Resumos RCOM

MIEIC

Janeiro 2019

Conteúdo

1 Physical Layer

1.1 Transmitting information

$$C = 2B \log_2 M \tag{1}$$

C
channel capacity

B
bandwith

2B
baudrate in symbol/s or baud

M
levels used to encode information

1.2 Types of modulations

- Binary signal
- Amplitude modulation
- Frequency modulation
- Phase modulation
- Quadrature Amplitude Modulation (M - QAM)

1.3 Shannon's Law

The maximum theoretical capacity of a channel (bit/s) is given by the following expressions:

$$SNR = \frac{P_r}{N_0 B_c} \tag{2}$$

$$C = B_c \log_2(1 + SNR) \tag{3}$$

SNR
: signal to noise ratio

B_c
: bandwidth of the channel (Hz)

P_r
: signal power as seen by receiver (W)

N_0
: White noise; noise power per unit bandwidth (W/Hz)

1.4 Free space loss

$$\frac{P_t}{P_r} = \frac{(4\lambda f d)^2}{c^2} \tag{4}$$

P_t
: signal power at transmitting antenna

P_r
: signal power at receiving antenna

λ
: carrier wavelength

d
: propagation distance between antennas

c
: speed of light $3 * 10^8$ m/s

1.5 Solved Exam Problems

2018R - 1

- 16 QAM
- bitrate (C) = 8kbit/s
- baudrate (2B) = ?

$$C = 2B \log_2 M$$

$$8 = 2B \log_2 16$$

$$8 = 2B * 4$$

$$2B = 2$$

2018N - 1

- 8PSK
- baudrate (2B) = 250 kbaud
- bitrate (C) = ?

$$C = 2B \log_2 M$$

$$C = 250 \log_2 8$$

$$C = 250 * 3$$

$$C = 750$$

2017N - 2

- baudrate (2B) = 100 kbaud
- bitrate (C) = 300 kbit/s
- phase modulation
- n^{o} of phases = ?

$$C = 2B \log_2 M$$

$$300 = 100 \log_2 M$$

$$3 = log_2 M$$

$$M = 2^3 = 8$$

2017N - 3

$$\frac{P_t}{P_r} = \frac{(4\lambda f d)^2}{c^2}$$

$$P_r = \frac{P_t}{\frac{(4\lambda f d)^2}{c^2}}$$

$$P_r = \frac{P_t c^2}{(4\lambda f d)^2}$$

$$SNR = \frac{P_r}{N_0 B}$$

Quanto maior d, menor P_r, e quanto menor P_r, menor SNR. Quanto maior B, menor SNR, logo menor a eficiência.

2016R - 1

- baudrate (2B) = 80 kbaud
- bitrate (C) = 320 kbit/s
- phase modulation
- nº of phases = ?

$$C = 2B \log_2 M$$

$$320 = 80 \log_2 M$$

$$4 = log_2 M$$

$$M = 2^4 = 16$$

2016N - 2

- 16 QAM
- bitrate (C) = 100kbit/s
- baudrate (2B) = ?

$$C = 2B \log_2 M$$

$$100 = 2B \log_2 16$$

$$100 = 2B * 4$$

$$2B = 25$$

2016N - 3 Canal rádio com propagação em espaço livre.

$$\frac{P_t}{P_r} = \frac{(4\lambda f d)^2}{c^2}$$

$$P_r = \frac{P_t}{\frac{(4\lambda f d)^2}{c^2}}$$

$$P_r = \frac{P_t c^2}{(4\lambda f d)^2}$$

$$SNR = \frac{P_r}{N_0 B}$$

$$C = B_c \log_2(1 + SNR)$$

Quanto menor a distância e frequência, maior P_r. Quanto maior P_r, maior SNR, logo maior a capacidade.

2015 - 2

- 2 ligações sem fios
- Pt1 = Pt2 (potência transmitida pelo emissor)
- B1 = B2 (largura de banda do canal)
- d1 < d2 (distância entre o emissor e o recetor)
- relação entre P e C das ligações?

$$\frac{P_t}{P_r} = \frac{(4\lambda f d)^2}{c^2}$$

$$P_t = P_r \frac{(4\lambda f d)^2}{c^2}$$

De Pt1 = Pt2,

$$P_r 1 * (4\lambda f d_1)^2 = P_r 2 * (4\lambda f d_2)^2$$

Como d1 < d2, P_r1 > P_r2, então C1 > C2

$$C_1 = B_1 \log_2(1 + \frac{P_r 1}{N_0 B_1})$$

$$C_2 = B_1 \log_2(1 + \frac{P_r 2}{N_0 B_1})$$

2014N - 1

- baudrate (2B) = 8 kbaud
- bitrate (C) = 32 kbit/s
- bandwith (B) = 4 kHz
- M = ?

$$C = 2B \log_2 M$$

$$32 = 8 \log_2 M$$

$$4 = \log_2 M$$

$$M = 16$$

2013N - 1

- bandwith (B) = 1 MHz
- baudrate (2B) = 2 MHz
- SNR = 40 dB
- 8 level impulses => M = 8
- bitrate (C) = ?

$$C = 2B \log_2 M$$

$$C = 2 \log_2 8$$

$$C = 2 * 3 = 6$$

2012N - 1

- 4 QAM
- baudrate (2B) = 100 kbaud
- bitrate (C) = ?

$$C = 100 \log_2 4$$

$$C = 100 * 2$$

$$C = 200$$

2011N - 2 Num canal sem fios, potência recebida é tanto maior quanto menor for a distância emissor-recetor e o comprimento de onda da portadora.

$$\frac{P_t}{P_r} = \frac{(4\lambda f d)^2}{c^2}$$

$$P_r = \frac{P_t}{\frac{(4\lambda f d)^2}{c^2}}$$

$$P_r = \frac{P_t c^2}{(4\lambda f d)^2}$$

2 Data Link Layer

2.1 Data Link layer functions and services

2.1.1 Main functions

- Provide service interface to the network layer.
- Eliminate/reduce transmission errors.
- Regulate data flow: Slow receivers not swamped by fast senders.

2.1.2 Services provided

Principal service: Transfer data from the network layer on the source machine to the network layer on the destination machine.

There are three reasonable possibilities that we will consider:

- **Unacknowledged connectionless service:**
  - No logical connection is established beforehand or released afterwards.
  - Transmitter sends independent frames without having the destination machine acknowledge them.
  - If a frame is lost due to noise on the line, no attempt is made to detect or recover from that loss.
  - Appropriate when the error rate is very low and for real-time traffic.
- **Acknowledged connectionless service:**
  - No logical connections used.
  - Each frame sent is individually acknowledged so the sender knows if a frame arrived correctly or has been lost.
  - If it has not arrived within a specified time interval, it can be sent again.

  - This service is useful over unreliable channels, such as wireless systems.(i.e. Wi-Fi).

- **Acknowledged connection-oriented service:**
  - The source and destination machines establish a connection before any data are transferred.
  - Each frame is numbered, and the data link layer guarantees that each frame sent is indeed received.
  - Guarantees that each frame is received exactly once and that all frames are received in the right order.
  - Appropriate over long, unreliable links (satellite channel, long-distance telephone circuit).
  - Divided in 3 phases:
    * **First phase:** The connection is established (initialize variables and counters needed to keep track of which frames have been received and which ones have not).
    * **Second phase:** One or more frames are actually transmitted.
    * **Third phase:** The connection is released (free the variables, buffers, and other resources used to maintain the connection).

2.2 Framing

Breaking up the bit stream into discrete frames, computing a short token called a checksum for each frame, and including the checksum in the frame when it is transmitted. When a frame arrives at the destination,the checksum is recomputed. If it is different from the one contained in the frame, the data link layer knows that an error occurred.

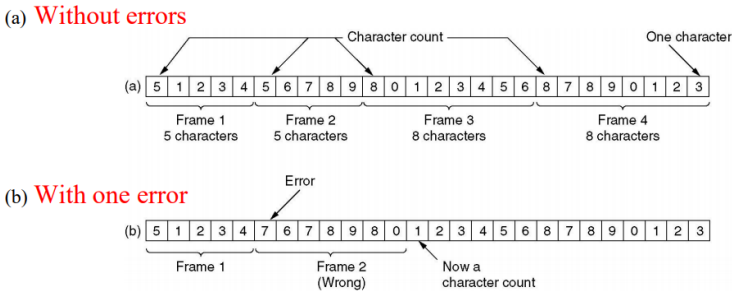
A good design must make it easy for a receiver to find the start of new frames while using little of the channel bandwidth. We will look at three methods:

2.2.1 Byte count

Uses a field in the header to specify the number of bytes in the frame. When the data link layer at the destination sees the byte count, it knows how many bytes follow and hence where the end of the frame is.

Issues

- The count can be garbled by a transmission error.
- A single bit flip, may trigger the destination to get out of synchronization.
- If an out-of-sync occurs it is unable to locate the correct start of the next frame.



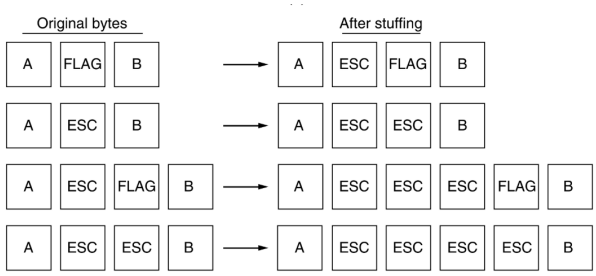
2.2.2 Flag bytes with byte stuffing

This method gets around the problem of resynchronization by having each frame start and end with special bytes (flag bytes).

However, this flag may occur in the middle of the data and induce the receiver in error by thinking the end of the frame was reached. This issue can be solved using **byte stuffing**.

Byte Stuffing

- Inserting a special escape byte (ESC) before each flag byte in the data.
- Makes framing flag bytes distinguishable from the ones in the data.
- Escape bytes present in the data also need to be escaped.



2.2.3 Flag bits with bit stuffing

- Each frame begins and ends with a special bit pattern (01111110 / 0x7E).
- When the sender finds five consecutive 1 bits in the data, it stuffs a 0 bit into the outgoing bit stream.
- When the receiver finds five consecutive incoming 1 bits, followed by a 0 bit, it destuffs the 0 bit.

Advantages:

- The boundary between two frames is unambiguously recognized by the flag pattern (flag sequences can only occur at frame boundaries and never within the data).
- Frames can contain an arbitrary number of bits made up of units of any size.
- Ensures a minimum density of transitions that help the physical layer maintain synchronization.

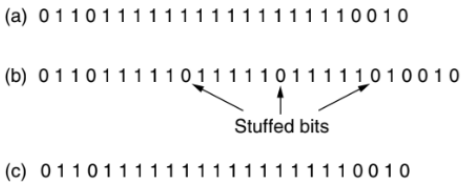


Figure 5. Bit stuffing. (a) The original data. (b) The data as they appear on the line. (c) The data as they are stored in the receiver's memory after destuffing.

Both byte and bit stuffing:

- Are completely transparent to the network layer in both computers.
- Have a frame length that depends on the contents of the data.

Many data link protocols use a combination of these three methods for safety.

2.3 Error detection

2.3.1 Types of Errors

- **Simple Error:** Random and independent from previous error.
- **Errors in burst:**
  - Not independent.
  - Affect neighbour bits.
  - Burst lenght defined by the first and last bits in error.

2.3.2 Counting Errors

Frame Error Probability(FER):

$$FER = 1 - (1 - BER)^n \tag{5}$$

BER = Bit Error Ratio
n = frame length

No Error Probability: $P = (1 - p)^n$
Error Probability: $P = 1 - (1 - p)^n$
i Error Probability: $P = \binom{n}{i} p^i (1 - p)^{n-i}$

p = bit error probability
n = frame length

2.3.3 Error Detection Techniques

Used by the receiver to determine if a packet contains errors. If a packet is found to contain errors, the receiver may request the transmitter to re-send the packet.

2.3.4 Parity Check

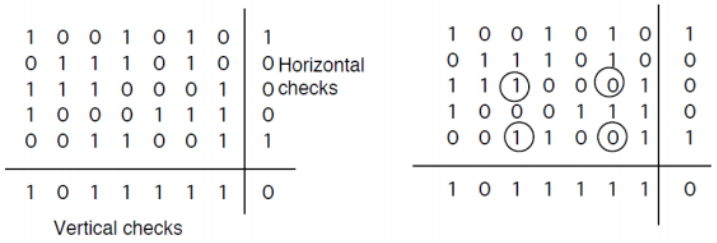
Simple Parity Check: One parity bit added to every k information bits so that:

- The total number of bits 1 even (even parity).
- The total number of bits 1 odd (odd parity).

Allows the detection of simple errors and any number of odd errors in a block of k + 1 bits. However, does not detect even number of errors in a block of k + 1 bits.

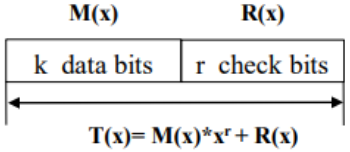
Bi-dimensional Parity

- Parity per row.
- Parity per column.
- Minimum code distance = 4.



2.3.5 Cyclic Redundancy Check (CRC)

A fixed number of check bits are appended to the message to be transmitted. Data receivers check on the check value attached by finding the remainder of the polynomial division of the contents transmitted. If it seems that an error has occurred, a negative acknowledgement is transmitted asking for data retransmission.



The bit string is represented as a polynomium (i.e. $110011 \Rightarrow x^5+x^4+x+1$)

How to compute the check bits: R(x)?

- Choose a generator string G(x) of length r+1 bits.
- Choose R(x) such that T(x) is a multiple of $G(x) : T(x) = A \times G(x)$.

Generating R(x):

$$R(x) = M(x)x^r \textbf{ mod } G(x) \tag{6}$$

Choice of G(x) is very important! ($G(x) = x^r + \ldots + 1$)

Generating R(x) example:

Assume for example:

- r=3.
- $G(x) = x^3 + 1 \Rightarrow 1001$.
- $M(x) = x^5 + x^4 + x^2 + 1 \Rightarrow 110101$.

Then:

- $x^r = x^3$.
- $M(x) \times x^3 = x^8 + x^7 + x^5 + x^3 \Rightarrow 110101000$.
- $R(x) = M(x)x^3$ **mod** $G(x) = 110101000$ **mod** $1001 = 011 = x^1 + 1$

Checking at the Receiver

- Divide T(x) by G(x):
  - If the remainder $R(x) = 0 \Rightarrow$ no errors.
  - If the remainder $R(x)! = 0 \Rightarrow$ errors have occurred.

Performance:

For r check bits per frame the following can be detected

- All patterns of 1, 2, or 3 errors (d $>$ 3).
- All bursts of errors of r or fewer bits.
- All errors consisting of an odd number of inverted bits.

2.4 Automatic Repeat reQuest (ARQ)

An error-control method for data transmission that uses acknowledgements (messages indicating whether or not the message has been correctly received) and timeouts to achieve reliable data transmission over an unreliable service. This mechanisms automatically request the retransmission of:

- Missing packets.
- Packets with errors.

There are three common ARQ schemes:

2.4.1 Stop and Wait

- Sender transmits information frame I and waits for positive confirmation ACK from receiver.
- Receiver receives I frame:
  - If I frame has no error sends ACK.
  - If I frame has error sends NACK.
- Sender receives I frame:

  - If ACK, proceeds and transmits new frame.
  - If NACK, retransmits frame I.

- If I, ACK or NACK is lost a timeout is required!

Issue: If the transmitter times-out and sends a packet twice, the receiver cannot tell whether the second frame is a retransmission or a new frame transmission.
Solution: Define a 1 bit sequence number in the header of the frame.

This sequence number alternates (from 0 to 1) in subsequent frames. The transmitter sends a frame with a sequence number attached to it so the receiver can check if it matches the expected. When the receiver sends an ACK, it includes the sequence number of the next packet it expects. This way, the receiver can detect duplicated frames by checking if the frame sequence numbers alternate.

Efficiency(S):

$$S = \frac{T_f}{T_f + 2 \times T_{prop}} = \frac{1}{1 + 2a} \tag{7}$$

where:

T_f = Data transmission time
T_{prop} = Propagation Delay

Probability of k Attempts required to transmit a frame with success

$$P[A = k] = {p_e}^{k-1}(1 - p_e) \tag{8}$$

where:

p_e = frame error probability(FER)

Expected number of Attempts to transmit a frame with success

$$E[A] = \frac{1}{1 - p_e} \tag{9}$$

where:

p_e = frame error probability(FER)

Efficiency with Errors

$$S = \frac{T_f}{E[A](T_f + 2 \times T_{prop})} = \frac{1 - p_e}{1 + 2a} \tag{10}$$

where:

p_e = frame error probability(FER)

2.4.2 Go Back N

Allows the transmission of new packets before earlier ones are acknowledged.

Sender:

- May transmit up to W frames without receiving RR(Receiver Ready = ACK).
- I frames are numbered sequentially I(NS): I(0), I(1), I(2), etc.
- Cannot send I(NS=i+W) until it has received the RR(NR=i).

Receiver:

- Does not accept frames out of sequence.
- Sends RR(NR) to sender indicating:
  - That all the packets up to NR-1 have been received in sequence.
  - The sequence number, NR, of the next expected frame.

Behaviour under Errors

- Frames with errors are silently discarded by the Receiver.
- If Receiver receives Data frame out of sequence:
  - First out-of-sequence-frame: Receiver sends REJ(NR) where NR = next in-sequence frame expected.
  - Following out-of sequence-frames: Receiver discards them; no REJ sent.
- When Sender receives REJ(NR=x), the Sender:
  - Goes-Back and retransmits I(x), I(x+1), etc.
  - Continues using Sliding Window mechanism.
- If timeout occurs, the Sender:
  - Requests the Receiver to send a RR message.
  - Sends a special message (RR command message).

Maximum Window Size(W):

$$W = M - 1 = 2^k - 1 \tag{11}$$

where:

M = Number of sequence numbers
k = Number of bits used to code sequence numbers

Efficiency:

- If $W \geq 1 + 2a \Rightarrow S = 1$.
- If $W < 1 + 2a \Rightarrow S = \frac{W}{1+2a}$.

