Fundamentals of Communication Networks

Prof. Alessandro Redondi

February, 03 2017

1. **Exercise (6 pts)**:
Host A transmits bursts of 10 packets to host B. Let X be the random variable that counts the number of lost packets in each burst. The Cumulative Density Function (CDF) of X is reported hereafter:

$$F_X(x) = \begin{cases} 5.9 \times 10^{-6},\ x < 1 \\ 1.4 \times 10^{-4},\ 1 \leq x < 2 \\ 0.0016,\ 2 \leq x < 3 \\ 0.0106,\ 3 \leq x < 4 \\ 0.0473,\ 4 \leq x < 5 \\ 0.1503,\ 5 \leq x < 6 \\ 0.3504,\ 6 \leq x < 7 \\ 0.6172,\ 7 \leq x < 8 \\ 0.8507,\ 8 \leq x < 9 \\ 0.9718,\ 9 \leq x < 10 \\ 1.0000,\ x \geq 10 \end{cases} \quad (1)$$

(a) Compute the Probability Mass Function of X.
(b) Compute the probability that exactly one packet is lost in a burst.
(c) Compute the probability that more than 5 packets are lost.
(d) Show how to compute the Packet Error Rate for the link between A and B.

Solution

(a) To compute the PMF of X, it is enough to compute the difference between consecutive entries in the CDF:

$$f_X(x) = \begin{cases} 5.9 \times 10^{-6},\ x = 0 \\ 1.4 \times 10^{-4} - 5.9 \times 10^{-6} = 1.3 \times 10^{-4},\ x = 1 \\ 0.0016 - 1.4 \times 10^{4} = 0.0014,\ x = 2 \\ 0.0106 - 0.0016 = 0.009,\ x = 3 \\ 0.0473 - 0.0106 = 0.0367,\ 4 \leq x < 5 \\ 0.1503 - 0.0473 = 0.103,\ 5 \leq x < 6 \\ 0.3504 - 0.1503 = 0.2001,\ 6 \leq x < 7 \\ 0.6172 - 0.3504 = 0.2668,\ 7 \leq x < 8 \\ 0.8507 - 0.6172 = 0.2335,\ 8 \leq x < 9 \\ 0.9718 - 0.8507 = 0.1211,\ 9 \leq x < 10 \\ 1.0000 - 0.9718 = 0.0282,\ x = 10 \end{cases} \quad (2)$$

(b) The probability that exactly one packet is lost is easily readable from the PMF, and it is $P(X = 1) = 1.3 \times 10^{-4}$

(c) The probability that more 5 packets are lost can be computed either looking at the CDF: we have $P(X > 5) = 1 - P(X \leq 5) = 1 - 0.1503 = 0.8497$. Alternatively, we can sum the entries of the PMF for $x > 5$, that is $P(X > 5) = 0.2001 + 0.2668 + 0.2335 + 0.1211 + 0.0282 = 0.8497$

(d) Let p be the packet error rate. We can write:

$$p^{10} = P(X = 0) \tag{3}$$

therefore we have $p = \sqrt[10]{5.9 \times 10^{-6}} = 0.3$

2. **Exercise (8 pts)**

A small enterprise purchases the IP address 131.175.128.0/19. Define an addressing plan to serve the following requirements:

- 1 subnet, 1000 host
- 2 subnets, 500 hosts each
- 3 subnets, 230 hosts
- 3 subnets, 100 hosts each
- 1 point to point link

Solution 1000 hosts need 10 bits. Therefore we can use 3 more bits for the network part and create 8 subnets:
sn1: 131.175.128.0/22 - netmask: 255.255.252.0 - broadcast: 131.175.131.255
sn2: 131.175.132.0/22 - to be divided
sn3: 131.175.136.0/22 - to be divided
sn4: 131.175.140.0/22 - to be divided
sn5: 131.175.144.0/22 - to be divided
sn6: 131.175.148.0/22 - to be divided
sn7: 131.175.152.0/22 - to be divided
sn8: 131.175.156.0/22 - to be divided

500 hosts need 9 bits. We can take sn2 and use 1 additional bit for the network part:
sn9: 131.175.132.0/23 - netmask: 255.255.254.0 - broadcast: 131.175.133.255
sn10: 131.175.134.0/23 - netmask: 255.255.254.0 - broadcast: 131.175.135.255

230 hosts need 8 bits. We can take sn3 and use 2 additional bits for the network part:
sn11: 175.203.136.0/24 - netmask: 255.255.255.0 - broadcast: 175.203.136.255
sn12: 175.203.137.0/24 - netmask: 255.255.255.0 - broadcast: 175.203.137.255
sn13: 175.203.138.0/24 - netmask: 255.255.255.0 - broadcast: 175.203.138.255
sn14: 175.203.139.0/24 - to be divided

For the 100-hosts subnets we 7 bits for the host part. We cannot use sn14 for accomodating all networks (only 1 bit would be available, that is only 2 subnets of 128 hosts can be accomodated there). It is better to take sn4 and use 3 bits to form 8 subnets:
sn15: 131.175.140.0/25 - netmask: 255.255.255.224 - broadcast: 131.175.140.127
sn16: 131.175.140.128/25 - netmask: 255.255.255.224 - broadcast: 131.175.140.255
sn17: 131.175.141.0/25 - netmask: 255.255.255.224 - broadcast: 131.175.141.127
sn18: 131.175.141.128/25 - - to be divided
sn19: 131.175.142.0/25 - - to be divided
sn20: 131.175.142.128/25 - to be divided
sn21: 131.175.143.0/25 - to be divided
sn22: 131.175.143.128/25 - to be divided

Finally, for the point to point link we can take e.g. sn18 and use 2 bits for the host part and 5 bits for the network part (32 p2p links can be created)
sn23: 131.175.141.128/30 - netmask: 255.255.255.252 - broadcast: 131.175.141.131

3. **Exercise (6 pts)**

Host A establishes a TCP connection with host B. The capacity and propagation delay of link connecting A with B are: $C = 500$ kbps, $\tau = 5$ms.

