ECE 333: Introduction to Communication Networks
Fall 2000

Lecture 21: Switching & Multiplexing III

Packet switching and statistical multiplexing

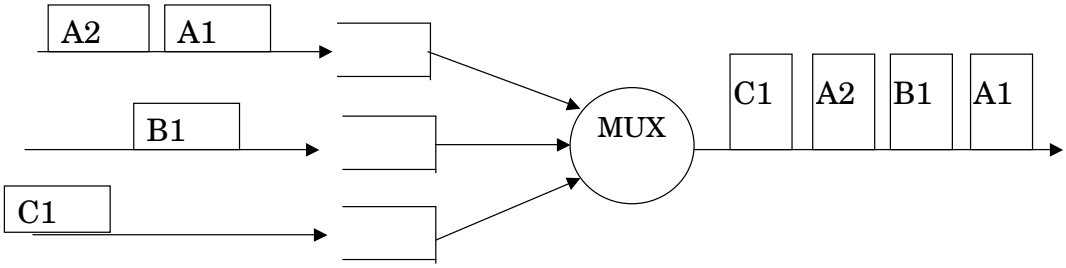
The last lecture addressed circuit switched networks and static multiplexing approaches such as TDM and FDM. The main example of a circuit switched networks that we considered was the public telephone network. This lecture will cover packet switched networks and statistical multiplexing. These techniques are used in WAN's. The Internet is one example of a packet switched network. A number of other packet switched network architectures have been standardized by the telephone companies and used to provide leased services to customers. These include

- **X.25** - a standard based on layer 3 of the OSI reference model.
- **Switched Multimegabit Data Service (SMDS)**
- **Frame relay**
- **Asynchronous Transfer Mode (ATM)**

In the following we will take a closer look at both statistical multiplexing and packet switching.

Statistical (time-division) multiplexing (SM)

With statistical multiplexing, packets are multiplexed together based on demand. One approach to this is to have all packets placed in a common queue and served FCFS. As we have seen (see lecture 13 and Problem set 4) with burst traffic such an approach results in smaller delay and smaller buffer sizes than with a static multiplexing approach.



Statistical multiplexing is usually implemented at the packet level. Packets may have variable sizes or may all be a fixed size depending on the network. For example the Internet allows for variable packet sizes, while in ATM networks all packets have a fixed size of 53 bytes. The outgoing link may be synchronous and require each packet to be in a fixed slot, or it may be asynchronous allowing a packet to be sent at any time.

Statistical multiplexing

Statistical multiplexing is not without it cost. One cost is that is much more difficult (than with static multiplexing) to guarantee service characteristics such as delay to users. Note with a static multiplexing approach the maximum delay the first packet from a user will have to wait in a multiplexer can be determined. Also for fixed rate users, with static multiplexers it can be guaranteed that no packets will be dropped, again this is more difficult with statistical multiplexing. Finally, with statistical multiplexing, users are not as isolated from each other as with static multiplexing.

Some of these issues can be addressed by implementing various service disciplines other than FCFS, such as priority service, round robin, etc. (see Lecture 11). Much of the current research in networking seeks to address how to provide users with an acceptable Quality of Service while still taking advantage of the benefits of statistical multiplexing. The service discipline used at the multiplexers is but one technique used to address this problem.

Packet Switching

In packet switched networks, a packet received at a switch is stored into memory and then forward on the correct output link when it is available. Because of this packet switching is also called ***store-and-forward*** switching. There are some variations of the above description, for example with ***cut-through switching*** a packet may be forwarded to the output port after only the header has been placed into memory. In circuit switched networks, the forwarding function of the switch is established during call set-up and based on the subchannel information arrives in. In packet switched networks the forwarding decision must be based on information in the packet header. This is implemented in two different ways, called ***datagram switching*** and ***virtual circuit switching***.