Efficiency with Errors:

$$S = \begin{cases} \dfrac{1-p_e}{1+2ap_e} & ,W \geq 1+2a \\ \dfrac{W(1-p_e)}{(1+2a)(1-p_e+Wp_e)} & ,W < 1+2a \end{cases}$$

Figura 1: pe - frame error probability (ratio, FER)

2.4.3 Selective Repeat

Similar to **Go Back N**, however it does not discard successful frames when errors occur.

Receiver:

- Accepts out of sequence frames.
- Confirms negatively, SREJ, a frame not arrived.
- Uses RR to confirm blocks of frames arrived in sequence.

Sender: Retransmits only the frames signaled by SREJ.

Maximum window size(W):

$$W = \frac{M}{2} = 2^{k-1} \tag{12}$$

where:

M = Number of sequence numbers
k = Number of bits used to code sequence numbers

Efficiency:

$$S = \begin{cases} 1 - p_e & , W \geq 1 + 2a \\ \dfrac{W(1-p_e)}{1+2a} & , W < 1 + 2a \end{cases}$$

Figura 2: pe - frame error probability (ratio, FER)

2.4.4 Useful Formulas for All Methods

Data transmission time (T_f):

$$T_f = \frac{L}{R} \tag{13}$$

where:

L = Frame Size
R = Data Rate

Propagation Delay(T_{prop}):

$$T_{prop} = \frac{d}{V} \tag{14}$$

where:

d = Distance between sender and receiver
V = Propagation Velocity

SUMETHIN(a)

$$a = \frac{T_{prop}}{T_f} \tag{15}$$

where:

T_{prop} = Propagation Delay
T_f = Data transmission time

Maximum Rate(R_{max}):

$$R_{max} = S \times R \tag{16}$$

where:

S = Efficiency
R = Data rate

Round Trip Time(RTT - Time of transmission and acknowledgement of a frame):

$$RTT = 2 \times T_{prop} + T_f \tag{17}$$

where:

T_{prop} = Propagation Delay
T_f = Data transmission time

2.5 Framing, Error detection and ARQ in common networks

2.5.1 Ethernet

- **Framing:**
  - Start of Frame - preamble + SFD.
  - End of Frame - end of signal transitions(Manchester code).
- **Error Detection:** CRC using ITU-32 (common polynomial for G(x)).
- **No ARQ:**
  - Bit Error ratio (BER) very low.
  - Frame Error Ratio (FER) low.
  - CRC/FCS strong: good detection of error frames.

2.5.2 Point to Point Protocol

- **Framing:** FLAGS - 0x7E; Byte Stuffing - 0x7D.
- **Error Detection:** Can be negotiated.
- **No ARQ.**

2.5.3 Wireless LAN

- **Framing:**
  - Synchronization.
  - Start Frame Delimiter.
  - Length: Payload length in us.
- **Error Detection:** CRC (Header Error Check) using ITU-16 (common polynomial for G(x)).
- **ARQ:** Modified version of Stop and Wait .

2.5.4 High-Level Data Link Control

- **Framing:** FLAGS and bit stuffing.
- **Error Detection:** CRC using ITU-16 (common polynomial for G(x)).
- **ARQ:** Selective Repeat.
- Used as basis for many telecom networks.

2.6 Reliability in the TCP/IP Reference Model

The TCP/IP reference model assumes:

- Every data link layer offers an error free service to the upper layer.
- Service Data Units are:
  - Delivered to upper layer without error.
  - Discarded.

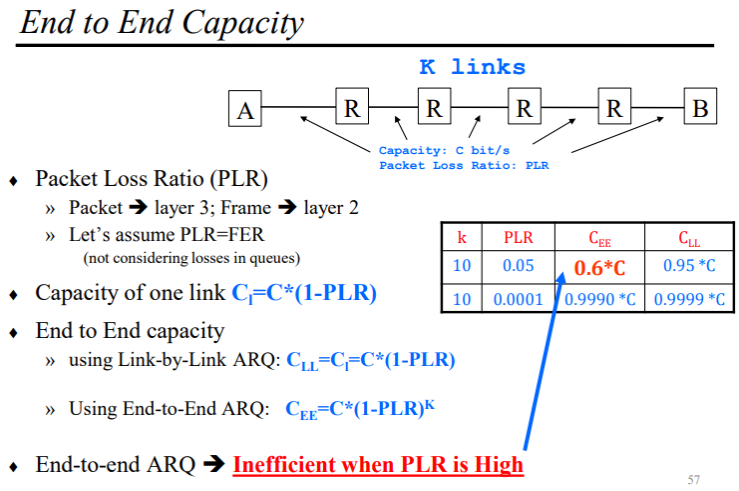
The layered model transforms bit error in packet losses. Therefore, packet losses must be repaired. Two strategies can be used:

2.6.1 Link-by-Link ARQ

- Repairs losses link by link.
- Requires network elements to
  - Remember information about packet flows ⇒ High processing per frame/packet.
  - Store packets in case they have to be retransmitted ⇒ Memory required.

2.6.2 End-to-End ARQ

- Low complexity : Intermediate network elements (switches, routers) become simple.
- Packets may follow different end to end paths.
- Not acceptable when Packet Loss Ratio is high



2.6.3 ARQ in the TCP/IP Reference Model

The TCP/IP architecture assumes that:

- The Data Link layer provides error-free packets to the network layer.
- Data link layer provides a service with very low FER.
- End-to-End ARQ is used, implemented at Transport or Application Layers.

In the TCP/IP reference model, packet losses are repaired:

- At the Data Link layer on lossy channels (e.g. wireless data links).
- At the end systems (transport layer or application layer).

2.7 Data Link Layer Some Exam Exercises - Part 2 Only

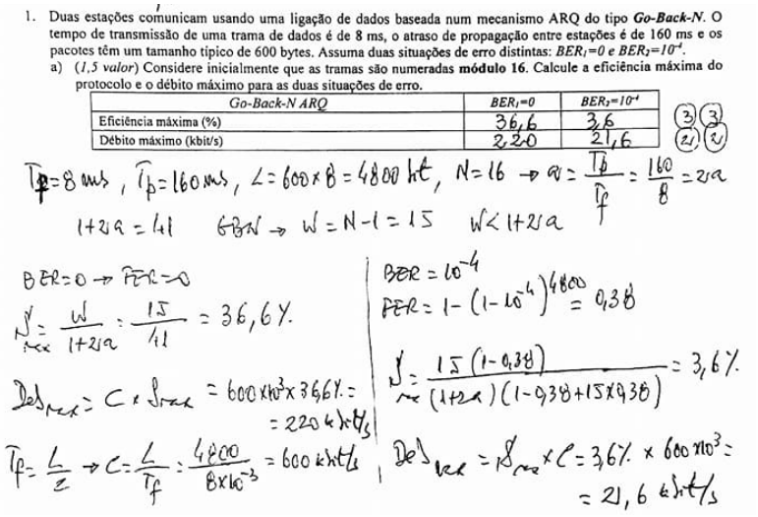
Note: In some exams, some question may need information given in previous questions.

2.7.1 2017/18 Exam

Efficiency and Maximum Rate

1. Duas estações comunicam usando uma ligação de dados baseada num mecanismo ARQ do tipo ***Go-Back-N***. O tempo de transmissão de uma trama de dados é de 8 ms, o atraso de propagação entre estações é de 160 ms e os pacotes têm um tamanho típico de 600 bytes. Assuma duas situações de erro distintas: $BER_1=0$ e $BER_2=10^{-4}$.
   a) (*1,5 valor*) Considere inicialmente que as tramas são numeradas **módulo 16**. Calcule a eficiência máxima do protocolo e o débito máximo para as duas situações de erro.

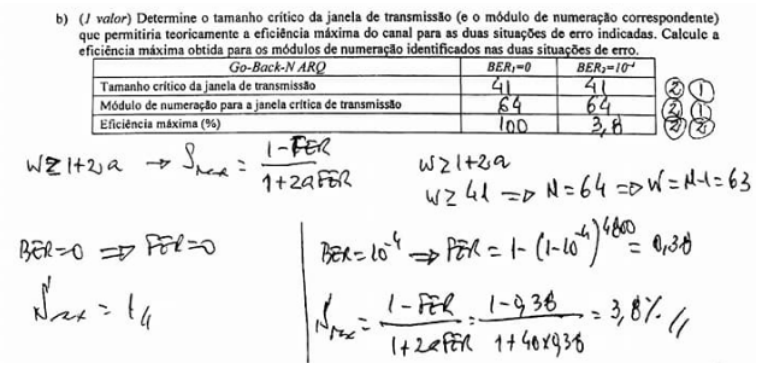
Go-Back-N ARQ	$BER_1=0$	$BER_2=10^{-4}$
Eficiência máxima (%)	36,6	3,6
Débito máximo (kbit/s)	2,20	21,6



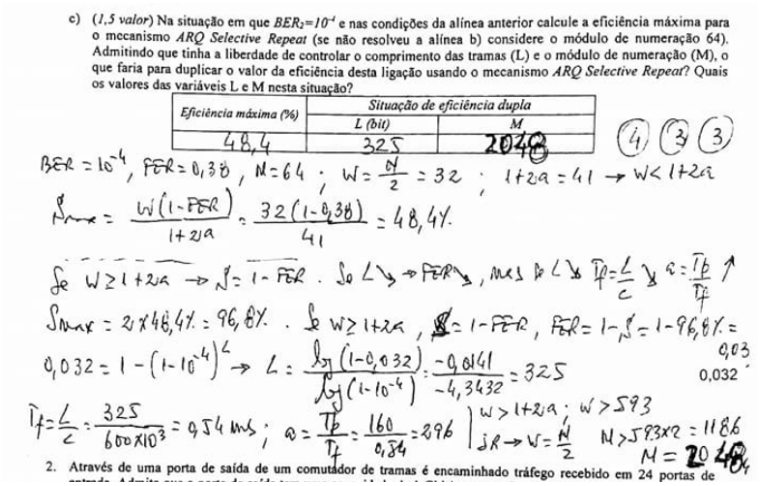
Window Maximum Size, Module (M) and Efficiency

   b) (*1 valor*) Determine o tamanho crítico da janela de transmissão (e o módulo de numeração correspondente) que permitiria teoricamente a eficiência máxima do canal para as duas situações de erro indicadas. Calcule a eficiência máxima obtida para os módulos de numeração identificados nas duas situações de erro.

Go-Back-N ARQ	$BER_1=0$	$BER_2=10^{-4}$
Tamanho crítico da janela de transmissão	41	41
Módulo de numeração para a janela crítica de transmissão	64	64
Eficiência máxima (%)	100	3,8



Selective Repeat Efficiency; Frame Size(L) and Module(M) to double that Efficiency



c) (*1,5 valor*) Na situação em que $BER_2=10^{-4}$ e nas condições da alínea anterior calcule a eficiência máxima para o mecanismo *ARQ Selective Repeat* (se não resolveu a alínea b) considere o módulo de numeração 64). Admitindo que tinha a liberdade de controlar o comprimento das tramas (L) e o módulo de numeração (M), o que faria para duplicar o valor da eficiência desta ligação usando o mecanismo *ARQ Selective Repeat*? Quais os valores das variáveis L e M nesta situação?

Eficiência máxima (%)	*Situação de eficiência dupla*	
	L (bit)	*M*
48,4	325	2048

(4) (3) (3)

$BER = 10^{-4}$, $FER = 0,38$, $N = 64$; $W = \frac{N}{2} = 32$; $1+2a = 41 \rightarrow W < 1+2a$

$S_{max} = \frac{W(1-FER)}{1+2a} = \frac{32(1-0,38)}{41} = 48,4\%$

Se $W \geq 1+2a \rightarrow S = 1-FER$. So $L \downarrow \rightarrow FER \downarrow$, mas se $L \downarrow$ $T_p = \frac{L}{c} \downarrow$ $a = \frac{T_p}{T_f} \uparrow$

$S_{max} = 2 \times 48,4\% = 96,8\%$. Se $W \geq 1+2a$, $S = 1-FER$, $FER = 1-S = 1-96,8\% = 0,032$

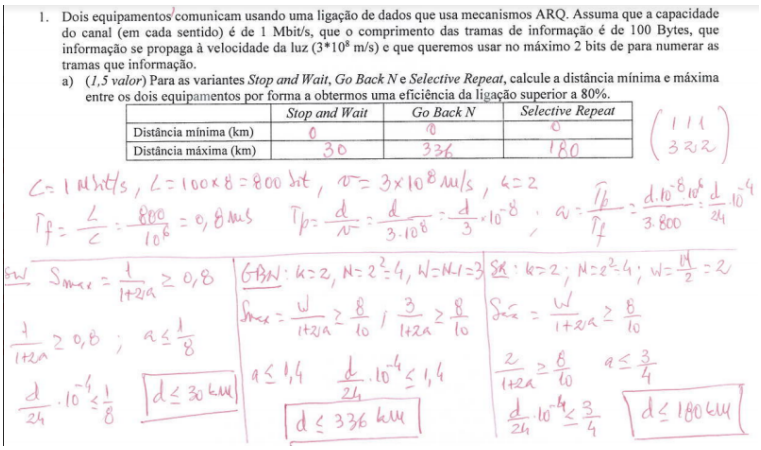
$0,032 = 1-(1-10^{-4})^L \rightarrow L = \frac{\log(1-0,032)}{\log(1-10^{-4})} = \frac{-0,0141}{-4,3432} = 325$ (0,03 / 0,032)

$T_f = \frac{L}{c} = \frac{325}{600 \times 10^3} = 0,54\,ms$; $a = \frac{T_p}{T_f} = \frac{160}{0,54} = 296$ | $W > 1+2a$; $W > 593$; $SR \rightarrow W = \frac{N}{2}$; $N > 593 \times 2 = 1186$; $N = 2048$

2. Através de uma porta de saída de um comutador de tramas é encaminhado tráfego recebido em 24 portas de

2.7.2 2016/17 Exam

Minimum and Maximum Distances between two machines, having an efficiency above X



1. Dois equipamentos comunicam usando uma ligação de dados que usa mecanismos ARQ. Assuma que a capacidade do canal (em cada sentido) é de 1 Mbit/s, que o comprimento das tramas de informação é de 100 Bytes, que informação se propaga à velocidade da luz ($3*10^8$ m/s) e que queremos usar no máximo 2 bits de para numerar as tramas que informação.
   a) (*1,5 valor*) Para as variantes *Stop and Wait*, *Go Back N* e *Selective Repeat*, calcule a distância mínima e máxima entre os dois equipamentos por forma a obtermos uma eficiência da ligação superior a 80%.

	Stop and Wait	*Go Back N*	*Selective Repeat*
Distância mínima (km)	0	0	0
Distância máxima (km)	30	336	180

(1 1 1 / 3 3 2)

$C = 1\,Mbit/s$, $L = 100 \times 8 = 800\,bit$, $v = 3 \times 10^8\,m/s$, $k = 2$

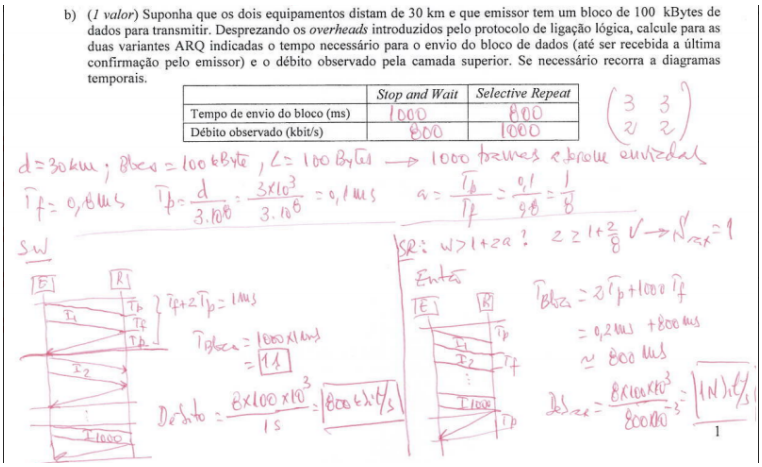
$T_f = \frac{L}{C} = \frac{800}{10^6} = 0,8\,ms$ $T_p = \frac{d}{v} = \frac{d}{3 \cdot 10^8} = \frac{d}{3} \cdot 10^{-8}$; $a = \frac{T_p}{T_f} = \frac{d \cdot 10^{-8} \cdot 10^6}{3 \cdot 800} = \frac{d}{24} \cdot 10^{-4}$

SW: $S_{max} = \frac{1}{1+2a} \geq 0,8$; $\frac{1}{1+2a} \geq 0,8$; $a \leq \frac{1}{8}$; $\frac{d}{24} \cdot 10^{-4} \leq \frac{1}{8}$; $d \leq 30\,km$

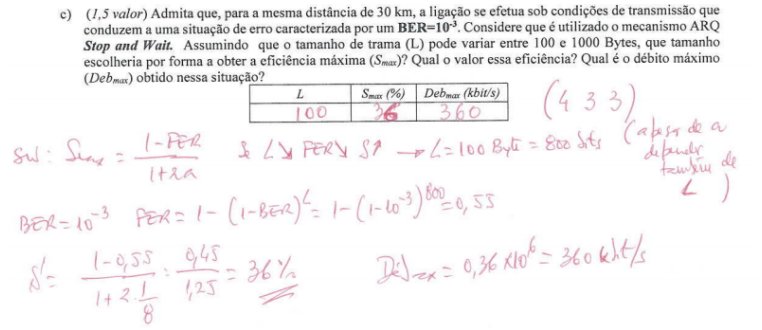
GBN: $k = 2$, $N = 2^2 = 4$, $W = N-1 = 3$; $S_{max} = \frac{W}{1+2a} \geq \frac{8}{10}$ / $\frac{3}{1+2a} \geq \frac{8}{10}$; $a \leq 1,4$; $\frac{d}{24} \cdot 10^{-4} \leq 1,4$; $d \leq 336\,km$

SR: $k = 2$; $N = 2^2 = 4$; $W = \frac{N}{2} = 2$; $S_{max} = \frac{W}{1+2a} \geq \frac{8}{10}$; $\frac{2}{1+2a} \geq \frac{8}{10}$; $a \leq \frac{3}{4}$; $\frac{d}{24} 10^{-4} \leq \frac{3}{4}$; $d \leq 180\,km$

Block of Data Send Time and Observed Rate



Choose Block of Data Size and calculate Efficiency and Maximum Rate



2.7.3 2015/16 Exam

Window Size, Efficiency and Maximum Rate

1. Duas estações separadas por uma distância de 2000 km comunicam usando um protocolo de ligação de dados do tipo ARQ. O atraso de propagação da informação é de 5 µs/km e a capacidade do canal é 1024 kbit/s (em cada sentido). Admita que as tramas de Informação usam 3 bits para numeração, têm um tamanho típico de 2048 bits e são imediatamente confirmadas por tramas de Supervisão em sentido oposto. Despreze o tamanho das tramas de Supervisão.

a) *(1,5 valor)* Para as variantes *Go-Back-N* e *Selective Repeat*, calcule a janela de transmissão, a eficiência máxima do protocolo e os débitos máximos.

