Chapter IV: Special Topologies

1. Loop Networks

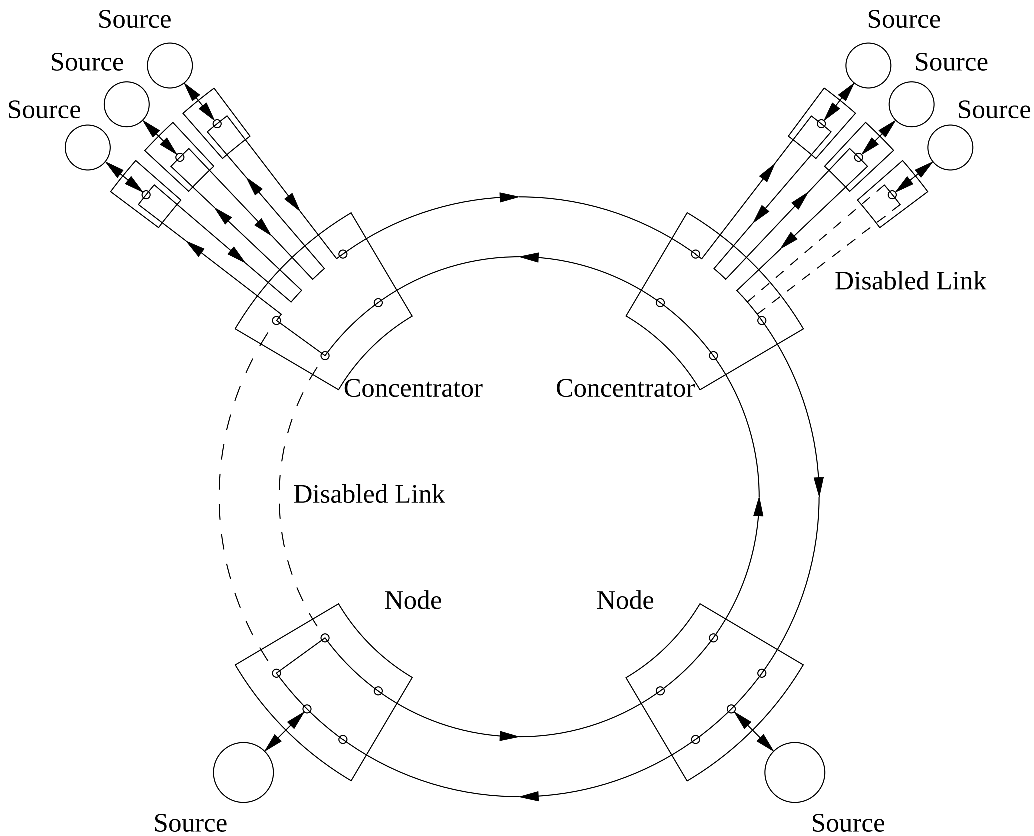
1.1 Introduction

Reference 1 Section 4.5.3

1. Describe Topology
   Daisy chain of nodes
2. Transmission
   i. Point-to-point transmission
      — Fixed signal levels at the receiver, no matter which node is the original source
      — Regenerate the the signal at each intermediate node
      — Point-to-point links
        fiber-optic devices are often used.
   ii. Continous Transmission on each link - independent of the access protocol
       Even when multiple sources contend for the channel
      — In is not necessary to obtain bit synchronization at the beginning of each transmission
      — Detect failures of nodes or breaks in a link by the absence of the continuous transmission.
      — Active components in the path increase the probability of failure

3. Bidirectional Loop

Preferred solution in most Metropolitan area networks
SONET Rings, FDDI



Operation:

When a terminal detects a loss of signal on an incoming link, it signals the source of the signal, instead of forwarding the signal.

The terminal at the input to the failed link receives the signal and transmits the received signal on the standby link.

The terminal that detected the failure forwards the signal that it receives on the standby link onto the normal link.

1.2 Access Protocols

1.2.1 Token passing -- IEEE 802.5

1. The objective is to guarantee that only one ring interface transmits at a time

   The objective is met by transmitting a single token on the ring.

   An interface that has data to transmit removes the token from the ring, transmits its data, then re-inserts the token on the ring when it is done.

2. Ring Delay

   Each ring interface delays transmission by $m \geq 1$ bits

   The delay is necessary to receive and regenerate the bits and to examine the bits to decide what action to take.

   In a network with v interfaces, the delay around the ring is $mv + \Delta_p$, where Δ_p is the propagation delay around the ring

   The packet length is variable and may either be less than be greater than ring delay

3. The token

   01111110 is a free token, and is an invitation to transmit data, and

   01111111 is a busy token, and indicates that an earlier ring interface has taken the token and is transmitting data.

4. Data Transmission

   When a ring interface has data to transmit and receives 0 followed by 6 1's, it examines the next bit.

