.212.100.32	\27	193.212.100.63	33 - 62	30
rete 3	193.212.100.64	\27	193.212.100.95	65 - 94	30
rete 4	193.212.100.96	\27	193.212.100.127	97 - 126	30
rete 5	193.212.100.118	\27	193.212.100.159	129 - 158	30
rete 6	193.212.100.160	\27	193.212.100.191	161 - 190	30

Esercizio #74 (Uomo)

maschera \24

IP 195.169.0.0

Dobbiamo fare 5 sottoreti $\leq 2^3$ bit \rightarrow Come $C = 8$ bit host id $\rightarrow 8-3=5$
 $2^5 = 32$ ma non occorrono minimo 80 host per un A \Rightarrow maschera variabile

name net	IP	maschera	Broadcast	Rango	HOST
eth0(A)	0	$2^7 \Rightarrow 32-7=125$	127	1-126	126
eth1(B)	128	$2^6 \Rightarrow 32-6=126$	191	129-190	62
eth2(C)	192	$32-5=127$	223	193-222	30
eth1(D)	224	\27	255	225-254	30
					195.169.0.225

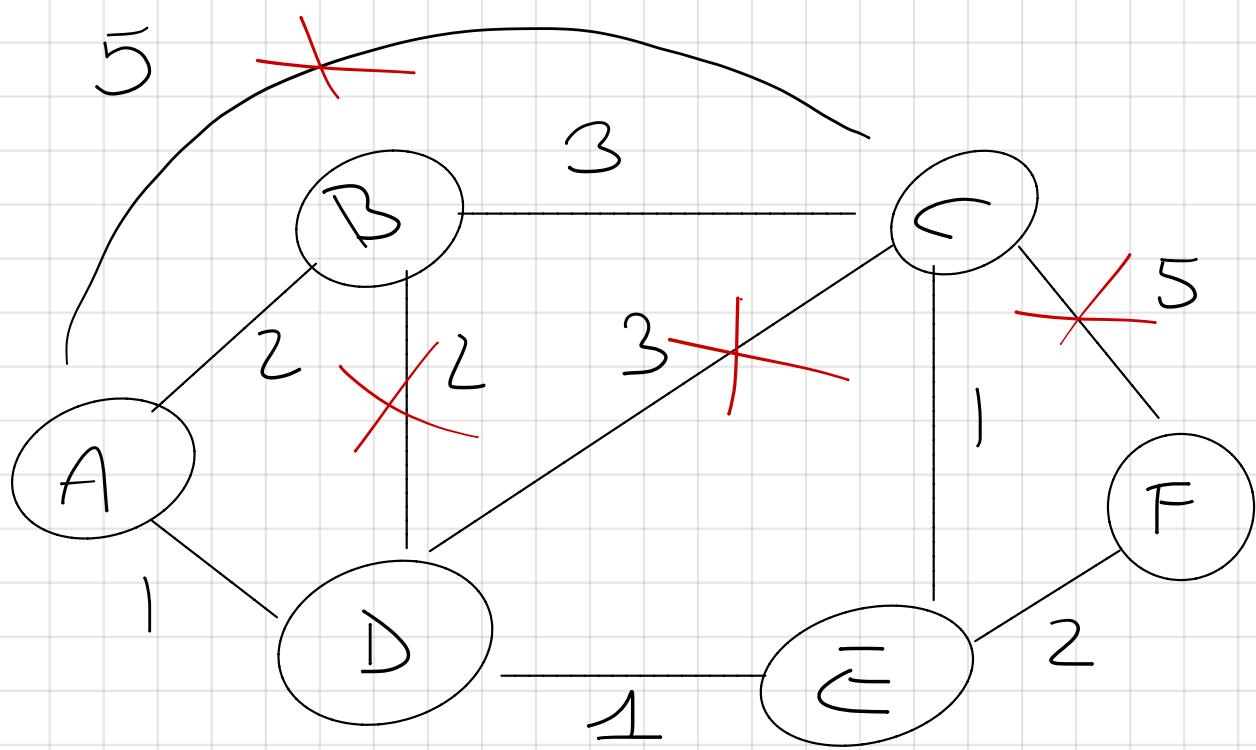
Routing Table

IP Dest.	Gateway	Mask	Interfaccia
195.168.1.0	195.168.1.2	DIR	.240
195.168.0.0	195.168.0.1	DIR	.128
195.169.0.128	195.168.0.129	DIR	.192
195.169.0.192	195.168.0.130	INDIR	.224
195.169.0.224	195.168.0.130	INDIR	.224
195.169.0.192	195.168.0.130	192	eth 1