	Go-Back-N	*Selective Repeat*
Janela de transmissão, W	7	4
Eficiência máxima, S (%)	63,6	36,1
Débito Máximo (kbit/s)	651	370

$\begin{pmatrix} 2 & 2 \\ 2 & 2 \\ 1 & 1 \end{pmatrix}$

$d = 2000\ km$; $T_p = 5\ \frac{\mu s}{km} \times 2000\ km = 10\ ms$; $T_f = \frac{L}{C} = \frac{2048\ bit}{1024 \times 10^3\ bit/s} = 2\ ms$

$a = T_p / T_f = \frac{10}{2} = 5$; $1 + 2a = 11$; $M = 2^k = 2^3 = 8$

GBN: $W = M - 1 = 8 - 1 = 7$

$W < 1 + 2a \Rightarrow S_{GBN} = \frac{W}{1+2a} = \frac{7}{11} = 63,6\%$

$Deb_{GBN} = S_{GBN} \times C = 0,636 \times 1024 = 651\ kbit/s$

SR: $W = \frac{M}{2} = \frac{8}{2} = 4$

$S_{SR} = \frac{W}{1+2a} = \frac{4}{11} = 36,1\%$

$Deb_{SR} = S_{SR} \times 1024 = 370\ kbit/s$

Efficiency using two different frame sizes

b) *(1 valor)* Pretende-se analisar o efeito dos erros de transmissão e do tamanho das tramas de Informação. Considere tramas com tamanhos 1024 e 2048 bits e uma situação de ruído caracterizada por **BER=10^{-3}**. Calcule a eficiência máxima dos dois mecanismos para estes 2 casos e discuta o comportamento destes mecanismos em relação ao tamanho das tramas

S_{max} (%)	*Go-Back-N*	*Selective Repeat*
L=2048	1,3	4,7
L=1024	2,5	6,9

$\begin{pmatrix} 3 & 2 \\ 2 & 3 \end{pmatrix}$

$L_1 = 2048$: $FER_1 = 1 - (1 - BER)^{L_1} = 1 - (1 - 10^{-3})^{2048} = 0,87$; $1 - FER_1 = 0,13$

GBN: $7 < 11 \Rightarrow S = \frac{W(1-FER)}{(1+2a)(1-FER+WFER)} = \frac{7 \times 0,13}{11 \times (0,13 + 7 \times 0,87)} = \frac{0,91}{68,42} = 1,3\%$

SR: $4 < 11 \Rightarrow S = \frac{W(1-FER)}{1+2a} = \frac{4 \times 0,13}{11} = 4,7\%$

$L_2 = 1024$; $T_f = \frac{L}{C} = \frac{1024}{1024 \times 10^3} = 1\ ms$; $a = \frac{T_p}{T_f} = \frac{10}{1} = 10$; $1 + 2a = 21$

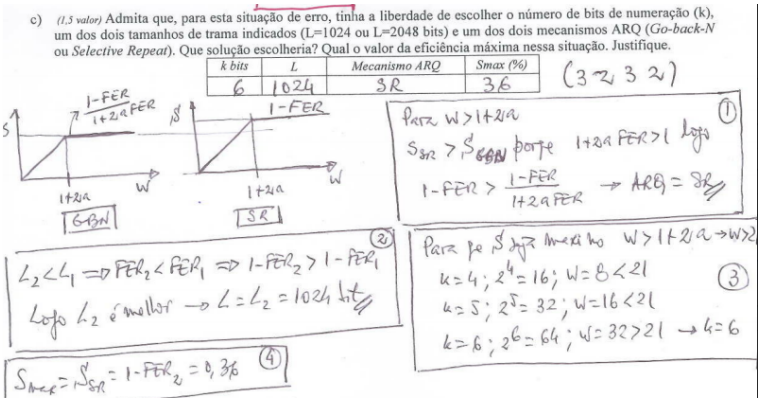
$FER_2 = 1 - (1 - BER)^{L_2} = 1 - (1 - 10^{-3})^{1024} = 0,64$; $1 - FER_2 = 0,36$

GBN: $7 < 21 \Rightarrow S = \frac{W(1-FER)}{(1+2a)(1-FER+WFER)} = \frac{7 \times 0,36}{21(0,36 + 7 \times 0,64)} = \frac{2,36}{101,64} = 2,5\%$

SR: $4 < 21 \Rightarrow S = \frac{W(1-FER)}{1+2a} = \frac{4 \times 0,36}{21} = 6,9\%$

1

Number of bits for sequence number, Frame Size, Mechanism ARQ, Efficiency



3 Delay Models

3.1 Communication Link

- Bit pipe with a given capacity C (bit/s)
- Link capacity -> rate at which bits are transmitted to the link
- Link may transport multiplexed traffic streams

Important Variables and Expressions

C
: channel capacity (total capacity)

3.2 Multiplexing Strategies

- Statistical Multiplexing
- Frequency Division Multiplexing
- Time Division Multiplexing

3.3 Statistical Multiplexing

- Packets of all traffic streams merged in a single quele (first-come, first-served)

Important Variables and Expressions

L
: Length of packet

T_{frame}
: time of transmition

$$T_{frame} = \frac{L}{C} \tag{18}$$

3.4 Frequency Division Multiplexing

- Link capacity C subdivided into m portions
- Channel bandwidth W subdivided into m channels of W/m Hz
- Capacity of each channel = C/m

Important Variables and Expressions

L
: Length of packet

T_{frame}
: time of transmition

m
: number of divisions

W
: channel bandwidth

$$T_{frame} = \frac{Lm}{C} \tag{19}$$

3.5 Time Division Multiplexing

- Time axis divided into m slots of fixed length
- Communication -> m channels with capacity C/m

Important Variables and Expressions

L
: Length of packet

T_{frame}
: time of transmition

m
: number of divisions

$$T_{frame} = \frac{Lm}{C} \tag{20}$$

3.6 Queue Models

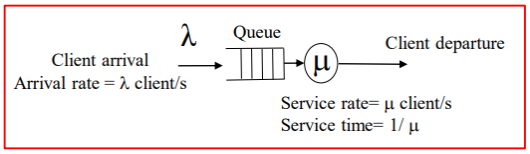


Figura 3: Depiction of a queue model

- Characterization of Delay - Important performance parameter in computer networks
- Customers (packet to be transmitted through a link) arrive at random times to obtain service (transmit a packet)

Important Variables and Expressions

λ
: arrival rate

μ
: service rate

N
: Average number of customers/packets in the network

T
: Average delay per packet -> waiting plus service times

ρ
: traffic intensity (occupation of the server)

$T_{pac(frame)}$
: Service time = packet transmission time

$$T_{pac(frame)} = \frac{L}{C} = \frac{1}{\mu} \tag{21}$$

$$\rho = \frac{\lambda}{\mu} \tag{22}$$

3.6.1 M/M/1 Queue

Important Variables and Expressions

T_W
: average waiting time

N_W
: average number of clients waiting

$$N = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda} \tag{23}$$

$$T = \frac{1}{\mu - \lambda} \tag{24}$$

$$T_W = T - T_S = \frac{1}{\mu - \lambda} - \frac{1}{\mu} = \frac{\rho}{\mu(1-\rho)} \tag{25}$$

$$N_W = T_w \lambda = \frac{\lambda}{\mu - \lambda} - \frac{\lambda}{\mu} = N - \rho \tag{26}$$

3.6.2 M/D/1 Queue

$$T_W = \frac{\lambda E(T^2_{pac(frame)})}{2(1-\rho)} \tag{27}$$

4 MAC Sublayer

4.1 Introduction

The medium access control (MAC) sublayer and the logical link control (LLC) sublayer together make up the data link layer.

The LLC sublayer provides multiplexing mechanisms that make it possible for several network protocols to coexist within a multipoint network and to be transported over the same network medium. It can also provide flow control and automatic repeat request (ARQ) error management mechanisms. The LLC sublayer acts as an interface between the media access control (MAC) sublayer and the network layer.

When sending data to another device on the network, the MAC block encapsulates higher-level frames into frames appropriate for the transmission medium, adds a frame check sequence to identify transmission errors, and then forwards the data to the physical layer as soon as the appropriate channel access method permits it. Controlling when data is sent and when to wait is necessary to avoid congestion and collisions. Additionally, the MAC is also responsible for compensating for congestion and collisions by initiating retransmission if a jam signal is detected, and/or negotiating a slower transmission rate if necessary.

4.2 Multiple Access Links

- **Point to Point** (single wire):
  - PPP for dial-up access;
  - Point-to-point link between Ethernet switch and host;
- **Broadcast** (shared medium, wired or wireless):
  - old-fashioned cabled Ethernet;
  - 802.11 wireless LAN;

4.2.1 Ideal Multiple Access Protocol

Used to coordinate the stations to use a common broadcast and shared channel of rate R bit/s.

- **one** station wants to transmit -> it uses the **R** bit/s;
- **m** stations want to transmit -> each station uses an average rate **R/m** bit/s
- **simple** and **decentralized** (no coordination, no synchronization of clocks).

4.2.2 MAC Model and Concepts

Independent Traffic: The model consists of N independent stations, each with a program or user that generates frames for transmission. The expected number of frames generated in an interval of length g is d*g, where d is a constant.Once a frame has been generated, the station is blocked and does nothing until the frame has been successfully transmitted. To analyze the protocols it uses Poisson Models.
Single Channel: A single channel is available for all communication. All stations can transmit on it and all can receive from it
Collision: Happens if two frames are transmitted simultaneously by different stations. A collided frame must be transmitted again later
Continuous Time: frame can be transmitted at any time.
Slotted Time: frame can be transmitted only at the beginning of a time slot.
Carrier Sense: station can know if medium (channel) is busy before using it.
No Carrier Sense: station cannot sense channel before using it.
Poisson models

4.3 MAC Protocols

The MAC Protocols can be separated into Three Classes: Channel Partitioning, Random Access and Taking turns.

4.4 Channel Partitioning

Divide channel into smaller "pieces"(time slots,frequency).
The communication resource is partitioned in N channels that are assigned to stations is a quasi-static way.
Poor efficiency on low loaded channels.
The principle protocols are:

- Time Division Multiplexing;
- Frequency Division Multiplexing.

4.5 Random access protocols

- Each station tries to access the full communication resource in a random, uncoordinated manner -> collisions occur;
- Poor efficiency on highly loaded channels;
- When station has packet to send a packet,transmits at channel data rate **R** bit/s.
- Random Access MAC protocol defines:
  - when to send data;
  - how to detect collisions;
  - how to recover from collisions.

4.5.1 Pure ALOHA

Aloha is a technique for coordinating the access of large numbers of intermittent transmitters in a single shared communication channel.

- In Aloha, whenever a station has data, it transmits it;
- If more than one frames are transmitted, they collide and are lost. The Sender finds out whether the transmission was successful or experienced a collision by listening to the broadcast from the destination station;
- If ACK (signal that data has been received successfully) not received within timeout, then a station picks random backoff algorithm to re-transmit;
- After the backoff time, the station re-transmits the frame.

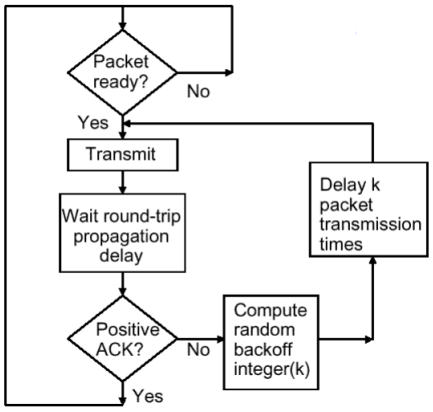


Figura 4: Station behaviour

4.5.2 Slotted ALOHA

In the Slotted ALOHA, the time is divided into time slots and each slot corresponds to one frame.
A station is not permitted to send whenever the user types a line. Instead, it is required to wait for the beginning of the next slot and the stations transmit frames only at the beginning of a time slot.
This method causes a reduction on the collision probability.

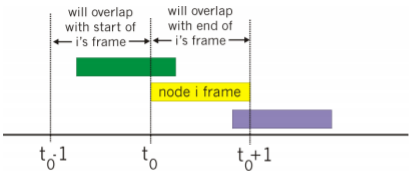


Figura 2- Pure ALOHA

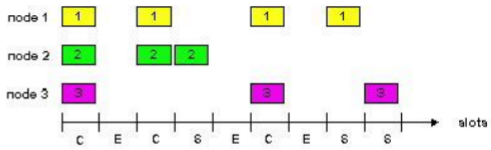


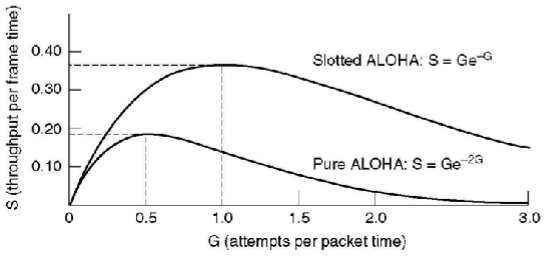
Figura 3- Slotted ALOHA

ALOHA Efficiency:

- Traffic model
  - » Poisson arrival, large number N of stations
  - » Constant frame length, T_{frame} =1
  - » S – Received traffic
    - λ_{rx} – rate of received frames (transmitted with success)
    - $S = \lambda_{rx} * T_{frame} < 1$; **S = efficiency**
  - » G – Generated traffic (new packets and retransmissions)
    - λ – rate of generated packets
    - $G = \lambda * T_{frame}$
  - » p – probability of **one station** generating a packet (new or retransmission) in T_{frame}
    - $N*p=G$

- Slotted Aloha
  - » $S = P(Success) = N\left(p(1-p)^{N-1}\right) \approx Npe^{-p(N-1)} \approx Npe^{-pN} = Ge^{-G} = Gp_0(T_{frame}) = Ge^{-G}$
  - » $S_{max} \Rightarrow \frac{\partial S}{\partial G} = 0; \quad G = 1; \quad S_{max} = \frac{1}{e} = 36{,}8\%$

- Pure Aloha
  - » $S = Gp_0(2 \times T_{frame}) = Ge^{-2G}$
  - » $S_{max} \Rightarrow \frac{\partial S}{\partial G} = 0; \quad G = \frac{1}{2}; \quad S_{max} = \frac{1}{2e} = 18{,}4\%$



4.5.3 CSMA (Carrier Sense Multiple Access)

Carrier-sense multiple access (CSMA) is a media access control (MAC) protocol in which a node verifies the absence of other traffic before transmitting on a shared transmission medium.
A transmitter attempts to determine whether another transmission is in progress before initiating a transmission using a carrier-sense mechanism. If a carrier is sensed, the node waits for the transmission in progress to end before initiating its own transmission.
Persistency - what to do after the medium is found busy

CSMA collisions:

- Collisions can still occur due to a propagation delay or because stations may not hear other transmissions;

- When a collision happens the station waits random time and repeats algorithm;
- Collision vulnerability time = 2*Tprop;
- Collision probability = a = Tprop/Tframe;

CSMA Variants:

- CSMA **1-persistent:** If the channel is idle (free), the stations sends its data. Otherwise, if the channel is busy, the station just waits until it becomes idle. Then the station transmits a frame.
- CSMA **Non-persistent:** If the channel is idle, the stations sends its data. Otherwise, if the channel is busy, the station waits a random time and repeats algorithm
- CSMA **p-persistent:** If the channel is idle, it transmits with a probability p. With a probability $q = 1 - p$, it defers until the next slot. If that slot is also idle, it either transmits or defers again, ith probabilities p and q. This process is repeated until either the frame has been transmitted or another station has begun transmitting. In the latter case, the unlucky station acts as if there had been a collision. If the station initially senses that the channel is busy, it waits until the next slot and applies the algorithm above.

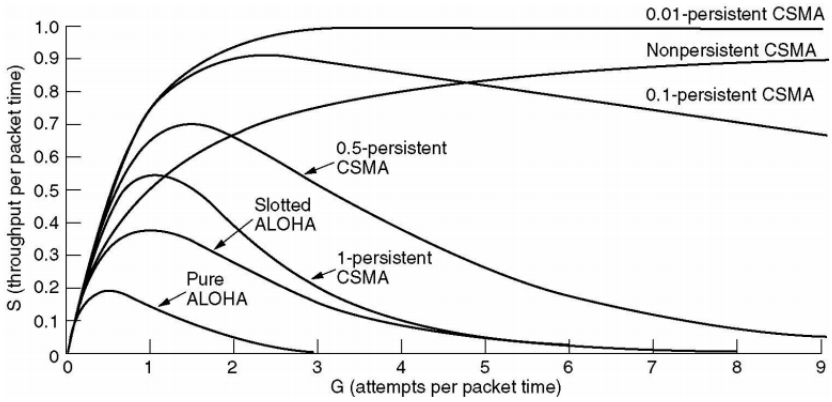


Figura 5: Efficiencies

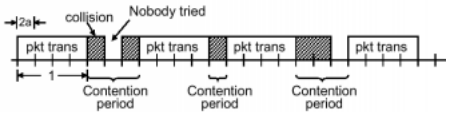
4.5.4 CSMA with Collision Detection (CSMA/CD)

CSMA/CD operates by detecting the occurrence of a collision. Once a collision is detected, CSMA/CD immediately terminates the transmission thus shortening the time required before a retry can be attempted. The last information

can be re-transmitted.
CSMA/CD is used by Ethernet.