   If the bit is 0, it converts it to 1 and transmits its data

   The interface transmits the free token at the end of its transmission to invite the next ring interface to transmit data.

   After the free token, the ring interface transmits an idle fill sequence of bits.

   The idle fill is usually a simple sequence, for instance alternating 0,1.

5. Preventing a token from appearing in the data

   If the pattern 0111111X appears in the data, it may be misinterpretted as a token.

   Whenever the 5 consecutive 1's appears in the data , the source inserts a 0

   The receiver removes one zero following 5 1's, so that a source can transmit any data pattern. data.

6. Removing Data

   The data travels around the entire ring, past one or more receivers, and is removed by the source.

   The source starts removing data from the ring when it starts to transmit, and continues removing data until it receives the token that it inserts at the end of its transmission.

   Once a source acquires a token, the next token that it receives is the token that it acquired.

      This token may arrive before or after it finishes transmitting its data, depending on the delay around the ring and its packet length.

   The second token the source receives is the token it transmitted at the end of its data.

      This token must arrive after it has finished transmitting its data and transmitted the token, so that the source no longer needs to insert its own data on the ring, and can forward the data received on the ring.

      If the token is busy, another ring interface has started transmitting data.

      If the token is idle, no other terminals have data to transmit, and the token can either be acquired by this source or will continue to circulate around the ring.

      A source can determine the beginning of the second token before it is received by counitng the number of bytes of data it has removed following the first token. This can be used to reduce the delay in the ring interface to one bit, rather than the token length - 8 bits.

7. Correct Reception

   If the source receives the same data that it inserted, it assumes that the data was also correctly received by the destination.

   If the data is distorted between the receiver and the source, it is possible for the transmitter to re-xmit data that has already ben correctly received

      Note: Sequence numbers are needed to prevent duplicate messages from being received

8. Fairness

   Token passing is fair because every ring interface gets a chance to transmit data before any ring interface gets a chance to send a second packet

9. Priorities

   some stations can be given higher priority by using more complicated tokens

   For instance, allow 4 types of tokens instead of 2, 011111100, 011111101, 011111110, and 011111111

      011111100 is the idle token for priority 1 terminals,

      011111101 is the idle token for priority 1 and 2 terminals

      011111110 is the idle token for priority 1 and 2 and 3 terminals, and

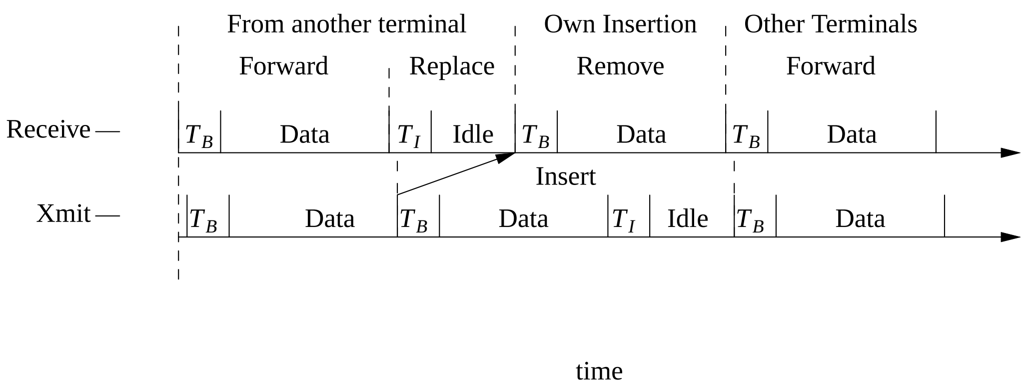
      011111111 is the busy token

   After ending transmission a station sends an idle token for priority 1 stations

   If that idle token circulates the loop, there are no priority 1 stations, and the station that inserted the token changes it to priority 1 and 2

   If that idle token circulates the loop, it reduces the priority to 1 and 2 and 3, and that token circulates until any station takes the token

10. The receive and transmit patterns at a single ring interface Reference [1] (fig 4.22a)



11. The maximum throughput approaches 1

12. Random Access (Lost Token)

    Tokens can be lost because of transmission errors or when a station that has the token fails

    If no busy or idle token is received in a specified time, the ring interfaces assume that the token is lost

    All stations with data, xmit

    If there is one station, that station succeeds and receives the data it transmitted.

    The system returns to normal when it transmits an idle token at the end of the data.