Mimone

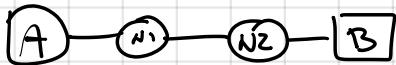
195.169.0.192 si trasforma in binario e conta gli uni
 195.169.0.224 in binario (26) $\rightarrow 126 \rightarrow .192$

0.0.0.0. 195.168.1.7 0.0.0.0. eth2



Passo	Nodi	B	C	D	E	F
0	A	(2, A)	(5, A)	(1, A)	∞	∞
1	A, D	(2, A)	(4, D)	—	(2, D)	∞
2	A, D, B	—	(4, D)	—	(2, D)	∞
3	A, D, B, E	—	(3, E)	—	—	(4, E)
4	A, D, B, E, C	—	—	—	—	(4, E)

Esercizio # P31 (Ross)



$$X = 8 \cdot 10^6 \text{ bit}$$

$$C = 2 \text{ Mbit/s} = 2 \cdot 10^6 \text{ bit/s}$$

a) 4 s, 12 s

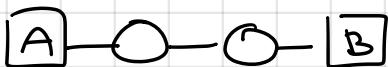
b) # pacchetti = 800

1 pacchetto \rightarrow 5 ms

2° pacchetto \rightarrow 10 ms

$$c) \frac{10000}{2 \cdot 10^6} \cdot 2 + \frac{800 \cdot 10000}{10 \cdot 10^6} = 0,01 + 4 = 4,01 \text{ s}$$

Esercizio # 29 (Levomo)



$$X = 560 \text{ bit}$$

$$C = 64 \text{ Kbit} = 64 \cdot 10^3 \text{ bit/s}$$

$$D_{\text{prop}} = 10 \text{ ms} \times 20 \text{ ms}$$

Circuito $\rightarrow S = \text{tempo istanziazione commessione}, \text{tempo nodo} = 0,5 \text{ ms}$

$$T = 30 \text{ ms} + 1 \text{ ms} + \frac{560}{64 \cdot 1000} + S = 39,75 + S$$

b) $H = 10 \text{ bit}$ $L_{\max} = 50 \text{ bit}$

$$T = 30 \text{ ms} + \frac{60}{64 \text{ K}} \cdot 2 + \frac{\left(560 + \lceil \frac{560}{50} \rceil \cdot 10 \right)}{64 \text{ K}} = 30 + 1,875 + 10,625 = 42,5$$

$$\boxed{S + 39,75 \leq 42,5}$$

Esercizio #15 (Cuomo)

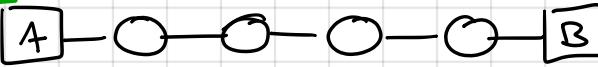
$T_{\text{trans}} = 0,5 \mu\text{s}$ 8 IT ci saranno 10 bit

$$a) \frac{10}{0,5 \cdot 10^{-6}} \cdot 8 = 160 \text{ Kbit}$$

$$b) \left\lceil \frac{\frac{35}{10}}{0,5 \cdot 10^{-6}} \right\rceil = 1,75 \Rightarrow 2$$

Provo Intermedia I #1

$$M = 25L \quad R = 10 \text{ Kbit/s}$$



$$a) D = 5 \frac{M}{R} = 0,0125L$$

$$b) D = \frac{H+L}{R} (N-1) + \left(\left\lceil \frac{M}{L} \right\rceil (H+L) \right) = \frac{15+L}{R} \cdot 4 + \left\lceil \frac{25L}{L} \right\rceil (15+L) = \\ = \frac{60+4L}{R} + \frac{375+25L}{R} = \frac{435}{R} + \frac{29L}{R}$$

$$0,0125L < 0,0435 + 0,0029L \Rightarrow 0,0096L < 0,0435 \Rightarrow L < 4,5 \approx 5$$

Provo Intermedia I #2

$$\# IT = 20 \quad T_{\text{trans}} = 1 \text{ ms} \quad N_{\text{bitIT}} = 500 \quad R = 650 \text{ Kbit}$$

$$\text{Capacità tot} = \frac{20 \cdot 500}{1 \cdot 10^{-3}} = 10 \text{ Mbit}$$

$$W_s = 4 \text{ kHz} \rightarrow \text{PCM 6 bit}$$

$$\text{bit rate min} = 6 \cdot 8000 = \underline{\underline{48 \text{ kbit/s}}}$$