Assuming:

- MSS = 200 byte
- SSTHRESH = 1 kB
- RCWND = 2 kB

- ACK and control packets length = 20 byte
- headers length negligible
- bidirectional links

Compute the time needed to transfer a 10 kB file from A to B (from the connection setup to the reception of last ACK at A).

Solution

Let T be the transmission time of one MSS and T_{ack} the transmission time of one ACK. We have $T = (200 \times 8)/(0.5 \times 10^{-6}) = 3.2$ ms and $T_{ack} = (20 \times 8)/(0.5 \times 10^{-6}) = 0.32$ ms.

The RTT between A and B is therefore $RTT = T + \tau + T_{ack} + \tau = 13.52$ ms.

The connection setup time is $T_{setup} = 2 \times T_{ack} + 2\tau = 10.64$ ms

The dimension of the window after which transmission becomes continuous is: $W > RTT/T = 5$ MSS

According to TCP operation, in the first RTT one MSS is transmitted. In the second RTT two MSS are transmitted. In the third RTT four MSS are transmitted. From the fourth RTT, transmission becomes continuous.

Therefore in the first 3 RTT, 7 MSS are transmitted and we can compute the time needed for the transmission of the entire file (50 MSS) as:

$T_{tot} = T_{setup} + 3RTT + 42T + RTT = 199.12$ ms

4. **Exercise (6 pts)**

Two hosts A and B are connected through a switch S. A is connected to Port 1 and B to Port 2. Let MAC-A and IP-A be the layer 2 and 3 addresses of A and MAC-B, IP-B the addresses of B. Host A transmits an IP packet to B and B replies to A with another IP packet.

(a) Assuming that the forwarding tables of S is empty and the ARP tables of A and B are empty, indicate which packets are generated and transmitted in order to complete the exchange of packets. For each packet indicate the addresses included in the layer 2 (MAC) and layer 3 (IP or ARP) headers.

(b) Indicate how the forwarding database of the switch changes after each packet.

Solution

1. A knows the IP of B but not its MAC (ARP tables are empty). First of all, an ARP request is issued with the following headers: Layer 2 [SRC: MAC-A, DST: BROADCAST], Layer ARP [IP-A, MAC-A, IP-B, ?]
2. The ARP request is received by the switch, which adds the entry [MAC-A, PORT 1] to its FDB. The ARP request is forwarded to all ports (its a broadcast packet) by the switch and reaches B. B recognizes its IP address in the ARP request and issues an ARP reply: Layer 2 [SRC: MAC-B, DST: MAC-A], Layer ARP [IP-A, MAC-A, IP-B, MAC-B]. B also stores in its ARP table the IP and MAC address of A.
3. The ARP reply is received by the switch which adds the entry [MAC-B, PORT 2] to its FDB. The ARP reply is forwarded by the switch to A. Now A knows the MAC address of B and can transmit the IP packet: Layer 2 [SRC MAC-A, DST: MAC-B] Layer IP [SRC IP-A, DST IP-B]
4. This time the packet is directly forwarded to B, since the FDB table contains its MAC address.
5. B receives the IP packet and replies with a new packet Layer 2 [SRC MAC-B, DST: MAC-A] Layer IP [SRC IP-B, DST IP-A]. The packet is then forwarded to A from the switch.

5. **Question (4 pts)**

Given the network of switches indicated in the figure (link costs are indicated on each link), run the Spanning Tree Protocol and identify the root switch and the root ports (R), designated ports (D) and blocked ports (B) for all other switches. What is the main purpose of the spanning tree protocol?

Solution

6. **Questions (4 pts - each answer can be either TRUE or FALSE)**

In case the answer is FALSE, briefly explain why.

- [T] [F] The throughput achievable with slotted ALOHA is twice the one achievable with pure ALOHA
  TRUE.
- [T] [F] Bellman-Ford shortest path algorithm is generally more efficient than Dijkstra's algorithm.
  FALSE. In the worst case, Bellman-Ford has a complexity of $O(N^3)$ while Dijkstra has complexity $O(N^2)$
- [T] [F] The RCVWND field in TCP is used to avoid that a link becomes congested.
  FALSE. It is used to perform flow control and avoid that the receiver's buffer is overloaded.
- [T] [F] NAT (Network Address Translation) devices translates hostnames into IP addresses.
  FALSE. They translate private IP addresses into public IP addresses.

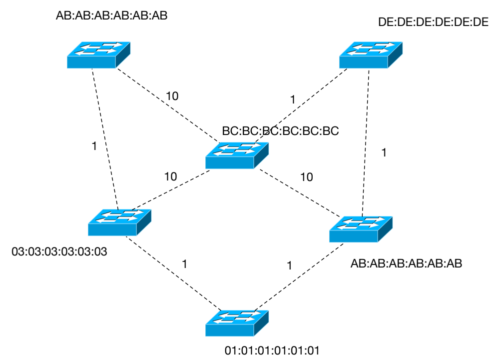

Figure 1: Network topology

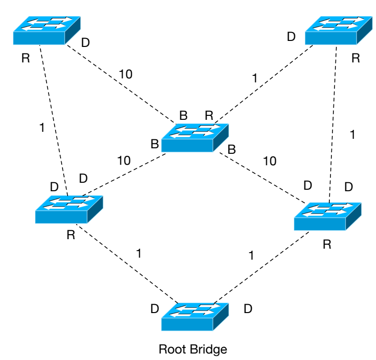

Figure 2: Network topology