- **Carrier Sense:** Station senses medium before transmitting. If free, station starts transmission. If busy, waits until its free and then transmits (= 1-persistent)
- **Collision Detection:** If collision is detected,the transmission is aborted and the re-transmission is delayed using a Binary Exponential Back-off algorithm.
- **Binary Exponential Back-off algorithm:**
  - Time is modeled in time slots and each Tslot= 2*Tprop;
  - After the i consecutive collision -> waits a random number of slots uniformly distributed in [0, 2i-1] and attemps to re-transmit.
- **Doesn't use ACK**.
- To detected a collision,**Tframe>2*Tprop**.

CSMA/CD - Efficiency



- Let's assume $T_{slot} = 2 \times T_{prop}$, $T_{frame} = 1$
- Number slots $\boldsymbol{n_{tx}}$ required to transmit a frame $n_{tx} = \frac{T_{frame}}{T_{slot}} = \frac{T_{frame}}{2 \times T_{prop}} = \frac{1}{2a}$
- Efficiency $S = \frac{n_{tx}}{n_{tx} + E[n_{cont}]}$
- Let's define
  - » p – probability that one station transmits in a slot
  - » A – probability that exactly one station transmits in a slot and gets the medium

$$A = \binom{N}{1} p^1 (1-p)^{N-1} = Np(1-p)^{N-1}$$

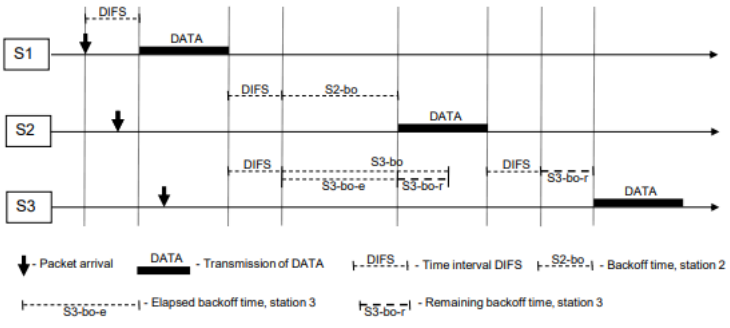
- $E[n_{cont}] = \sum_{i=1}^{+\infty} i(1-A)^i A = \frac{1-A}{A} \quad \Rightarrow S = \frac{1/2a}{1/2a + (1-A)/A} = \frac{1}{1+2a(1-A)/A}$
- p=1/N ➔ $A_{MAX} = \left(1 - \frac{1}{N}\right)^{N-1} \quad \lim_{N\to\infty} A_{\max} = \lim_{N\to\infty} \left(1 - \frac{1}{N}\right)^{N-1} = \frac{1}{e} \quad \Rightarrow \lim_{N\to\infty} S = \frac{1}{1+3.44a}$

4.5.5 CSMA with Collision Avoidance (CSMA/CA)

Carrier-sense multiple access with collision avoidance (CSMA/CA) is a network multiple access method in which carrier sensing is used, but nodes attempt to

avoid collisions by beginning transmission only after the channel is sensed to be idle. When they do transmit, nodes transmit their packet data in its entirety. It is an unreliable method

- if medium free, transmits frame
- if medium busy:
  - Random backoff interval is selected which will be decremented as long as the channel is sensed idle;
  - Stopped when a transmission is detected on the channel;
  - Re-activated when the channel is sensed idle again for more than a DIFS;
  - The station transmits when the backoff time reaches 0.
- To avoid channel capture, station waits random backoff time between two consecutive frame transmissions, even if the medium is sensed idle in the DIFS time.
- **It uses ACKs**.



4.5.6 Taking-turns protocols

Tightly coordinate shared access to avoid collisions.
Usage of the communication resource is disciplined by some turning mechanisms.

- Each station has its own turn;
- Stations with more information to send, might have bigger turns;
- **Polling:**
  - Controlled by a master station which invites slave stations to transmit in turn.
  - **concerns**: polling overhead; latency; single point of failure (master).

- **Token passing:**
  - The stations will pass the control token from one station to next sequentially, warning which is able to transmit.
  - **concerns**: token overhead; latency; single point of failure (token).

4.6 MAC

The standard for wireless LANs is called IEEE 802.11, aka WiFi.
The most common type of wired LANs is called IEEE 803.2, aka Ethernet.

- 48 bit address
- Unique for each adaptor
- Broadcast address FF-FF-FF-FF-FF-FF

4.7 Ethernet

- Bus Topology :

  Popular in the mid 90s

  Stations in same collision domain
- Star Topology :

  Current Topology

  Active switch in center

  Each station runs individual Ethernet protocols

  Stations do not collide with each other
- **Full-Duplex**: The stations don't have to wait for each other and there is no collision. The CSMA/CD algorithm is not needed.

4.8 WiFi

- Bus Topology :

  Popular in the mid 90s

  Stations in same collision domain
- Star Topology :

  Current Topology

  Active switch in center

  Each station runs individual Ethernet protocols

  Stations do not collide with each other

4.9 Switch

- Link-layer device
- Forwards Ethernet frames
- Transparent to hosts
- Does not need to be configured
- Has forwarding table

4.9.1 When the Switch receives a frame:

- Records link associated with sending host
- index forwarding tabel using MAC destination address
- if entry is found in table

    if destination is on segment from which frame arrived,

    drop the frame

    else

    forward the frame on interface indicated else flood (forward on all but the interface on which the frame arrived)

4.10 Virtual LANs

- One bridge/switch simulates multiple LANs/broadcast domains
- One LAN may be extended to other bridges

4.11 Previous Exams Questions:

Dos protocolo de acesso aleatório ao meio estudados

Selecione uma opção de resposta:

- a. CSMA, o CSMA/CD e o CSMA/CA usam a trama de confirmação ACK.
- b. o Aloha, o CSMA e o CSMA/CD usam a trama de confirmação ACK.
- c. o Aloha, o CSMA e o CSMA/CA usam a trama de confirmação ACK. ✓

6. No protocolo de acesso ao meio **CSMA/CD**, quando uma estação emissora deteta uma colisão, esta estação
a) Continua a transmitir a trama até ao fim e retransmite a trama após espera de um número aleatório de *timeslots*.
b) Continua a transmitir a trama até ao fim e retransmite a trama de forma persistente no *timeslot* seguinte.
c) Aborta a transmissão da trama e retransmite a trama após espera de um número aleatório de *timeslots*.
d) Aborta a transmissão da trama e retransmite a trama de forma persistente no *timeslot* seguinte.

5. Assuma que 8 estações competem para aceder a um meio partilhado, que cada estação gera em média 1 pacote/s e que o meio é capaz de transportar 10 pacote/s. Neste cenário, sob o ponto de vista do atraso,
a) Um mecanismo de acesso aleatório (ex. CSMA/CD) é preferível a um mecanismo de TDMA.
b) Um mecanismo TDMA é preferível a um mecanismo de acesso aleatório.
c) Os dois tipos de mecanismos são equivalentes.
d) Nenhum dos dois tipos de mecanismos consegue comutar a quantidade de tráfego indicada.

5. Considere um meio partilhado por um conjunto de computadores. Assuma que a maior distância entre dois computadores é L [m] e que a informação se propaga no meio com uma velocidade S [m/s]. Assuma ainda que os computadores acedem ao meio usando o protocolo CSMA/CD (*Collision Detection*). Nesta situação, o tempo de transmissão T [s] de uma trama deve satisfazer a seguinte condição
a) $T < L/S$
b) $L/S < T < 2L/S$
c) $T > 2L/S$
d) Nenhuma das anteriores.

5. Considere a tecnologia de acesso ao meio *Carrier Sense Multiple Access* (CSMA), o tempo de transmissão de uma trama T_{frame} e o tempo de propagação de uma trama no meio partilhado T_{prop}. O CSMA usa-se em situações em que
a) $T_{frame} >> T_{prop}$.
b) T_{frame} é aproximadamente igual a T_{prop}.
c) $T_{frame} << T_{prop}$.
d) A sua utilização é independente da relação entre T_{frame} e T_{prop}.

6. Assuma um cenário composto por 2 computadores A e B implementando o protocolo de acesso ao meio CSMA/CD (*Collision Detection*), e interligados entre si através de um comutador Ethernet (switch igual ao do laboratório). As portas de rede dos computadores e do comutador funcionam em modo **full-duplex**. Se o computador A estiver a transmitir uma trama e o computador B também tiver uma trama para transmitir, o computador B
a) Escuta até ao fim da transmissão de A e só depois transmite a sua trama.
b) Transmite de imediato a sua trama causando uma colisão.
c) Transmite de imediato a trama mas só haverá colisão se a trama enviada por A tiver como destino B.
d) Transmite de imediato e não haverá colisão.

Quando uma trama é recebida por um switch Ethernet e a tabela de encaminhamento do switch não contém uma entrada para o endereço de destino da trama, o switch

Selecione uma opção de resposta:

- a. lança um pedido na rede para que o computador de destino da trama anuncie a sua presença.
- b. envia a trama para todas as portas excepto a porta através da qual a trama foi recebida. ✓
- c. elimina a trama.

5 Network Layer

6 Network

Network label - camada responsável pela transferência de pacotes

6.1 Exercises

(1º part) (2º part)

1. 2018 Recurso (7,9) (3)
2. 2018 Normal (2,7,9) (3)
3. 2017 Normal (1,8) (3)
4. 2016 Recurso (6,7) (3)
5. 2016 Normal (7) (3)
6. 2015 Normal (1,7) (3)
7. 2014 Normal (2,7,9) (3)
8. 2013 Normal (7) (3)
9. 2012 Normal (6,7,8) (3)
10. 2011 Normal (6,7,8) (3)
11. 2010 Normal (2,4) (3)

6.2 Overview

- Camada de Network (Network layer)
  - Transporta os pacotes(datagrams)
  - "from sending host to receiving host"
  - funções localizadas em todos os hosts e routers
- Transmissor(Sender):
  - Encapsula a informação em pacotes
  - Cria os pacotes
- Receptor(Receiver):
  - Recebe os pacotes
  - Envia a informação para o transport layer
- Router:

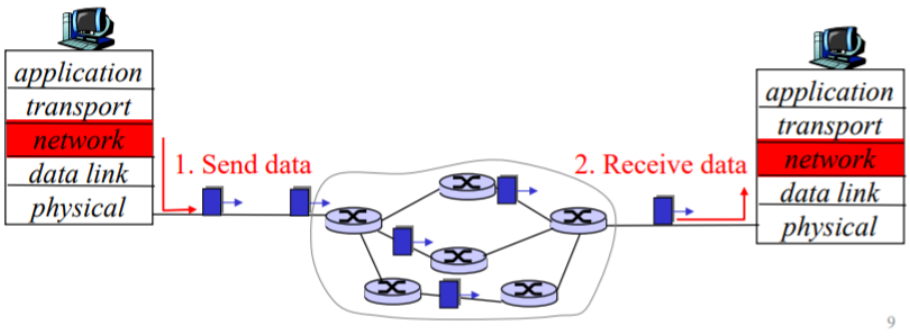
- Recebe os pacotes pela linha de input
- Examina o cabeçalho dos pacotes
- Reencaminha os pacotes para o sítio certo
- Tem de saber o caminho mais curto para determinar o caminho

6.3 Funções principais da camada de rede

- Forwarding
  - router trata de enviar o pacote desde a porta de entrada(input) até à porta de saída(outpu)
- Routing
  - determina a rota definida pelos packets
  - algoritmos, caminho mais curto

6.4 Rede de datagramas

- Serviço não orientado à ligação
- Não há o conceito de circuito
- Os pacotes são redirecionados de acordo com a fonte e o destino
- Pacotes com o mesmo par fonte-destino podem seguir caminhos diferentes



Destination Address Range	Output Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

2^{32} possible entries in IPv4

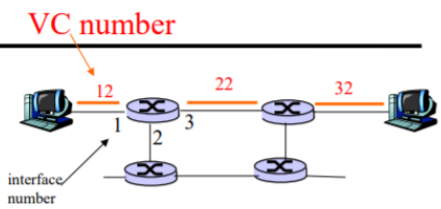
6.5 Circuitos Virtuais

- Serviço orientado à ligação
- Fases:
  1. Estabelecer o circuito
  2. Transferência de dados
  3. Terminação do circuito
- Cada pacote carrega um identificador do circuito virtual
- Caminho da fonte ao destino -> sequência de identificadores virtuais, um para cada ligação
- Estado de cada circuito mantido pelo router, que pode alocar recursos (bandwidth, buffers) por circuito virtual

6.5.1 Forwarding Table

Contém prefixos e a respetiva porta de saída <Endereço/Mask, port>

VC - Forwarding Table



Forwarding table in northwest router:

Incoming interface	Incoming VC #	Outgoing interface	Outgoing VC #
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87
…	…	…	…

Routers maintain connection state information!

8

6.5.2 Ex: Maior correspondência de prefixo

Prefix Match	Link Interface
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
otherwise	3

Examples. Which Interface?

DA: 11001000 00010111 00010110 10100001 → 0

DA: 11001000 00010111 00011000 10101010 → 1,2 → 1

longest prefix

6.6 Circuitos Virtuais versus Rede de Datagramas

Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

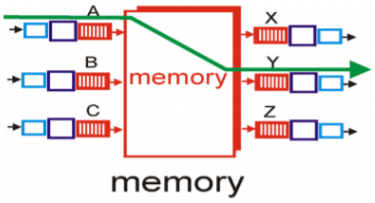
6.6.1 Exame 2016 Recurso - Ex:6

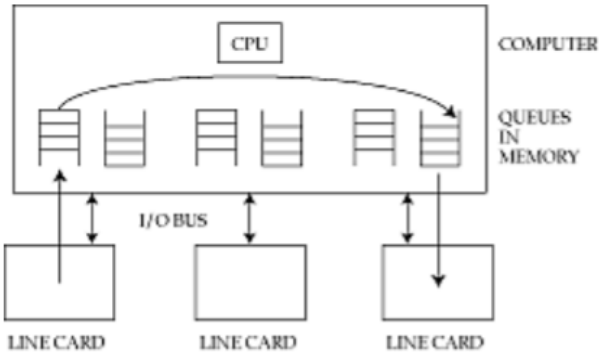
6. Uma rede composta por um conjunto de routers IP interligados entre si que transporta apenas tráfego TCP constitui
a) Uma rede de comutação de pacotes e oferece um serviço não orientado às ligações.
b) Uma rede de comutação de pacotes e oferece um serviço orientado às ligações.
c) Uma rede de circuitos virtuais e oferece um serviço não orientado às ligações.
d) Uma rede de circuitos virtuais e oferece um serviço orientado às ligações.

6.7 Arquitetura do router

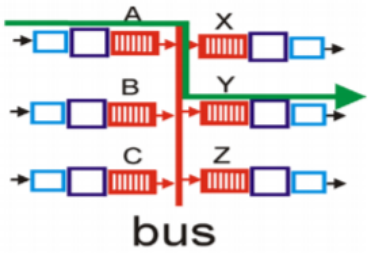
- Funções principais:
  - Correr algoritmos de roteamento e protocolos (RIP, OSPF, BGP)
  - Reencaminhar pacotes
- Componentes principais:
  - Input Port
    - Physical Layer (bit-level)
    - Data Link Layer (e.g., Ethernet)
    - Queuing (se os pacotes chegarem rápido demais)
    - Lookup + Forwarding (faz algum reencaminhamento imediatamente)
  - Output Port
    - Buffering (quando é excedida a velocidade de saída)

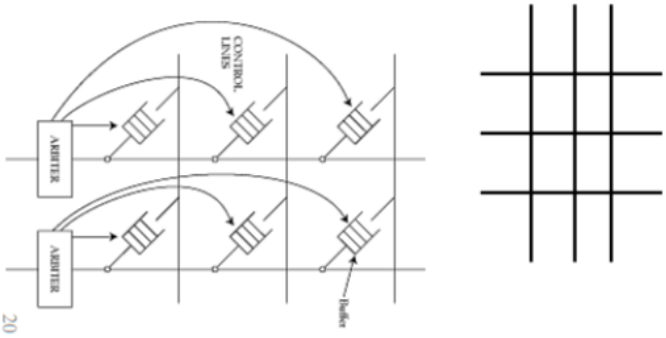
    * Queuing (com disciplina de agendamento)(Queuing perda e espera - devido ao overflow do buffer da porta de input)
    * Data Link Layer (protocol, desencaplsulação)
    * Physical Layer (linha de terminação)
- Switching Fabric
    * Controla o reencaminhamento (fisicamente ou através dum CPU)
    * Switching Via Memória do Computador
        · Router de primeira geração
        · Em computadores tradicionais, switching é controlado pelo CPU
        · Cada pacote é copiado para a memória do sistema e transferida duas vezes pelo bus



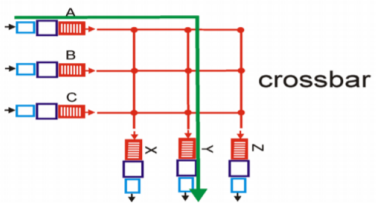


    * Switching via a Bus
        · Os pacotes sao processados por um bus partilhado
        · A transferência dos pacotes desde a linha de input e output é realizada de foram direta
        · A taxa da conecção do bus é limitada pela bus bandwidth





* Switching via a Crossbar
  - 2N buses
  - Possibilita transferências simultaneas de pacotes
  - a cross bar pode conter buffers intermos
  - Ultrapassa os limites da bus bandwidth



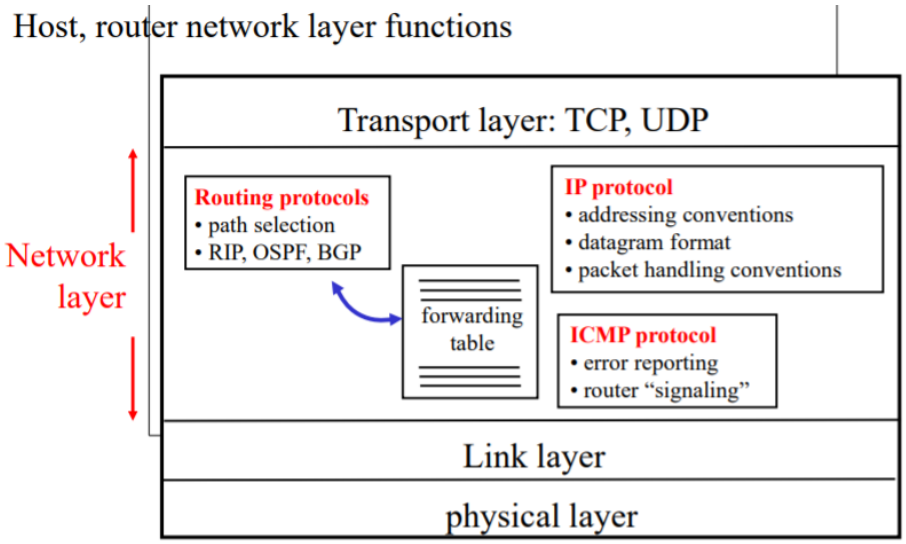
Exame 2016 Normal - Ex.7

7. Quando uma trama é recebida por um *Switch Ethernet* e a tabela de encaminhamento do *Switch* não contém uma entrada para o endereço de destino da trama, o *Switch*

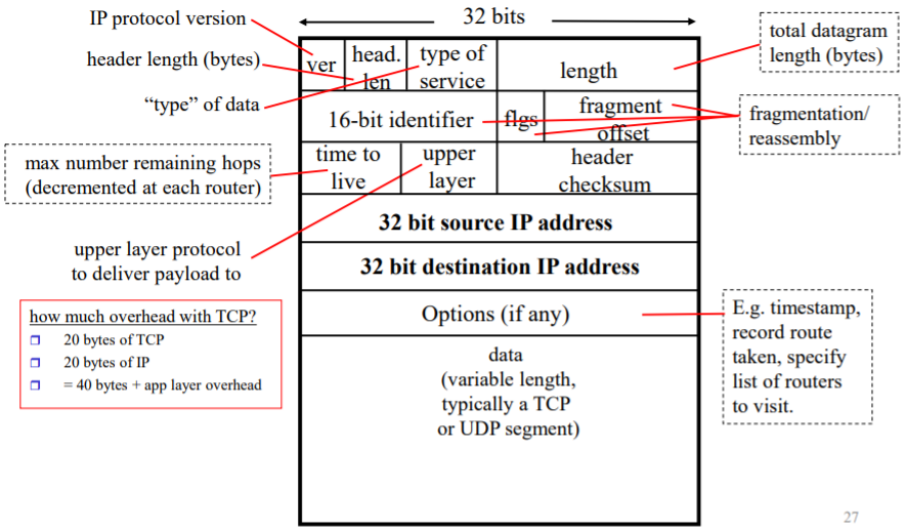
a) Elimina a trama.
b) Invoca um procedimento do *Address Resolution Protocol* (ARP).
c) Envia a trama para todas as portas ativas exceto a porta através da qual a trama foi recebida.
d) Envia a trama para através da porta ligada ao *default gateway* do *Switch*.