Durata blocco = 12 ms

$$F_c = 44100 \text{ Hz} \quad \text{bit campione} = 16 \quad W_c = 1,5 \text{ MHz}$$

durata trasferimento \leq 1 s

$$\text{bit file} = 16 \cdot 12 \cdot 44100 = 8,5 \text{ Mbit}$$

Esercitazione II #1

Bordo traslato

$$\text{Canale 1} = 5 \text{ MHz} \rightarrow 4 \text{ QAM}$$

$$\text{Canale 2} = 10 \text{ MHz} \rightarrow 16 \text{ QAM}$$

$$\text{Capacità 1} \Rightarrow 4 = 2^m \Rightarrow m = 2 \text{ bit a livello} = 2 \cdot 5 \text{ MHz} = 10 \text{ Mbit/s}$$

$$\text{Capacità 2} \Rightarrow m = 4 \Rightarrow 4 \cdot 10 \text{ MHz} = 40 \text{ Mbit/s}$$

$$R_p = 500 \text{ kbit/s} \quad \alpha = 0,25$$

coefficiente utilizzazione $\leq 0,75 = \nu$

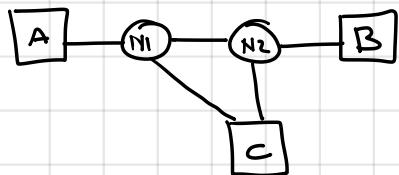
$$R_m = \alpha \cdot R_p = 125 \text{ kbit/s}$$

$$NR_k \leq 0,75$$

$$N_1 \leq \frac{0,75 \cdot 10 \text{ Mbit/s}}{125 \text{ kbit/s}} = 60$$

$$N_2 \leq \frac{0,75 \cdot 40 \text{ Mbit/s}}{125 \text{ kbit/s}} = 240$$

Esercizio #1



$$U = 0,65$$

A genera 34 flussi

$$R_p = 200 \text{ kbit/s}$$

$$\lambda = 0,2$$

24 flussi da A a B e 10 da A a C

	CAPACITÀ DISPONIBILE	# FLUSSI	COEFFICIENTE UTILIZZO
C1	2150 1397,5	34	0,97
C2	1710 1111,5	27	0,97
C3	1500 975	24	0,98
C4	640 416	10	0
C5	640 416	10	0,96

$$R_m = 0,2 \cdot 200 = 40 \text{ kbit/s}$$

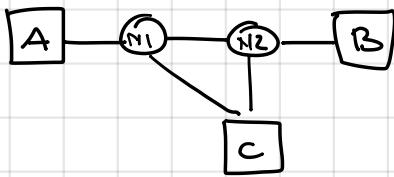
flussi C1 : $N \leq 1397,5 \Rightarrow N \leq 34$

flussi C2 : $N \leq 27$

flussi C3 : $N \leq 24$

flussi C4, C5 : $N \leq 10$

Esercizio #18 (luomo)



$$C_1 = 1 \text{ Mbit/s} = 1000 \text{kbit/s}$$

$$C_2 = 600 \text{ kbit/s}$$

$$C_3 = 500 \text{ kbit/s}$$

$$C_4 - C_5 = 300 \text{ kbit/s}$$

$$R_p = 600 \text{ kbit/s}$$

$$A = 3 \text{ km/s} \quad (2 \text{ a } C)$$

$$\text{coefficiente attivato} = 0,4$$

$$\text{coefficiente utilizzazione} = 0,85$$

$A - N_1 \Rightarrow 3 \cdot 240 \leq 1000 \cdot 0,85 \Rightarrow \text{OK} \quad \text{possono 3 flussi}$

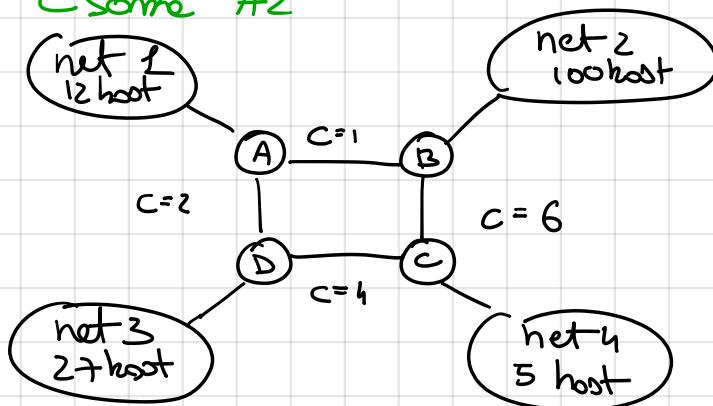
$N_1 - C \Rightarrow 2 \cdot 240 \leq 255 \Rightarrow \text{No massimo 1}$

