CS538 Take-home Submission

Siting Chang, schang13

October 6, 2014

1 Problem 1: Data Transmission in Internet

1.1 part 1

Advantages of Packet Switching v.s. Circuit Switching Transmission are:

- Packet switching is able to achive higher utility in terms of bandwidth. The key feature of circuit switching is that it reserves a connection/channel between a pair of hosts for a certain period of time, with and without actual data or messgaes being transmitted. In other words, there are times the connection sends nothing while keeping other hosts connect to each other.

  However, packet switching allows packets between various hosts to use the same connection at the same time. It increases the utility by allowing messages sent between various hosts to take advantage of the dull time when the connection is not being used by others.

- Packet switching is able to resend only damaged or lost packets while circuit switching has to resend the entire message. Packets in packet switching have sequence numbers which help to identify which packet got damaged or lost, so the specific packet could be resent. However, circuit switching does not keep such sequence numbers. So when a piece of message gets lost, the entire data or file needs to be resent.

Disadvantages of packet switching are:

- Packet switching cannot guarantee a bandwidth while the circuit switching can. The delivery under packet switching is best-effort delivery.

- The forwarding in circuit switching would be much simpler compared to packet switching. Because a channel is reserved in circuit switching, routers visited by the transmitted messages are forwarding messages to the same destination. Therefore, as long as a channel is alive, the routers can forward the messages to the same outgoing link.

  However, in packet switching, packets are being forwared to multiple destinations which requires the routers to decide which outgoing inks to send these packets. So more effort is cost when forwarding packets in packet switching.

1.2   part 2

1.2.1   part a

NCP was not sufficient because it had no end-to-end host error control. To be more clear, NCP was designed only to serve ARPANET – the only existing network which was so reliable that no error control was neeeded, so NCP had no end-to-end host error control. However, when network enlarged, network reliability became one critical issue which NCP could not resolve. Therefore, NCP was not sufficient anymore.

One missing feature of NCP to address the scale of Internet growth is error detection. When Internet dramatically grows, the reliability of Internet decreases which introduces corrupted packets. NCP does not provide the serivice to identify these corrupted packets. Another missing feature of NCP is congestion control. Since growing number of computers are trying to send data packets to the network, it is easy to forecast that there will be a high potential of traffic congestion. Therefore, having some protocol to restrict the amount of data being sent into the network is critical, which NCP did not have.

1.2.2   part b

TCP/IP is a replacement protocol for NCP. The new features are:

1. TCP/IP provides reliable transmission by including a sequence number in its header. The sequence number is used to identify the order of received data packets and reconstruct the original message/file regardless of any disordering or packet loss.

2. TCP/IP provides error detection since it has a checksum field in its header. After receiving a data packet, a checksum procedure will be performed by the receiver to ensure the correctness of the received data.

3. TCP/IP provides sliding window flow control protocol to avoid letting senders sending data to fast that receivers cannot properly receive and process the received packets. The flow control protocol lets receivers specify restricts the maximum amount of data a sender could send, and the sender could only move forward to sending more data after it received an acknowledgement from the receiver.

4. TCP/IP provides congestion control by adjusting the speed senders send data into the network. Generally speaking, if a potential traffic congestion is detected, senders decrease their sending rates to less the traffic goes into the network. If no sign of traffic congestion is detected, senders are allowed to increase their sending rates to take fully advantage of the network.

1.3   part 3

1. Domain Name System (DNS) was introduced to address the naming and addressing issue with the development of scale of Internet. The early stage network has a limited

number of hosts and their names and addresses are able to be stored in a single table. However, with the increasing number of hosts in network, the single table is not suitable anymore. Therefore, the hierachical distributed naming system, DNS, was developed to solve this issue by associating various information with domain names assigned to each of entities and it translates easily domain names to the numerical IP addresses to locate computers.

2. A hierarchical model of routing using an Interior Gateway Protocol (IGP) and an Exterior Gateway Protocol (EGP) was developed to connect the networks and regions together. Each region can run their selected IGP to deliver packets inside their region, while use EGP to route packets among different regions.

2 Problem 2: Secure and Reliable Data Transmission Service

2.1 part 1: possible attacks and faults

3 Problem 3: High Speed Routers

3.1 part 1

1. Each forwarding engine has a complete set of the routing tables. Traditionally, routers kept a central master routing table and the satellite processors each keep only several latest used routes. This leads to the problem that if a route information is not available in the satellite processors, then requests need to be made to obtain the information from the central master routing table. Therefore, at high speeds, the cost of requesting routing table multiple times is much higher than processing the packet header.

   By letting each forwarding engine has a complete set of routing tables would overcome the bottelneck issue.

2. The MGR uses a switched backplane. Switched backplane allows parallelism of a switch compared to the traditionally applied shared bus mechanism.

3. The MGR includes quality of service (QoS) processing in the router by spiltting the QoS function. The forwarding engine classifies packets and a specialized processor called QoS processor takes charge of the scheduling of the backets. This design proves the possibility of building a router that includes line-speed QoS.

3.2   part 2

3.2.1   part a

The Ethernet-used ARP does not work for the MGR architecture is because the pipelined MGR does not have a convenient place in the forwardingn engine to store datagrms awaiting an ARP reply.

3.2.2   part b

The ARP is implemented following a two-part strategy. The first part is the router ARP's for all apossible addresses on each interface to collect link-layer addresses for the forwarding tables at a low frequency. And the second part is datagrams for which the destination link-layer address is unknown are passed to the network processor, which does the ARP and, once it gets the ARP reply, forwards the datagram and incorporates the link-layer address into future forwarding tables.