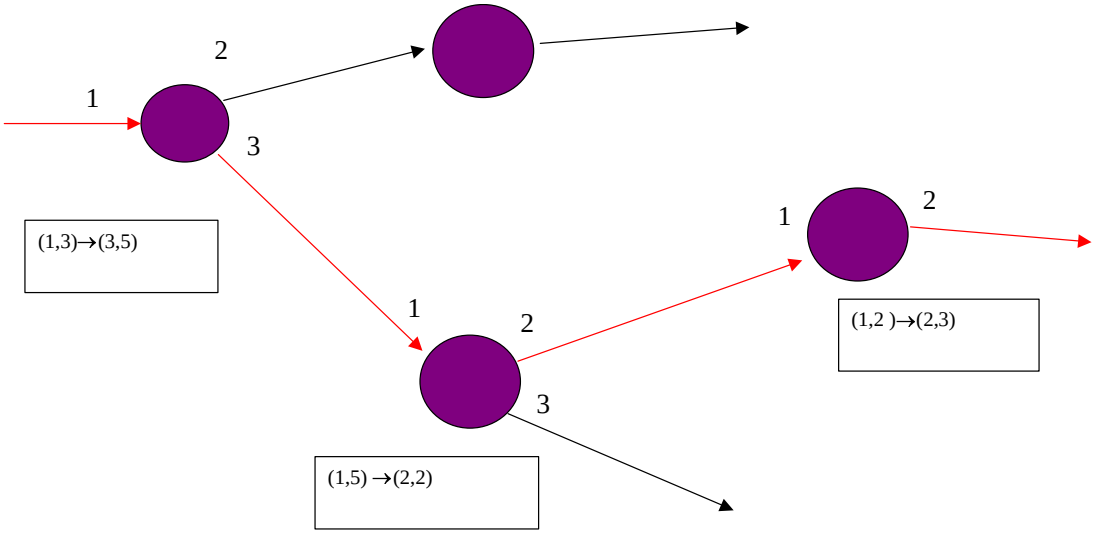
Datagram switching- Datagram switching is a connectionless approach, requiring no call set-up. Each packet or datagram contains the destination address. The packet is forwarded according to a routing table stored in the router. This table may be static or periodically change. In datagram networks, the forwarding decision for each datagram is made independently; different packets for the same destination may take different routes and arrive out-of-order. Datagram switching is used in IP and SMDS networks.

Virtual Circuit - Virtual circuit switching is a connection-oriented approach. As in a circuit switched networks, a call set-up is required to select a route before data is transmitted, however with virtual circuit networks, no fixed rate circuit is reserved. Data is routed along the fixed route using ***virtual circuit identifier*** (***VCI***); VCI's are usually much shorter than destination addresses. The switch may change the VCI in a packet at each hop. At call set-up, each node establishes a forwarding table entry that contains the incoming port and VCI for a connection and the corresponding outgoing port and VCI. Each session on a given link needs a unique VCI, but the same VCI may be re-used elsewhere in the network.

In some virtual circuit networks, permanent virtual circuits can be established between two locations. In this case no call set-up is required for a connection between these locations. Since virtual circuit networks require a call set-up, they can also block certain requests, for example to ensure that the delay of existing users is not too large.

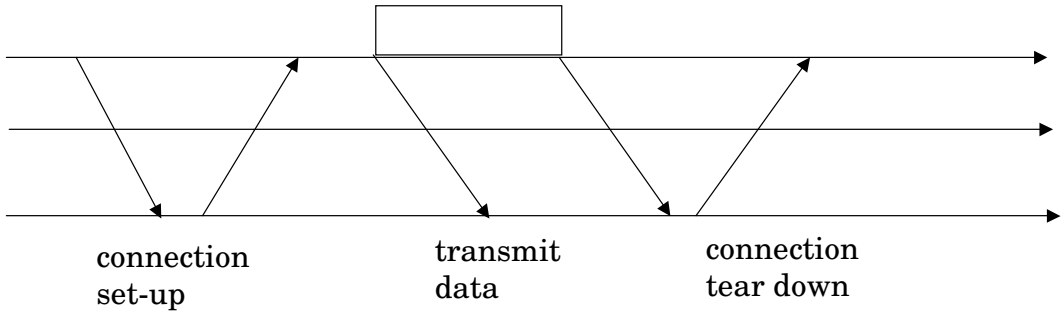
Virtual circuits are used in X.25, Frame Relay and ATM networks. A virtual circuit approach, called Multi-Protocol Label Switching (MPLS) is also being introduced in the Internet.