$N_1 - N_2 \Rightarrow 2 \cdot 240 \leq 510 \Rightarrow \text{OK} \quad \text{possono 2 flussi}$

$N_2 - B \Rightarrow 1 \cdot 240 \leq 425 \Rightarrow \text{OK}$

$N_2 - C \Rightarrow 1 \cdot 240 \leq 155 \Rightarrow \text{OK}$

Esercizio #2



192.170.20.0

SOTTORETE	INDIRIZZO RETE	MASCHERA	EFFICIENZA	LINK	INDIRIZZO LINK	MASCHERA
net 1	.0	\28	0,85	A-B	.16	\30
net 2	.20	\25	0,79	B-C	.148	\30
net 3	.152	\27	0,96	C-D	.184	\30
net 4	.188	\25	0,83	D-A	.196	\30

INDIRIZZO RETE	MASCHERA	NEXT-HOP	INTERFACCIA USATA	DIRETTO / INDIRETTO
.152	\27	.152	D-net 3	DIRETTO
.0	\28	.196	D-A	INDIRETTO
.188	\29	.184	D-C	INDIRETTO
.20	\25	.196	D-A	INDIRETTO

Esercizio #74 (Cuomo)

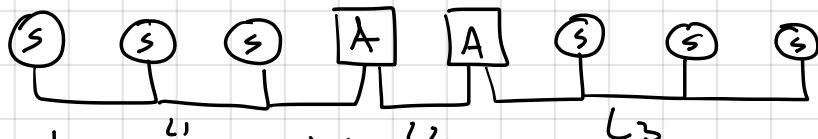
NAME	INDIRIZZO RETE	MASK	BROADCAST	RANGE	HOST
LAN A	.0	\25	.127	1-126	126
LAN B	.128	\26	.191	.129-190	62
LAN C	.192	\27	.223	.193-222	30
LAND	.224	\27	.255	.223-254	30

2)

DESTINAZIONE	GATEWAY	MASK	INTERFACE
195.168.0.0	1 diretto	.128	eth0
.128	.129 diretto	.192	eth1
.192	.130 indiretto	.224	eth1
.224	.130 indiretto	.224	eth1
0.0.0.0	1.7 indiretto	0.0.0.0	eth2
1.0	1.2 diretto	.240	eth2

SuperNetworking perché gateway mascherata:
 $192 \rightarrow$ binario
 $224 \rightarrow$ binario
XOR esclusivo

Esercizio #43 (Cuomo)



$$\text{dimensione frame} = F = 128 \text{ byte}$$

$$\text{capacità trasferimento} = R = 10 \text{ Mbit/s}$$

$$L_1 = 100 \text{ m} \quad L_2 = 300 \quad L_3 = 100$$

$$\text{Velocità prop.} = 10 \cdot 10^6$$

$$\text{Efficienza Hub} = \frac{\frac{1}{e} \cdot \frac{1074}{10 \text{Mbit}}}{\frac{1}{e} \cdot \frac{1024}{10 \text{Mbit}} + \frac{1000}{10 \cdot 10^6}} = \frac{0,000037}{0,000037 + 0,0001} = \frac{0,000037}{0,000137} = 0,27$$

$$\text{Efficienza Switch} = \frac{0,000037}{0,000037 + 0,00006} = \frac{0,000037}{0,000097} = 0,38$$

Esercizio #10 (Guomo)

$$R = 3800 \text{ bit}$$

$$L_{UDP} = 400 \text{ bit} \quad H_{UDP} = 20 \text{ bit}$$

$$L_{IP} = 420 \text{ bit} \quad H_{IP} = 40 \text{ bit}$$

$$L_2 = 250 \text{ bit} \quad H_2 = 10 \text{ bit}$$

5 IT ognuno do 10 ms $1 \text{ IT} = 1 \text{ UI}$ 4 IT utilizzabili

$$UDP = \lceil \frac{3800}{400} \rceil = 10 \quad 9 \text{ pacchetti do } 400 + 20 + 1 \text{ do } 200 + 20$$

$$IP = 9 \text{ pacchetti do } 420 + 40 + 1 \text{ do } 220 + 40$$

$$\lceil \frac{420 + 40}{250} \rceil = 2 \quad \lceil \frac{420 + 24}{250} \rceil = 2 \times 9$$

$$\lceil \frac{220 + 40}{250} \rceil = 2 \quad \lceil \frac{200 + 24}{250} \rceil = 2 \Rightarrow 1 + 50$$

$$20 \cdot 10 + 40 \cdot 10 + 200 =$$