6.8 Protocolo Internet

1. Camada de rede Internet



2. Formato datagrama IP



3. Internet Checksum

Internet Checksum

- The Internet (not layer 2) uses a checksum
  - » easily implementable in software
  - » 1's complement sum of 16 bit words
  - » Performance: d=2

```
u_short
cksum(u_short *buf, int count)
{
    register u_long sum = 0;

    while (count--)
    {
        sum += *buf++;
        if (sum & 0xFFFF0000)
        {
            /* carry occurred,
               so wrap around */
            sum &= 0xFFFF;
            sum++;
        }
    }
    return ~(sum & 0xFFFF);
}
```

- One's complement sum
  - » Mod-2 addition with carry-out
  - » Carry-out in the most-significant-bit is added to the least-significant bit
  - » Get one's complement of "one's complement sum"

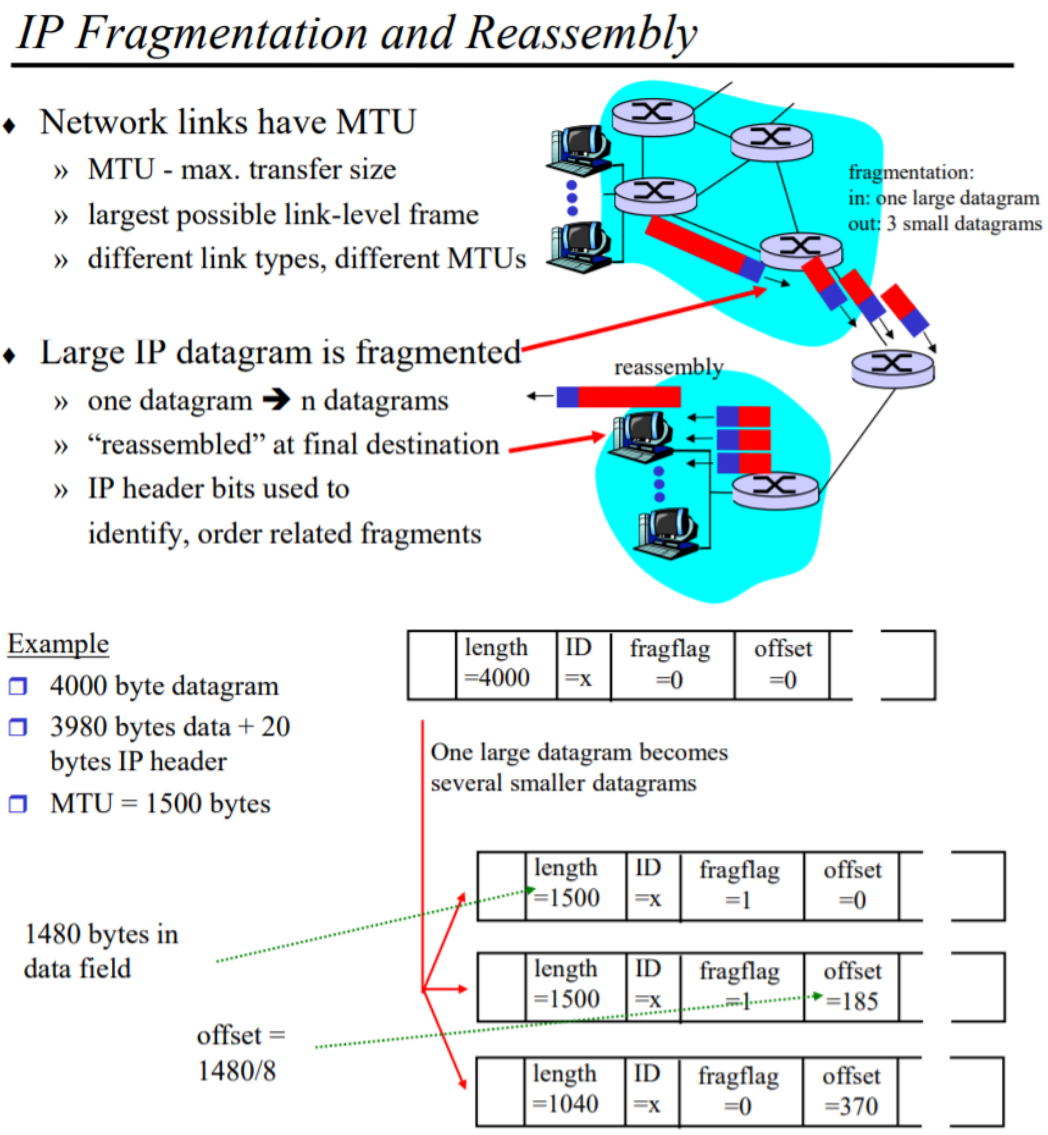
```
                    1010011
                    0110110
carry-out (1)       0001001
Carry wrap-around   0000001
                    0001010
One's complement = 1110101
```

6.8.1 Cada pacote contém:

- Versão do protocolo IP
- Tamanho do Header
- Tipo de serviço
- Tamanho da informação
- Identificador + Flags + Offset de Fragmento (Permite fragmentar mensagens em vários pacotes)
- Time To Live (para os pacotes não ficarem indefinidamente perdidos na rede)
- Upper Layer Protocol
- Checksum do Header
- IP de Origem
- IP de Destino
- Opções (opcional)
- Informação (Normalmente pacote TCP ou UDP)

6.8.2 Fragmentação IP e Reassembly

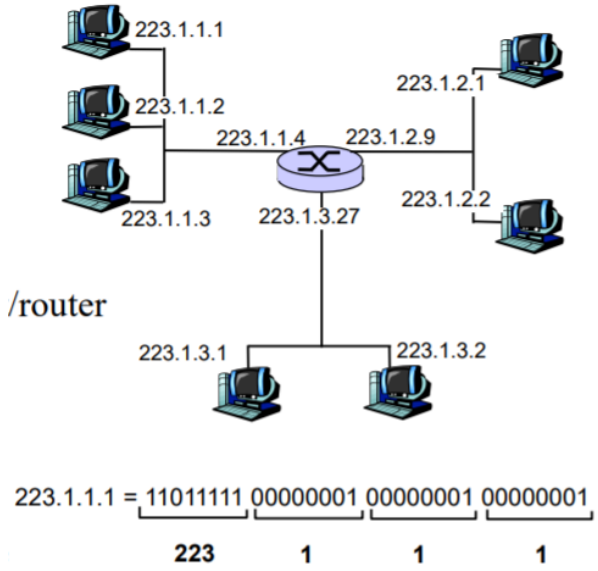
- Identificador <- Identifica o pacote
- fragflag <- 1 se houver mais informação, 0 se for o último fragmento
- Offset <- Offset do fragmento em bytes / 8



6.8.3 Endereço IP

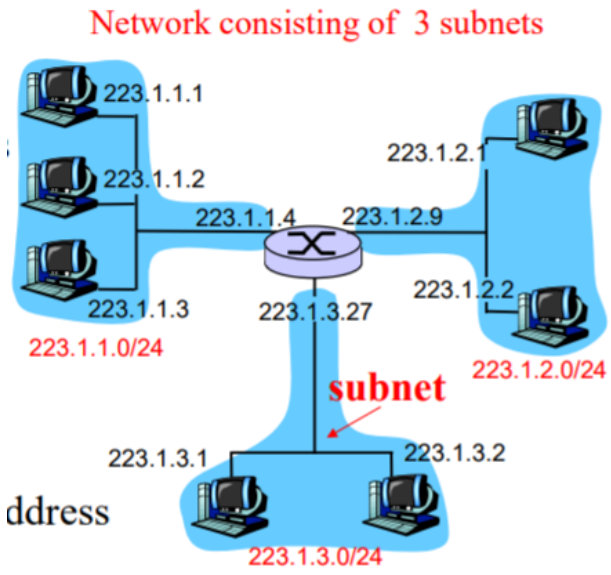
Endereço IP - é formado por um identificador de 32-bit para uma interface host/router Interface possuem:

- conecção entre host/router e link físico(physical link)
- routers com multiplas interfaces
- endereços IP associados com as interfaces



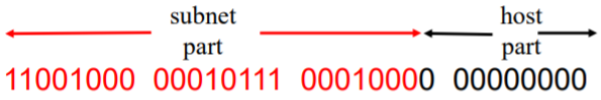
6.8.4 Subnets

- Parte mais significativa do IP: Subnet parte
- Parte menos significativa: host(interface) parte
- Subnet é um set de interfaces
- cada um tem a subnet parte do IP igual para comunicação
- Cada computador consegue aceder a outro sem intervenção do router



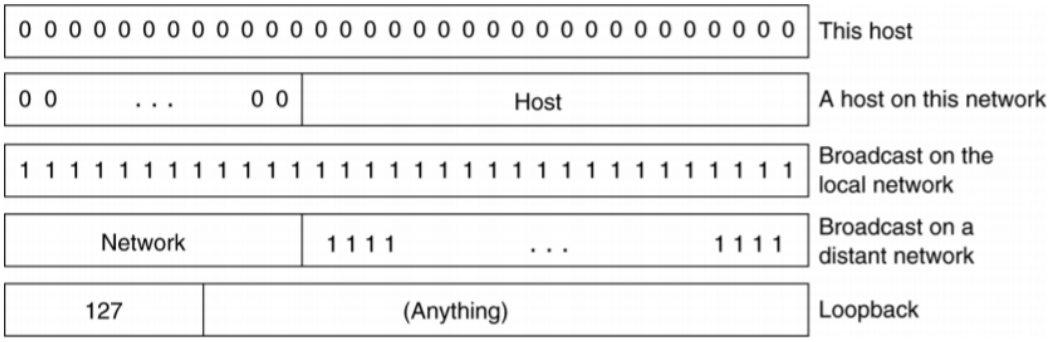
CIDR - Classless InterDomain Routing

- a porção de bits do endereço subnet tem tamanho arbitrário
- formato -> a.b.c.d/x, em que x é o número de bits na porção do endereço subnet



200.23.16.0/23

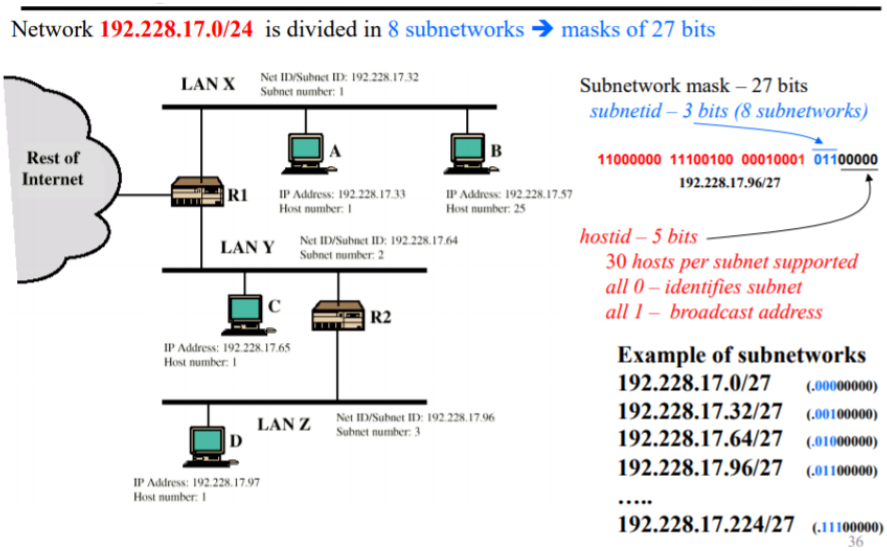
6.8.5 Endereços especiais



- 0.0.0.0 - este host
- 127.0.0.0 - loopback
- 255.255.255.255 - broadcast
- x.x.255.255 - broadcast na subnet x.x.0.0/16
- x.x.0.0 - subnet x.x.0.0/16

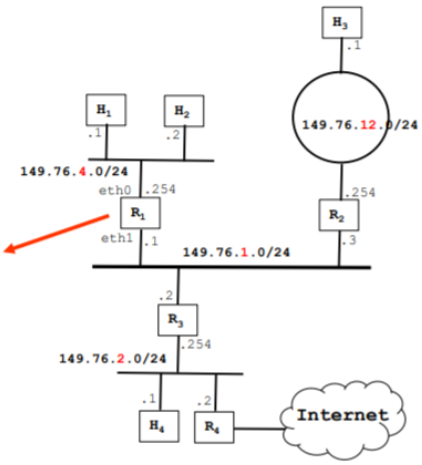
De Notar: - Uma subrede xx.xx.xx.0/24 suporta 255 endereços, no entanto, dois já estão reservados (xx.xx.xx.0 e xx.xx.xx.255), logo só suporta 253 máquinas.

Forming Sub-Networks (importante)

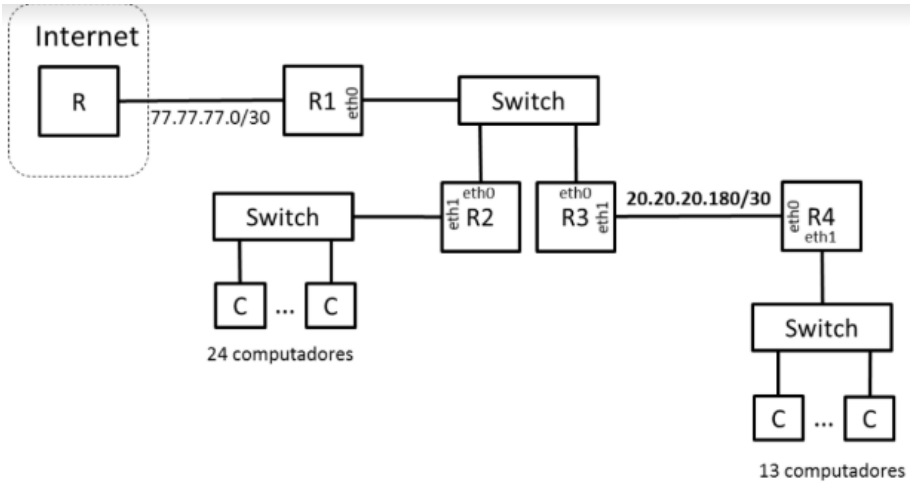


Criar table em R1 (importante)

Destination	Gateway	Interface
149.76.1.0/24	-	eth1
149.76.2.0/24	149.76.1.2	eth1
149.76.4.0/24	-	eth0
149.76.12.0/24	149.76.1.3	eth1
0/0	149.76.1.2	eth1



Exercício: Exame 2018R-ultima questao (3) Considere que a uma empresa foi atribuído o bloco de endereços IP 20.20.20.128/26. A empresa tem um rede de comunicações com a arquitetura descrita na figura, composta por 4 routers (R1, R2, R3, R4) e 3 switches Ethernet. Um dos switches serve 24 computadores, outro serve 13 computadores e o terceiro interliga os routers R1, R2 e R3. Os routers R3 e R4 estão interligados por uma ligação ponto-a-ponto, à qual foi atribuído o endereço de rede 20.20.20.180/30.



a) Calcule os endereços de rede associados às redes indicadas

	Endereço da subrede (endereço/máscara)	Endereço de *broadcast* da subrede	Nº de endereços de interfaces
Rede dos 24 computadores	20.20.20.128/27	20.20.20.159	30
Rede dos 13 computadores	20.20.20.160/28	20.20.20.175	14
Rede dos routers R1, R2 e R3	20.20.20.184/29	20.20.20.191	6

b) Atribua endereços IP às interfaces dos routers R1, R2, R3 e R4. Use os endereços mais baixos de cada sub-rede. Numa sub-rede atribua os endereços mais baixos aos routers de índice Ri mais baixo. Por exemplo, o endereço de R3.eth1 deverá ser inferior ao endereço R4.eth0.