Example of virtual circuit switching:

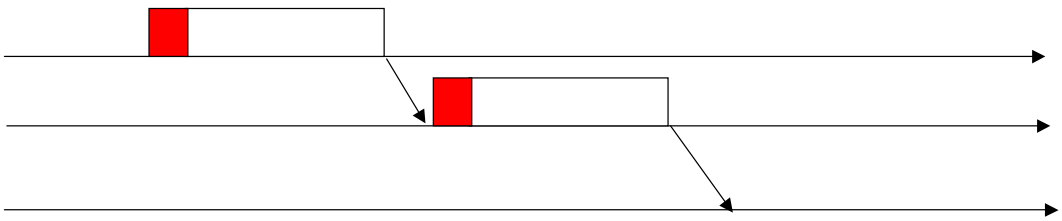


Each box shows the forwarding table entry for a connection along the path shown in red. The notation used is (incoming port, VCI)→(outgoing port, VCI). Thus at the first switch, packets arriving with VCI 1, will be forwarded to port 3 and their VCI will be set to 5.

Comparison of circuit switching & datagram switching:



Circuit switched path



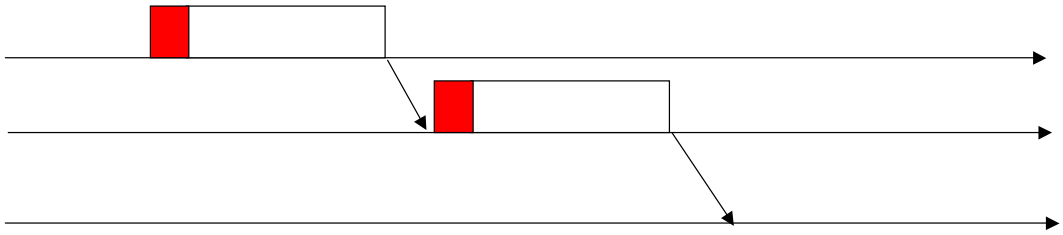
Transmit data &
header

Datagram switched path

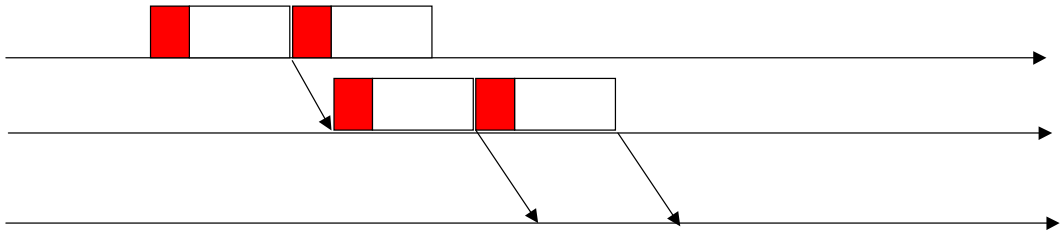
The timing diagrams above show a single packet sent on a circuit switched network and a datagram switched network. Here we assume that in both cases the entire transmission rate is used for the packet. In the circuit switched case, if FDM or TDM is used then the transmission time would be longer. For the circuit switched approach, extra delay is incurred for call set-up. While for the packet switched approach, extra delay is incurred for forwarding at intermediate nodes.