3.3   part 3

3.3.1   part a

The reason IP header checksum is not checked is due to its high cost. In the best situation, it would require 17 instructions to be spread over a minimum of 14 cycles which increase the time to perform the forwarding code about 21%. It is considered as high cost to check for a rare error that can be caught end-to-end.

3.3.2   part b

1. If the destination in a header of a data packet is missed in the route cache,

2. Since the forwarding engine is designed to instruct the inbound line card to discard the errored packets, therefore packets whose headers have errors will be dicarded and appear as lost.

3. The forwarding engine does not handle packets whose headers have IP options.

3.4   part 4

The advantage of switched architecture is it does not have the problem of head-of-line blocking since each input keeps its own FIFO and bids separately for each output. And it was shown that such a switch can achieve 100% throughput.

   The disadvantage of switched architecture is that it is a point-to-point switch without the function of one-to-many, so it does not support multicasting.

3.5 part 5

One option is as described in the "A 50-Gb/s IP Router" paper. Split the forwarding table memory on forwarding engines into two parts, call them A and B. When getting the latest forwarding table from network processor, only one or the two memories is used. Let's assume the updated information is feeding in to part A. At the same time, part B is still being used by forwarding engines to forward packets. As soon as the complete forwarding information finished updating in part A, forwarding engines start to use the forwarding information from A. Therefore, in the described process, there is no time that is spent on updating forwarding table solely. In other words, while updating the forwarding table, packets are still being sent by forwarding engines which makes the updating process seems as no designated time is required.

Another option is in the situation when bandwidth is a high limitation which means updating forwarding table for all forwarding engines would cause congestion in the network. Furthermore, it would slow down the updating process for all forwarding engines. Therefore, we could devide the network into different zones and let each zone has a master forwarding engine. When forwarding information needed to be pushed from network processor to forwarding engines, the forwarding engines will feed the forwarding table to those master forwarding engines. After the table in master forwarding engines got updated, they will push the table to other forwarding engines in their zones respectively.

4 Problem 4

4.1 Source routing algorithm

To simplify the notations when describing paths, labels are assigned to paths as shown in Figure 4.1.

Step 1: generate all paths from s to d that satisfy the bandwidth requirement.

Noticed that there is only one link, Link 8, in the graph that has a bandwidth under 100 Mbs. Therefore, any paths including Link 8 will not satisfy the bandwidth requirement. When generating candidate paths, we exclude those paths which include Link 8. And all other generated paths satisfy the bandwidth requirement.

The paths generated are:

1. s-1-3-5-10-d

2. s-1-3-6-d

3. s-1-3-7-9-d

4. s-1-4-5-6-d
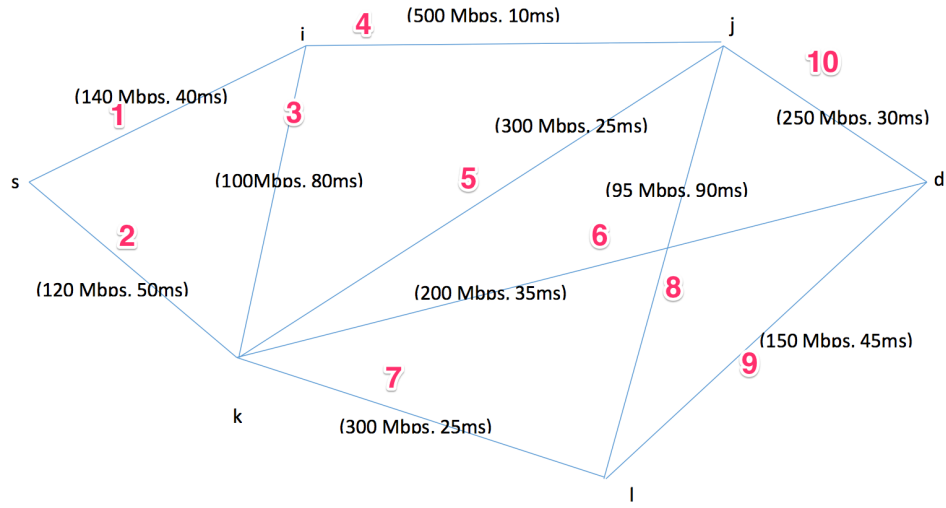
5. s-1-4-5-7-9-d

6. s-1-4-10-d

Figure 4.1: Graph

7. s-2-3-4-10-d

8. s-2-5-10-d

9. s-2-6-d

10. s-2-7-9-d

Step 2: keep paths that satisfy the latency requirement.

What we do is, for each path listed above, sum up the total latency and exclude those with a total latency higher than 100 ms.

The remained paths are:

1. s-1-4-10-d

2. s-2-6-d

In conclusion, the best paths that satisfy both requirements are:

1. s-(s,i)-(i,j)-(j,d)-d

2. s-(s,k)-(k,d)-d

4.2   Hop-by-Hop routing Algorithm

The link preference is to choose link with the highest bandwidth. The steps are listed as follows:

1. Start from origin s.

2. Choose Link 1 over Link 2.

3. Now we reaches node i. Choose Link 4 over Link 3.

4. Reaches node j. Choose Link 5 over Link 8 and Link 10.

5. Reaches node k. Choose Link 7 over Link 6.

6. Reaches node l. Choose Link 9 over Link 8.

7. Reaches destination d.

However, the total latency of the above path 145ms which is greater than the restriction. This means that following this preference, the best path with all parameters at their optimal values does not exist.

Changing the link preference to choosing link with the lowest latency, this gives the same path as described above. And again, we reach the conclusion that the best path with all parameters at their optimal values does not exist.