Router.interface	Endereço(s) IP
R1.eth0	20.20.20.185
R2.eth0	20.20.20.186
R2.eth1	20.20.20.129
R3.eth0	20.20.20.187
R3.eth1	20.20.20.181
R4.eth0	20.20.20.182
R4.eth1	20.20.20.161

c) Escreva a tabela de encaminhamento do router R2. Este router deverá ser capaz enviar pacotes para todos os endereços IP unicast. Use o menor número possível de entradas na tabela.

Destino (endereço/máscara)	Gateway	Interface
20.20.20.128/27	-	eth1
20.20.20.184/29	-	eth0
20.20.20.180/30	20.20.20.187	eth0
20.20.20.160/28	20.20.20.187	eth0
0/0	20.20.20.185	eth0

Fazer pergunta 3 (last question) de todos os exames. É igual, apenas alterando os valores dos IPs

função IP forwarding (importante)

- Forwarding table has entries in format
  `<networkAddress/mask, port>`
- Forwarding function
  - When a datagram arrives with destination address A, then
    - For each entry of the forwarding table
      - `val= A & mask*` (e.g., mask=8, mask*=255.0.0.0)
      - if `( val == networkAddress` & mask*)
        - add corresponding output port to the set of candidate ports
    - Select the port with the largest mask → most specific route
  - Example
    - `frdTbl={<128.32.1.5/16,1>, <128.32.225.0/18,3>, <128.0.0.0/8,5>}`
    - Datagram with destination address `A=128.32.195.1`
    - Set of candidate output ports → `{1,3,5}.`
    - Selected port → 3 ← largest mask, 18 bits

Exame 2018N - Exercicio 7

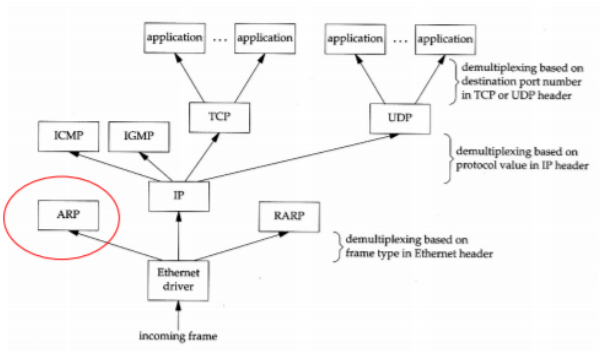
7. Admita que a tabela de encaminhamento de um router IP contém entradas no formato
<endereçoRede/máscara, portaSaída> e que a tabela contém as seguintes entradas
{<222.0.0.0/8, 1>, <222.0.0.0/16, 2>, <222.0.128.0/18, 3>}.
Assuma que ao router chega um pacote com o endereço de destino **222.0.127.8**. Nesta situação o pacote

a) É encaminhado para a porta 1.
b) É encaminhado para a porta 2.
c) É encaminhado para a porta 3.
d) É eliminado.

6.9 Address Resolution Protocol APR

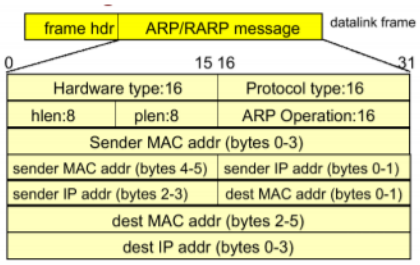
Demultiplexing

- Ethernet header (type)
  - » IP - 0x0800
  - » ARP - 0x0806
  - » RARP - 0x8035
  - » IPX- 0x8037
  - » IPv6 - 0x86DD
  - » MPLS - 0x8847
- IP header (protocol)
  - » ICMP - 1
  - » IGMP - 2
  - » TCP - 6
  - » UDP – 17
- TCP/UDP header (port)
  - » FTP – 21
  - » Telnet - 23
  - » HTTP - 80
  - » SMTP - 25

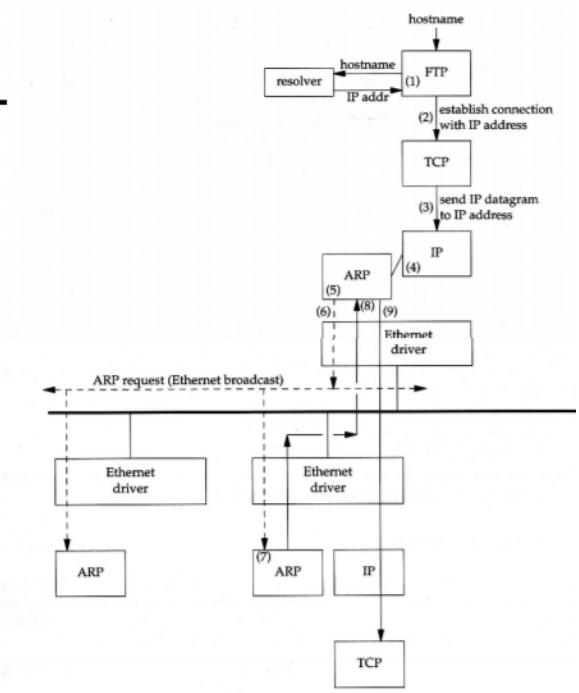


- Uma interface de rede tem 1 endereço MAC e 1 (ou mais) endereços IP
- ARP - protocolo usado para obeter o endereço MAC associado a um endereço IP dado
- RARP - reverso de ARP - protocolo usado para obter o endereço IP associado ao endereço MAC

ARP Example



- **hardware type : Ethernet=1 ARCNET=7, localtalk=11**
- **protocol type : IP=0x800**
- **hlen : length of hardware address, Ethernet=6 bytes**
- **plen : length of protocol address, IP=4 bytes**
- **ARP operation : ARP request = 1, ARP reply = 2**
  RARP request = 3, RARP reply = 4



6.10 Obter endereço IP

- Parte do endereço da subnet é definido pelo ISP

ISP's block	11001000 00010111 00010000	00000000	200.23.16.0/20
Organization 0	11001000 00010111 00010000	00000000	200.23.16.0/23
Organization 1	11001000 00010111 00010010	00000000	200.23.18.0/23
Organization 2	11001000 00010111 00010100	00000000	200.23.20.0/23
...	.....	....	....
Organization 7	11001000 00010111 00011110	00000000	200.23.30.0/23

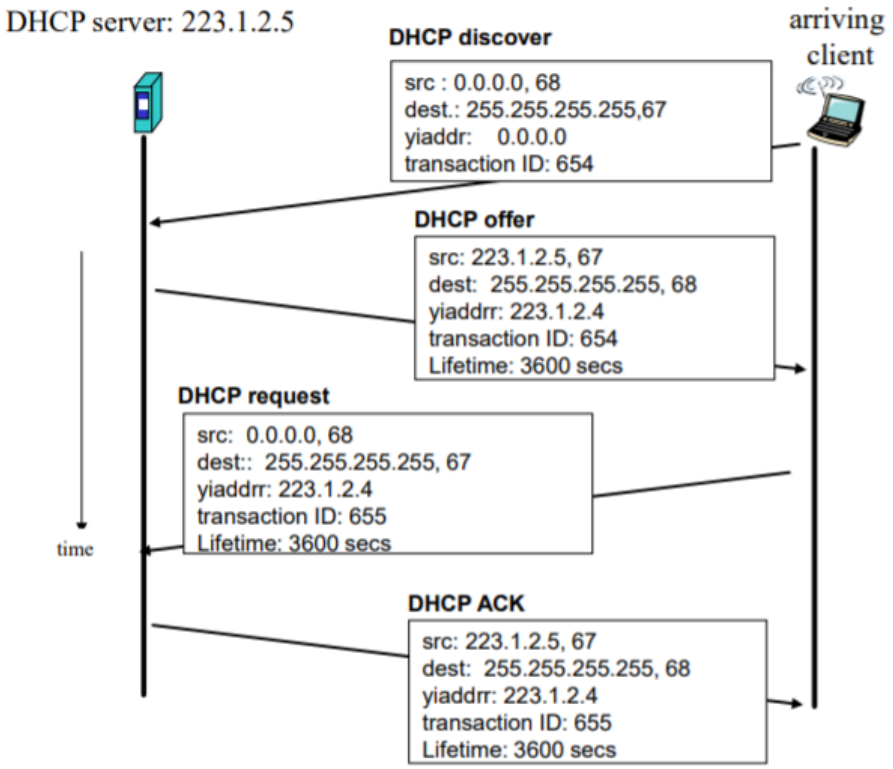
- endereçamento hierárquico permite eficiência da informação do router
- O ISP depois trata internamente das suas subredes
- O ISP obtém endereços pela ICANN
- ICANN: Internet Corporation for Assigned Names and Numbers
  - aloca endereços

  - controla o Domain Name Service (DNS)
  - associa os nomes do domínio
  - resolve conflitos

- o host obtém endereços IP de forma hard-coded pelo sistema admin num ficheiro ou pelo DHCP

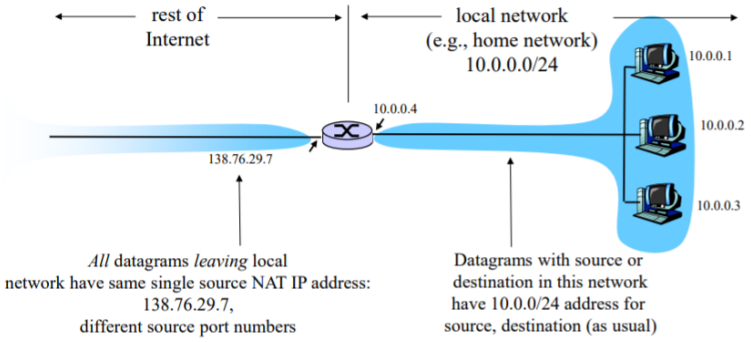
- DHCP: Dynamic Host Configuration Protocol
  - Dinamicamente recebe endereços do servidor
  - "plug-and-play"
  - permite descobrir e obter endereços da rede do servidor
  - reusa os endereços
  - Overview:
    * O host faz broadcast de "DHCP discover"(msg)
    * O servidor DHCP oferece um endereço, enviando em broadcast "DHCP offer"(msg)
    * O host pede esse endereço enviando em broadcast "DHCP request"(msg)
    * Se tudo estiver em ordem, o DHCP responde em broadcast com um "DHCP ACK"(msg)
    * Todas as mensagens entre o host e o DHCP possuem um id de transação

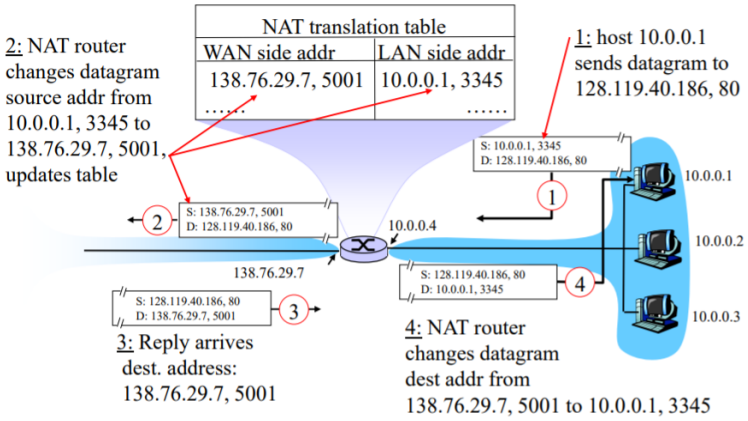
(Com o seguinte gráfico analisar pergunta 7 do exame 2018N)



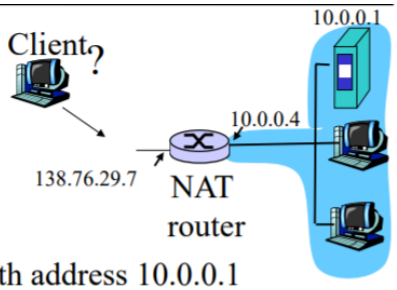
6.11 NAT - Network Address Translation

- Permite que cada computador tenha um IP interno numa rede, sendo o IP externo diferente
- Para isso, possui uma hash table a que associa um IP interno e uma porta a um número, que será a porta de saída
- Caso um cliente se queira ligar a um servidor dentro de uma rede com NAT, é necessário configurar o port forwarding





6.11.1 NAT Transversal



- Client wants to connect to server with address 10.0.0.1
  - » but server address 10.0.0.1 is private
  - » only one externally visible NATed address: 138.76.29.7
- Possible solution – Port forwarding
  - » statically configure NAT
    to forward incoming connection requests at given port to server
  - » e.g., (138.76.29.7, port 2500) always forwarded to 10.0.0.1 port 25000

6.11.2 Question 8 - Exame 2017N

8. Assuma que a tabela NAT de um router tem a seguinte entrada < (140.76.29.6, 80), (10.0.1.4, 8080) >. A rede privada tem o endereço 10.0.0.0/16 e existe um servidor HTTP na porta 8080 da máquina com o endereço 10.0.1.4. Nesta situação, os endereços IP e TCP de origem de um pacote observado na rede privada para este servidor são os seguintes

a) IP=140.76.29.6, Port= 80.
b) IP=140.76.29.6, Port= 8080.
c) Os endereços IP e TCP da máquina da rede pública que está a contactar o servidor.
d) Nenhuma das anteriores.

6.12 ICMP - Internet Message Control Protocol

- Usado pelo router ou host para mandar mensagens de erro ou de controlo (como o ping)

6.12.1 Exame 2017N Ex:1

1. O programa ***ping*** usado nas aulas laboratoriais gera pacotes de informação do
a) protocolo UDP, que por usa vez são encapsulados em pacotes IP, que por sua vez são encapsulados em tramas Ethernet.
b) protocolo ICMP, que por usa vez são encapsulados em pacotes IP, que por sua vez são encapsulados em tramas Ethernet.
c) protocolo IP, que por sua vez são encapsulados em tramas Ethernet.
d) protocolo ARP, que por sua vez são encapsulados em tramas Ethernet.

6.12.2 Exame 2018N Ex:2

2. O protocolo Internet Control Message Protocol (ICMP) usa serviços oferecidos pelo protocolo
a) TCP.
b) UDP.
c) IP.
d) Ethernet 802.3.

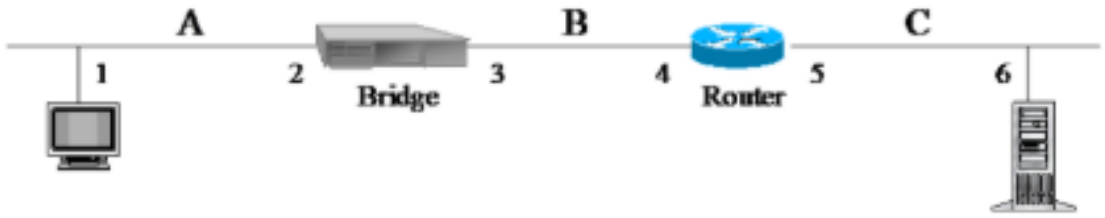
6.12.3 Exame 2016 Recurso - Ex:7

7. Assuma o seguinte cenário de ligações: $[C_1]$—$_0$[S]$_1$—$_0$[R]$_1$—$[C_2]$. Neste cenário o computador C_1 está ligado à porta 0 do switch S, a porta 1 do switch S está ligada à porta 0 do router R, e o computador C_2 está ligado diretamente à porta 1 do router R. Nesta situação, quando o **computador C_1 envia um pacote IP com destino ao computador C_2**, os endereços IP e MAC de **origem** constantes do pacote **recebido por C_2** são:
a) Endereço IP de C_1, endereço MAC de C_1.
b) Endereço IP de C_1, endereço MAC de R.porta$_1$
c) Endereço IP de R.porta$_1$, endereço MAC de C_1.
d) Endereço IP de R.porta$_1$, endereço MAC de R.porta$_1$.

6.12.4 Exame 2013 Recurso - Ex:7

7. Assuma o seguinte cenário de ligações: $[C_A]$—$[L_A]$—$_0$$[R_{NAT}]_{1,pub}$—$[L_B]$—$[C_B]$. Neste cenário, o computador C_A está ligado à porta 0 do router R_{NAT} através da LAN L_A e o computador C_B está ligado à porta 1 do router R_{NAT} através da LAN L_B. O router R_{NAT} implementa NAT e a sua porta 1 encontra-se ligada à Internet pública. Nesta situação, quando o computador **C_A envia um pacote de dados para o computador C_B**, os **endereços IP e MAC de origem** constantes do **pacote recebido em C_B** são os seguintes:
a) Endereço IP de C_A, endereço MAC de C_A.
b) Endereço IP de C_A, endereço MAC de R_{NAT}.porta$_1$.
c) Endereço IP de R_{NAT}.porta$_1$, endereço MAC de C_A.
d) Endereço IP de R_{NAT}.porta$_1$, endereço MAC de R_{NAT}.porta$_1$.

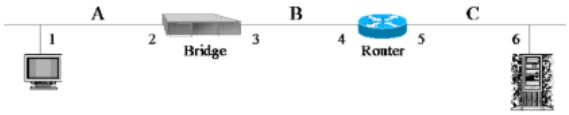
6.12.5 se o computador do segmento C fizer ping ao Computador do segmento A, indique os endereços IP e MAC constantes do pacote que transporta a mensagem ICMP Echo Request no segmento A. (Questão 2018R-ex:9)



R: IP origem = 6, IP destino = 1, MAC origem = 4, MAC destino = 1.

6.12.6 Exame 2014 Normal - Ex:9)

9. Na Figura seguinte, se o computador do segmento C fizer *ping* ao Computador do segmento A, indique os endereços IP e MAC constantes do pacote que transporta a mensagem ICMP Echo Reply no segmento B.



a) IP_{orig}=1, IP_{dest}=6, MAC_{orig}=1, MAC_{dest}=4.
b) IP_{orig}=1, IP_{dest}=6, MAC_{orig}=3, MAC_{dest}=4.
c) IP_{orig}=1, IP_{dest}=4, MAC_{orig}=1, MAC_{dest}=4.
d) IP_{orig}=1, IP_{dest}=4, MAC_{orig}=3, MAC_{dest}=4.