The delay incurred at intermediate nodes in store and forward switching can be reduced through *pipelining:*

Large packets:

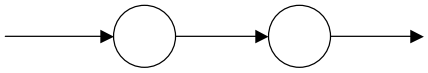


Shorter packets:



Trade-offs in packet size for pipelining:

Suppose you want to send an ***M*** bit message over ***L*** switches from a source to destination in a store-and-forward network.



Ex. with L = 2.

Assume the message is sent as ***K*** packet*s* and each packet requires an ***H*** bit header. Also, assume that each link has a transmission rate of ***C*** bps. Thus, each packet will contain ***M***/***K*** bits of the message plus ***H*** header bits. (Assume ***M***/***K*** is an integer, if not then the last packet will contain fewer bits than the rest.) In this case the transmitted message will contain $M + KH$ bits, this will take $\frac{M+KH}{C}$ seconds for the source to transmit.

After transmission (ignoring propagation delays) it will take an additional $\frac{M/K+H}{C}$ seconds of transmission delay at each of the L switches. Thus the total delay is given by:

$$\frac{M+KH}{C} + L\left(\frac{M/K+H}{C}\right)$$

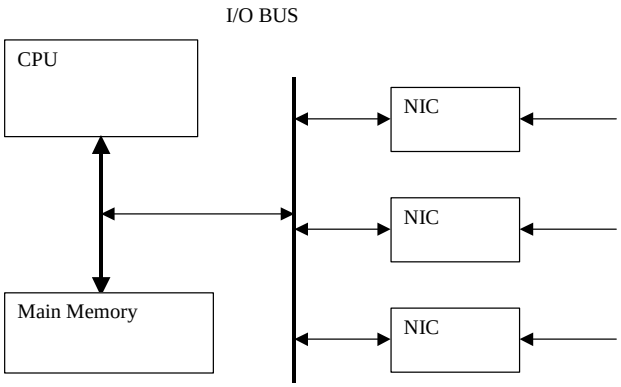
In this case, it can be shown that $K_{opt} \approx \sqrt{LM/H}$.

Comparisons of circuit switching and packet switching

- For bursty traffic, datagram switching usually achieves higher utilization than circuit switching.
- For short sessions, datagram switching can achieve less delay.
- With datagrams, end-to-end delay is variable, no guarantees. Also requires buffering and results in possible losses within network.
- With datagram switching, re-routing is easier than circuit switching, for example due to a failure.
- Datagram switching usually provides connectionless service, circuit switching is connection oriented.
- Virtual circuit switching is a compromise. Less efficient than datagram switching, but better efficiency than circuit switching. Can provide some guarantees if admission control used.

Examples of packet switches

A basic design for a packet switch (router) is simply a workstation with mulitple interfaces, as shown below. (here NIC= Network Interface Card)



In this switch, when a packet arives, it gets transferred into memory. The CPU executes a routing table lookup for output port. The packet then gets transfered to the output NIC and forwarded. In this buffering is needed in the interfaces for each input, in case the I/O bus is busy. This design can be limited by the I/O bus speed, the memory bandwidth, or the CPU speed.

The maximum throughput of the switch is the maximum number of packets/sec that can be transferred through the switch. Suppose the I/O bus BW is limited to P packets/sec. Each packet must go over this bus at least twice - once going from the NIC to memory and once the other way. Thus the maximum throughput of the switch is limited to $P/2$ packets/sec. Suppose the switch has N input and output ports and each port is connected to a line with a transmission rate of R packets per second. Then for the bus to not be a bottleneck at high loads, we would need $P > 2NR$. Similar arguments can be made for the memory and CPU. In many cases, the speed of a switch is limited by the processing per packet, not per bit.

Several things can be done to improve rates in packet switches including to do more processing in lines cards, to use techniques for faster table look-ups and to replace the bus with a ***switching fabric***; the switching fabirc may be a cross-bar or space-division switch as discussed in the previous lecture. An example of this is shown below.

High-speed Packet Switch