6.12.7 Exame 2010 Normal - Ex:4 e 5)

4. Admita que uma *bridge* transparente Ethernet / IEEE 802.3 recebe uma trama MAC com endereço de destino que não está presente na sua tabela de comutação (*forwarding table*). Neste caso a *bridge*:

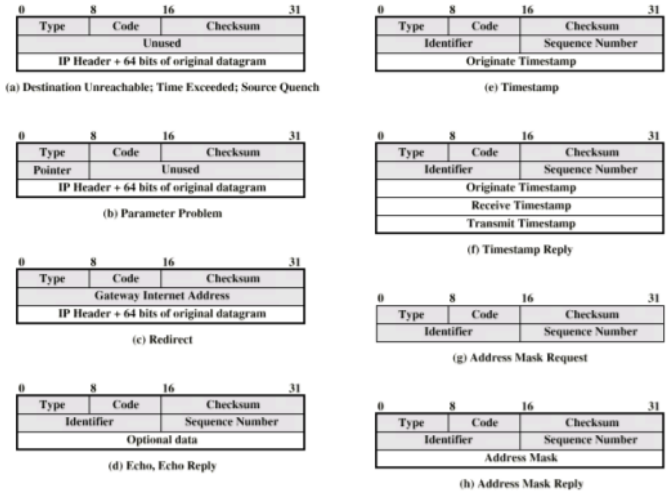
a) Transmite uma cópia inalterada da trama em todas as portas, com excepção da porta onde foi recebida.
b) Transmite uma cópia da trama em todas as portas, com excepção da porta onde foi recebida, após alterar o endereço de destino para *broadcast*.
c) Descarta a trama.
d) Retém a trama temporariamente, inicia um processo de resolução de endereços para localizar a estação de destino e, em caso de sucesso, actualiza a tabela de comutação e envia a trama pela porta correspondente.

5. No protocolo TCP o emissor controla uma janela de congestionamento; no início da sessão TCP ou após *time-out* entra-se numa fase de *slow start*, que é seguida, após se atingir um limiar, por uma fase de *congestion avoidance*.

a) A janela do emissor aumenta durante *slow start* e mantém-se constante durante *congestion avoidance*.
b) A janela do emissor aumenta mais rapidamente durante *slow start* do que durante *congestion avoidance*.
c) A janela do emissor aumenta mais lentamente durante *slow start* do que durante *congestion avoidance*.
d) A janela do emissor aumenta rapidamente durante *slow start*; ao entrar na fase de *congestion avoidance* a janela é reduzida a metade, após o que aumenta mais lentamente até se atingir de novo o limiar (e o processo repete-se).

6.12.8 IP datagramas info:

- Carried in IP datagrams



Type	Code	Description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
5		Redirect
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

62

6.12.9 Tracerout and ICMP

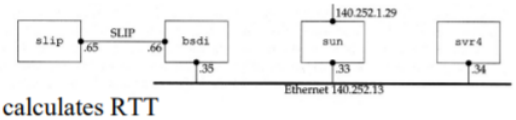
- Permite fazer traceroute enviando mensagens com TTL=1,2,3... e esperando respostas de erro "TTL expired"até receber um "Host unreachable"

- Source sends series of UDP segments to destination
  - » first segment has TTL =1
  - » second segment has TTL=2, …
  - » unlikely port number
- When n^{th} datagram arrives

  to n^{th} router
  - » router discards datagram
  - » sends to source: ICMP TTL expired
  - » message includes router name & IP address

```
svr4% traceroute slip
traceroute to slip (140.252.13.65), 30 hops max. 40 byte packets
1 bsdi (140.252.13.35) 20 ms 10 ms 10 ms
2 slip (140.252.13.65) 120 ms 120 ms 120 ms

slip% traceroute svr4
traceroute to svr4 (140.252.13.34), 30 hops max, 40 byte packets
1 bsdi (140.252.13.66) 110 ms 110 ms 110 ms
2 svr4 (140.252.13.34) 110 ms 120 ms 110 ms
```



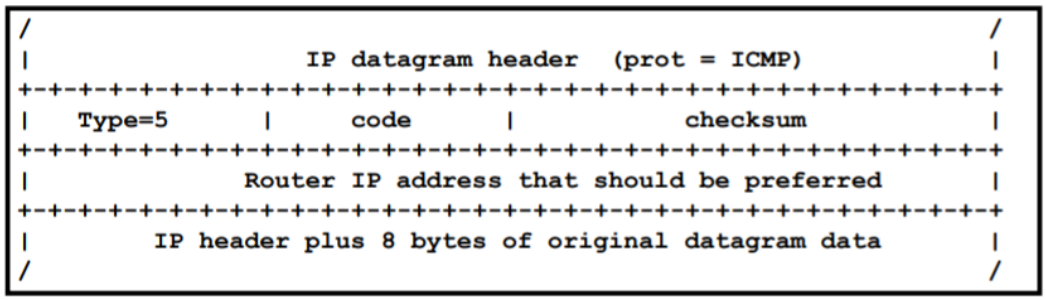
- When ICMP message arrives, source calculates RTT
- Traceroute does this 3 times for each TTL
- Stop criterion
  - » UDP segment eventually arrives at destination host
  - » Destination returns ICMP "dest port unreachable" packet
  - » source stops

6.12.10 ICMP Redirect

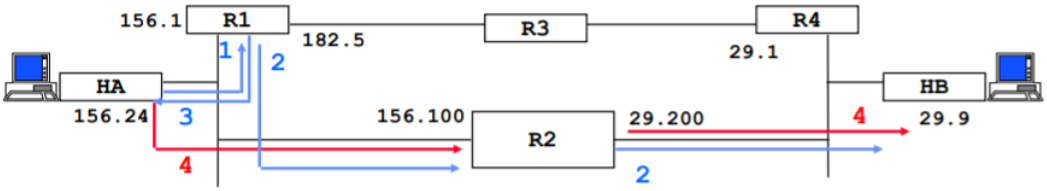
- ICMP Redirect - Permite informar outros hosts do caminho mais rápido para determinado destino

- General routing principle of the TCP/IP architecture
  - » routers have extensive knowledge of routes
  - » hosts have minimal routing information ➔ learn routes also from ICMP redirects
- ICMP redirect message
  - » Sent by router R1 to source host A
    - – when R1 receives a packet from A with destination = B, and R1
      - ◆ finds that the next hop is R2 and
      - ◆ A is on-link with R2
  - » R1 sends ICMP redirect to A saying next hop for destination B is R2
  - » A updates its forwarding table with a host route

ICMP Redirect Format



ICMP Redirect Example



	dest IP addr	srce IP addr	prot	data part
1:	193.154.29.9	193.154.156.24	udp	xxxxxxx
2:	193.154.29.9	193.154.156.24	udp	xxxxxxx
3:	193.154.156.24	193.154.156.1	icmp	type=redir code=host cksum 193.154.156.100 xxxxxxx (28 bytes of 1)
4:	193.154.29.9	193.154.156.24	udp	.....

After 4

```
HA$ netstat -nr
Routing Table:
  Destination           Gateway           Flags  Interface
-------------------- -------------------- ----- ---------
127.0.0.1            127.0.0.1             UH     lo0
193.154.29.9         193.154.156.100       UGH    eth0
193.154.156.0        193.154.156.24        U      eth0
224.0.0.0            193.154.156.24        U      eth0
default              193.154.156.1         UG     eth0
```

Flags:
U - *route Up*
G - *route to a Gateway (next hop router)*
H - *route to a Host*

67

6.13 IPv6

- IPv4
  - espaço reduzido de endereçamento (32 bits)
  - uso não continuo
  - o uso de algumas soluções como private networks (NAT) e classless networks (CDIR) superava os problemas acima

- IETF developed new IP version: IPv6
  - Uso dos mesmos princípios do IPv4
  - muitas melhorias
  - Header foi redefinido

6.13.1 IPv6 - Melhorias

- Endereços 128 bits (16 octets, 8 shorts). No classes
- Melhor QoS suporte (native flow level)
- funções nativas de segurança (autenticação, data encriptação)
- Autoconfiguração (Plug-n-play)
- Routing
- Multicast

6.13.2 Representação dos endereços

- 8 x 16 bit, hexadecimal, separados por:
  47CD : 1234 : 3200 : 0000 : 0000 : 4325 : B792 : 0428
- formato comprimido:
  FF01:0:0:0:0:0:0:43 -> FF01::43
- compatibilidade com IPv4:
  0:0:0:0:0:0:13.1.68.3 or ::13.1.68.3
- Loopback endereço:
  ::1
- Prefixo de rede "/", igual ao IPv4:
  FEDC:BA98:7600::/40 -> network prefix = 40 bits

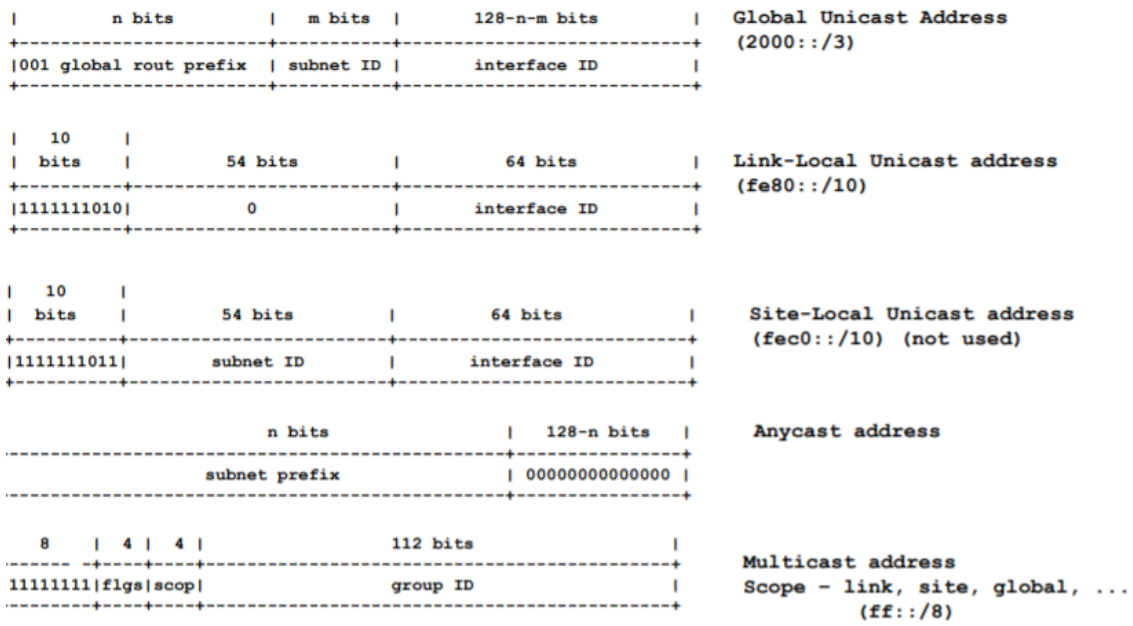
6.13.3 Endereços Reservados

Allocation	Prefix (binary)	Fraction of Address Space
Unassigned	0000 0000	1/256
Unassigned	0000 0001	1/256
Reserved for NSAP Allocation	0000 001	1/128
Unassigned	0000 01	1/64
Unassigned	0000 1	1/32
Unassigned	0001	1/16
Global Unicast	001	1/8
Unassigned	010	1/8
Unassigned	011	1/8
Unassigned	100	1/8
Unassigned	101	1/8
Unassigned	110	1/8
Unassigned	1110	1/16
Unassigned	1111 0	1/32
Unassigned	1111 10	1/64
Unassigned	1111 110	1/128
Unassigned	1111 1110 0	1/512
Link-Local Unicast Addresses	1111 1110 10	1/1024
Site-Local Unicast Addresses	1111 1110 11	1/1024
Multicast Addresses	1111 1111	1/256

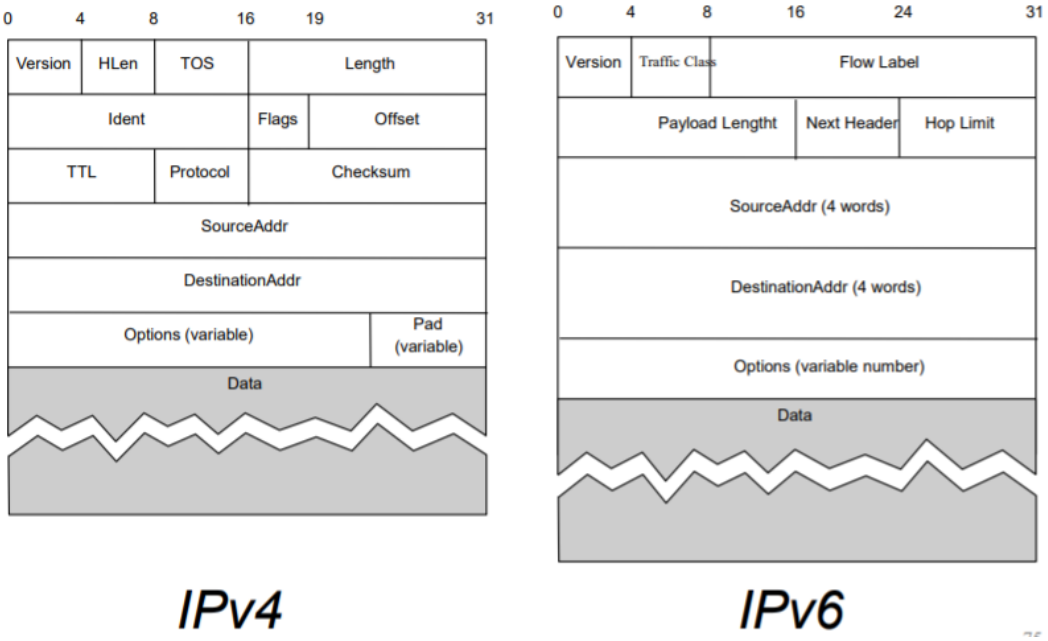
6.13.4 Tipo de Endereços

- Link-Local
  - Usado para a comunicação entre hosts na mesma LAN/link
  - Endereço criado pelo endereço MAC
  - Routers nao enviam pacotes tendo endereços de destino Link-Local
- Global Unicast
  - Endereços globais
  - Endereços: prefixo de rede + identificador do computador
  - Prefixos estruturados: Agregação de rede; menos entradas nas router forwarding tables
- Anycast
  - Endereços de grupo
  - Um pacote é recebido por um e um só membro do grupo
- Multicast
  - Endereços de grupo
  - Um pacote pode ser recebido por vários membros do grupo

6.13.5 Formato dos Endereços



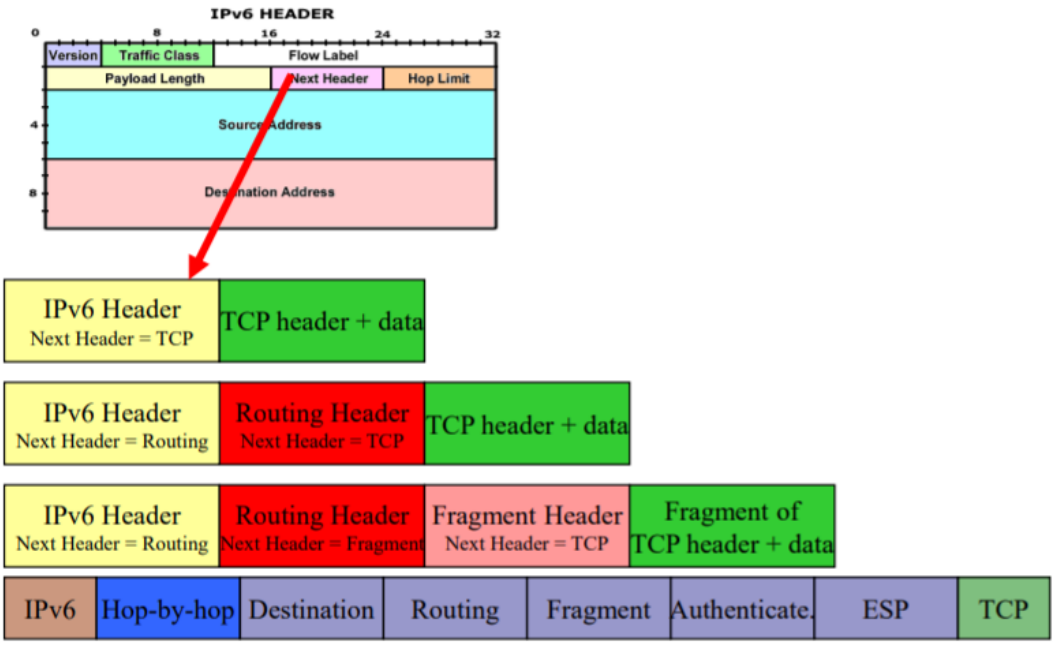
6.13.6 Headers IPv4 e IPv6



6.13.7 IPv6 Header

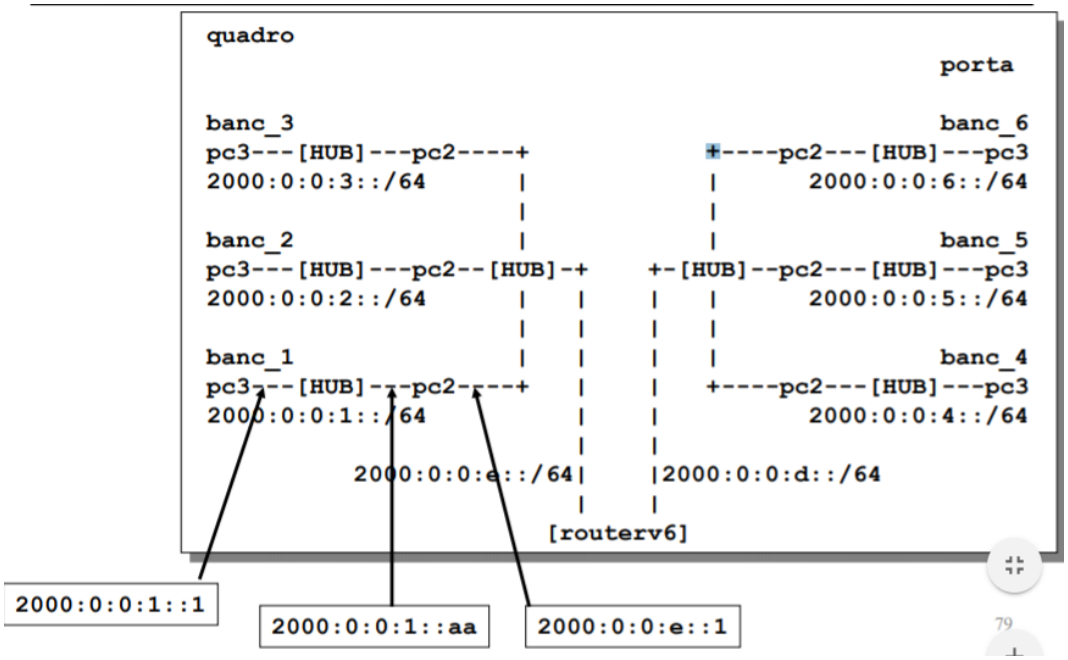
- Flow label - identifica o fluxo do pacote
  - QoS, ressalva de recursos
  - Pacotes recebem o mesmo serviço
- Payload lenght - Header não incluído
- Next header - identifica o próximo header/extensão
- Options - incluída nas extensões dos headers

6.13.8 Extension Headers



- Hop-by-Hop: inspeciona todos os nodes atravessados pelo pacote
- Destination: informação do node de destino
- Routing: Lista dos nodes para serem visitados pelo pacote
- Fragmentation: feito pelo source, deve encontrar MPU
- Authentication: autenticação (assinatura) do header do pacote
- ESP: encriptação da informação(data)

6.13.9 Exemplo da Rede do Laboratório



6.13.10 Protocol Neighbor Discovery (ND)

IPv6 node usa ND para:

- Encontrar outros nodes no mesmo link/LAN
- Encontrar o node do endereço MAC (ND substitui ARP)
- Encontrar routers na sua rede
- Manter/Segurar a informação sobre os nodes vizinhos

ND similar às funções IPv4:

- ARP IPv4
- ICMP Router Discovery
- ICMP Redirect

6.13.11 ND Mensagens

- ICMP mensagens (over IP), Uso de endereços Link Local
- **Neighbor Solicitation:** Enviado pelo host para obter o endereço MAC de um vizinho/para verificar a sua presença

- **Neighbor Advertisement:** resposta ao pedido
- **Router Advertisement:** Informação sobre o prefixo da rede, periodica ou abaixo do pedido. Enviado pelo router para o endereço IP do Link Local multicast
- **Router Solicitation:** Hosts solicitam do router uma mensagem Router Advertisment
- **Redirect:** Usado pelo router para informar o host acerca da melhor rota para o destino

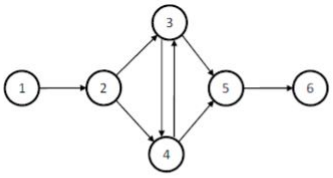
7 Transport Layer

Hello, here is some text without a meaning. This...

8 Routing

8.0.1 Graphs

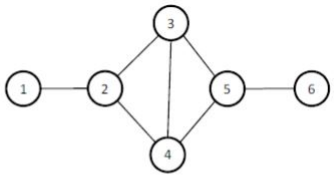
Directed:



a) Directed graph

$$
\begin{array}{lll}
G = & (V, E) & \\
V = & \{v_1, v_2, v_3, v_4, v_5, v_6\}, & |V| = 6 \\
E = & \{(v_1, v_2), (v_2, v_3), (v_2, v_4), (v_3, v_4), & \\
 & (v_4, v_3), (v_3, v_5), (v_4, v_5), (v_5, v_6)\}, & |E| = 8
\end{array}
$$

Undirected:

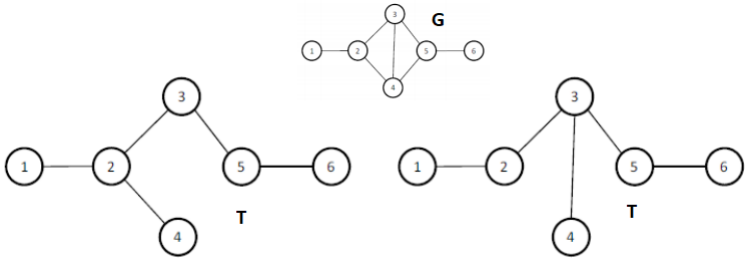


b) Undirected graph

$$
\begin{aligned}
G = \ & (V, E) \\
V = \ & \{v_1, v_2, v_3, v_4, v_5, v_6\}, && |V| = 6 \\
E = \ & \{(v_1, v_2), (v_2, v_3), (v_2, v_4), (v_3, v_4), \\
& (v_3, v_5), (v_4, v_5), (v_5, v_6)\}, && |E| = 7
\end{aligned}
$$

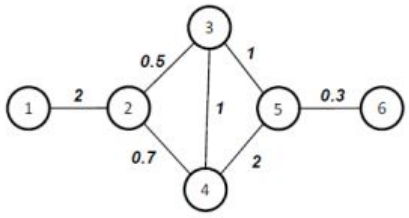
8.0.2 Tree

- Tree T = (V,E)
  - Graph with no cycles
  - |E| = |V| - 1
  - Any two V connected by only one E
- A tree T spans a graph G = (V,E) (spanning tree) if
  - T = (V,E') & E ⊆ E' (T must have the same vertices and a subset of the graph edges)



8.0.3 Shortest Path Trees

- Graphs and Trees can be weighted
  - G=(V,E,W)
  - T=(V,E',W)



- Total cost of a tree T

$$C_{total}(T) = \sum_{i=1}^{|E|}$$

(sum of all tree edges weight)

- Minimum spanning tree T^*

$$C_{total}(T^*) = min(C_{total}(T))$$

  - algorithms used to compute MST: Prism, Kruskal

- **Shortest Path Tree (SPT) rooted at vertex** s
  - tree composed by the **union of the shortest paths between** s **and each vertex of** G
  - algorithms used to compute SPT: **Dijkstra**, **Bellman-Ford**
- Computer networks use **Shortest Path Trees**

8.1 Routing in Layer 3 Networks

8.1.1 Forwarding, Routing

- **Forwarding** → data plane
  - directing packet from input to output link
  - using a forwarding table
- **Routing** → control plane
  - computing paths the packets will follow
  - routers exchange messages
  - each router creates its forwarding table

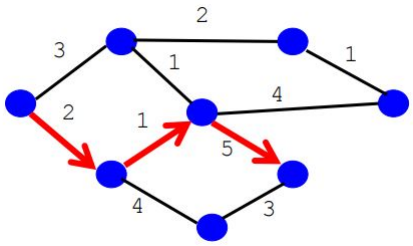
8.1.2 Importance of Routing

- End-to-end performance
  - path affects quality of service
  - delay, throughput, packet loss
- Use of network resources
  - balance traffic over routers and links
  - avoiding congestion by directing traffic to less-loaded links
- Transient disruptions
  - failures, maintenance
  - limiting packet loss and delay during changes

8.1.3 Shortest-Path Routing

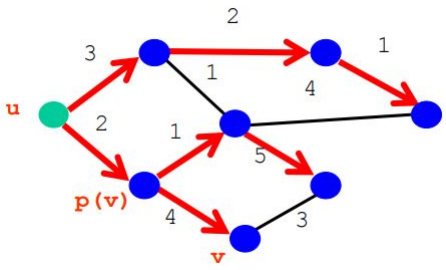
Path-selection model

- Destination-based
- Load-insensitive (ex: static link weights)
- Minimum hop count or minimum sum of link weights



8.1.4 Shortest-Path Problem

- Given a network topology with link costs
  - **c(x,y)** - link cost from node x to node y
  - ∞ if x and y are not direct neighbors
- Compute the least-cost paths from source **u** to all nodes
  - **p(v)** - node predecessor of node v in the path from u



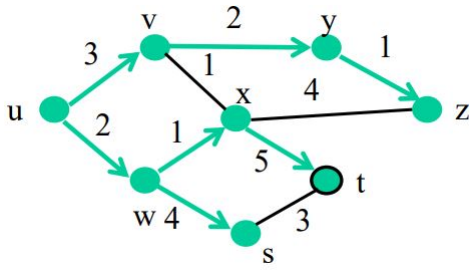
8.1.5 Dijkstra's Shortest-Path Algorithm

- Iterative algorithm
  - After k iterations → known least-cost paths to k nodes
- **S** → set of nodes for which least-cost path is known
  - Initially, **S**=**{u}**, where **u** is the source node
  - Add one node to **S** in each iteration
- **D(v)** → current cost of path from source to node **v**
  - Initially
    - **D(v)**=**c(u,v)** for all nodes adjacent to **u**
    - **D(v)**=∞ for all other nodes **v**
  - Continually update **D(v)** when shorter paths are learned

```
1  Initialization:
2    S = {u}
3    for all nodes v
4      if v adjacent to u {
5          D(v) = c(u,v) }
6      else D(v) = ∞
7
8  Loop
9    find node w not in S with the smallest D(w)
10   add w to S
11   update D(v) for all v adjacent to w and not in S:
12       D(v) = min{D(v), D(w) + c(w,v)}
13 until all nodes in S
```

8.1.6 Shortest-Path Tree

- Shortest-path tree from u



- Forwarding table at u

	link
v	(u,v)
w	(u,w)
x	(u,w)
y	(u,v)
z	(u,v)
s	(u,w)
t	(u,w)

8.1.7 Link-State Routing

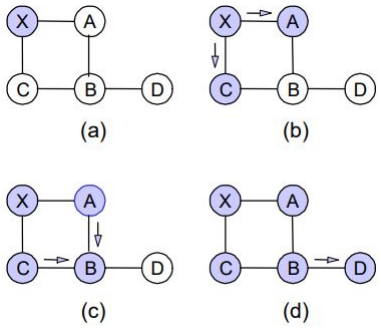
- Each router keeps track of its incident links
  - link up, link down
  - cost on the link
- Each router broadcasts link state
  - every router gets a complete view of the graph
- **Each router runs Dijkstra's algorithm**, to
  - compute the shortest paths
  - construct the forwarding table

8.1.8 Detection of Topology Changes

- Beacons generated by routers on links
  - periodic "hello" messages in both directions
  - few missed "hellos" → link failure

8.1.9 Broadcasting the Link State

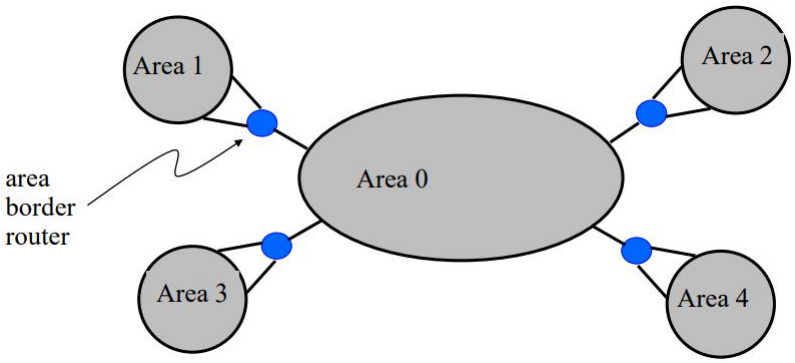
- How to Flood the link state?
  - every node sends link-state information through adjacent links
  - next nodes forward that info to all links except the one where the information arrived



- When to initiate flooding?
  - Topology change
    - link or node failure/recovery
    - link cost change
  - Periodically
    - refresh link-state information
    - typically 30 minutes

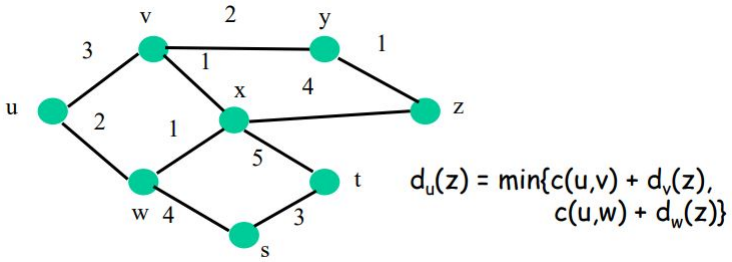
8.1.10 Scaling Link-State Routing

- Overhead of link-state routing
  - flooding link-state packets throughout the network
  - running Dijkstra's shortest-path algorithm
- Introducing hierarchy through "areas"



8.1.11 Bellman-Ford Algorithm

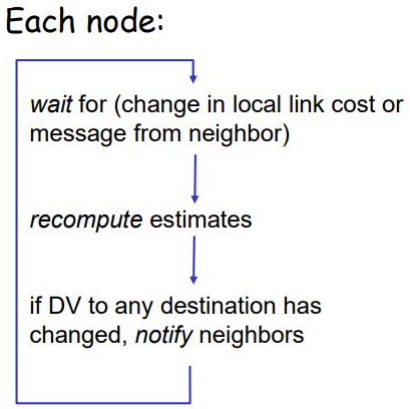
- Define distances at each node x
  - $d_x(y)$ = cost of least-cost path from x to y
- Update distances based on neighbors
  - $d_x(y)$ = min $\{c(x,v) + d_v(y)\}$ over all neighbors v



8.1.12 Distance Vector Algorithm

- c(x,v) = cost for direct link from x to v
  - node x maintains costs of direct links **c(x,v)**
- $D_x(y)$ = estimate of least cost from x to y
  - node x maintains distance vector $\mathbf{D}_x = [\mathbf{D}_x(\mathbf{y}): \mathbf{y} \in \mathbf{N}]$
- Node x maintains also its neighbors' distance vectors
  - for each neighbor v, x maintains $\mathbf{D}_v = [\mathbf{D}_v(\mathbf{y}): \mathbf{y} \in \mathbf{N}]$
- Each node v periodically sends D_v to its neighbors
  - and neighbors update their own distance vectors
  - $D_x(y) \leftarrow \min_v\{c(x,v) + D_v(y)\}$ for each node $y \in N$

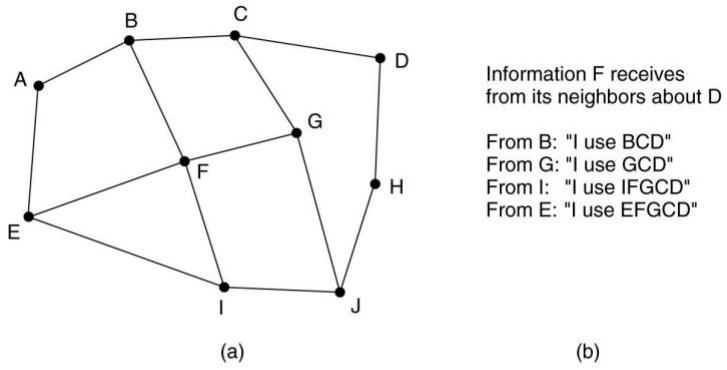
- Over time, the distance vector D_x converges
- Iterative, asynchronous, each local iteration caused by:
  - local link cost change
  - distance vector update message from neighbor
- Distributed
  - node notifies neighbors only when its DV changes
- Neighbors then notify their neighbors, if necessary



8.1.13 Routing Information Protocol (RIP)

- Distance vector protocol
  - nodes send distance vectors every 30 seconds
  - or when an update causes a change in routing
- RIP is limited to small networks

8.1.14 BGP – The Exterior Gateway Routing Protocol

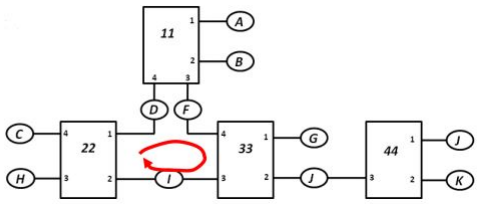


(a) A set of BGP routers. (b) Information sent to F

8.2 Unique Spanning Tree in Ethernet Networks

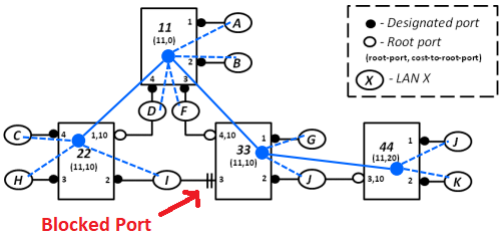
8.2.1 L2 Networking - Single Tree Required

- Ethernet frame
  - No hop-count
  - Could loop forever
  - broadcast frame, mis-configuration



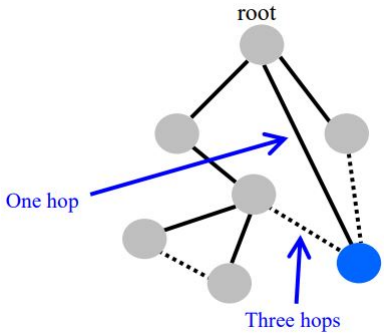
- Layer 2 network
  - **Required to have tree topology**
  - Single path between every pair of stations
- Spanning Tree Protocol (STP)
  - Running in bridges
  - Helps building the spanning tree

– Blocks ports



8.2.2 Constructing a Spanning Tree

- Distributed algorithm
  - switches need to elect a "root"
    - the switch with the smallest identifier
  - each switch identifies if its interface is on **the shortest path from the root**
  - messages(Y,d,X)
    - from node X
    - claiming Y is the root
    - and the distance is d



8.2.3 Steps in Spanning Tree Algorithm

- Initially, each switch thinks it is the root
  - switch sends a message out every interface
  - identifying itself as the root with distance 0
- Other switches update their view of the root
  - upon receiving a message, check the root id

- if the new id is smaller, start viewing that switch as root

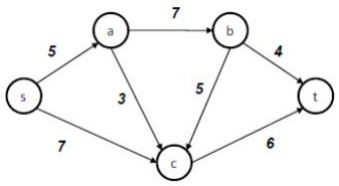
- Switches compute their distance from the root
  - add 1 to the distance received from a neighbor
  - identify interfaces not on a shortest path to the root and exclude them from the spanning tree

8.3 Maximum Flow of a Network

8.3.1 Flow Network Model

- **Flow network**
  - source s
  - sink t
  - nodes a, b and c

- Edges are labeled with **capacities (ex: bit/s)**



- Communication networks are not flow networks
  - they are queue networks
  - flow networks enable to determine limit values

8.3.2 Maximum Capacity of a Flow Network

- Max-flow min-cut theorem
  - maximum amount of flow transferable through a network
  - equals minimum value among all simple cuts of the network

- Cut → split of the nodes V into two disjoint sets S and T
  - $S \cup T = V$
  - there are $2^{|V|-2}$ possible cuts

- Capacity of cut (S,T):

$$c(S,T) = \sum_{(u,v)|u \in S, v \in T, (u,v) \in E} c(u,v)$$

  - (sum of the cost of all edges from S to T)