    If there is a collision, they all retry a random time later

1.2.2 Slotted systems

Operation

The slots are fixed size

The delay around the loop is adjusted to a multiple number of slots by a controller

A source inserts data in an empty slot

The data is removed by

1. The source

   This provides ARQ as in token ring

2. The destination

   Since packets only travel half way around the ring - on the average - this has the potential of doubling the throughput

3. By a controller

   When the same packet passes the controller more than once, the device that is supposed to remove the packet has failed

   The controller removes the packet

Disadvantage:

Partially full fixed-size packets are less efficient than variable size packets in token ring

predecessor to ATM

small packets to reduce inefficiency

fixed size packets imply switching rather than store-and-forward

Advantage

Formatted transmission makes it easier to detect failures

1.2.3 Register Insertion Ring

Destination removal with variable size packets

Operation

When src has a packet ready and register W is empty

Xmit packet up to max. size W

Xmit packet at any break in the data
independent of the size of the gap

Store Data from the loop in W

When register W is not empty, xmit W to the loop

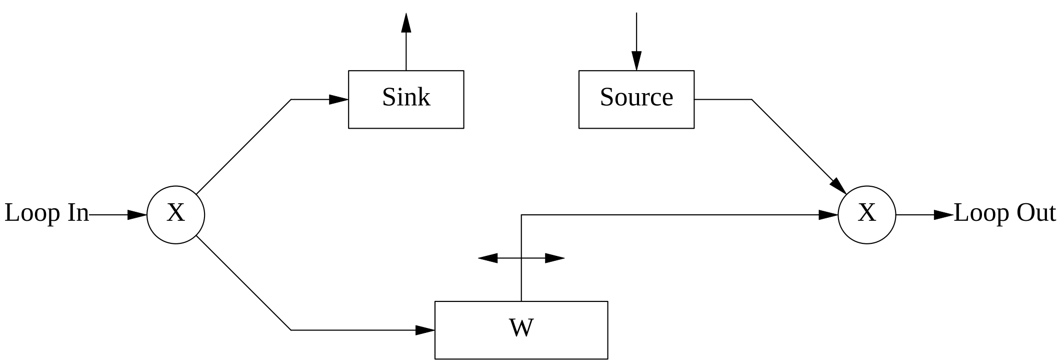
Delay W remains the same while data is received

Delay decreases during gaps in the data
and when this node removes data

Eventually W goes to zero and the source can transmit again

Data can be removed by the source, destination or controller,
as in slotted rings

Loop Delay -> to propagation delay, Δ_P, when loop is idle
and expands up to $M \times W + \Delta_P$ when the loop is busy



1.2.4 FDDI

Characteristics

Token loop

2 priorities of traffic

High priority traffic has a delay guarantee
packet voice

Technique

Each node measures time since last token visit

The node can always transmit the high priority traffic

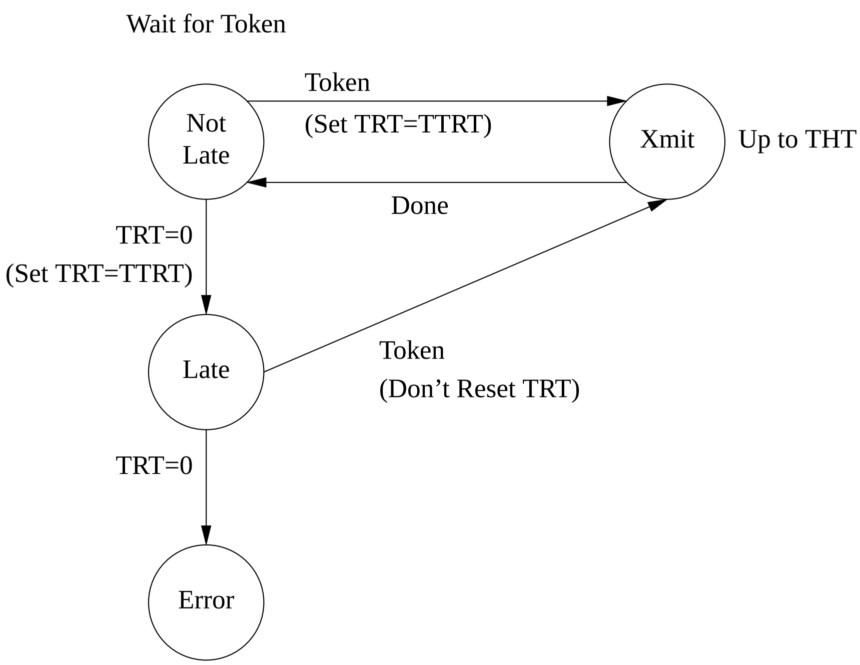
but, only transmits the low priority traffic if the token arrives early

Definitions

- TTRT - Target token rotation time
  - This is the interval that we would like to circulate within
  - It is the same for every terminal
- TRT(i) = Token rotation time
  - This is the time since the token was last received at terminal "i"
  - It is maintained at each terminal
  - In implementations, a terminal sets TRT to TTRT.
    TRT is decremented periodically
- THT(i) = Token hold time for terminal i
  - This is the amount of data that terminal "i" can transmit when it receives the token
  - High priority terminals can always transmit up to an allocated amount of data
  - Low priority terminals can transmit data up to TRT, the amount of time that is left before the token is late

Protocol for High Priority Sources

TRT is decremented periodically



Rule:

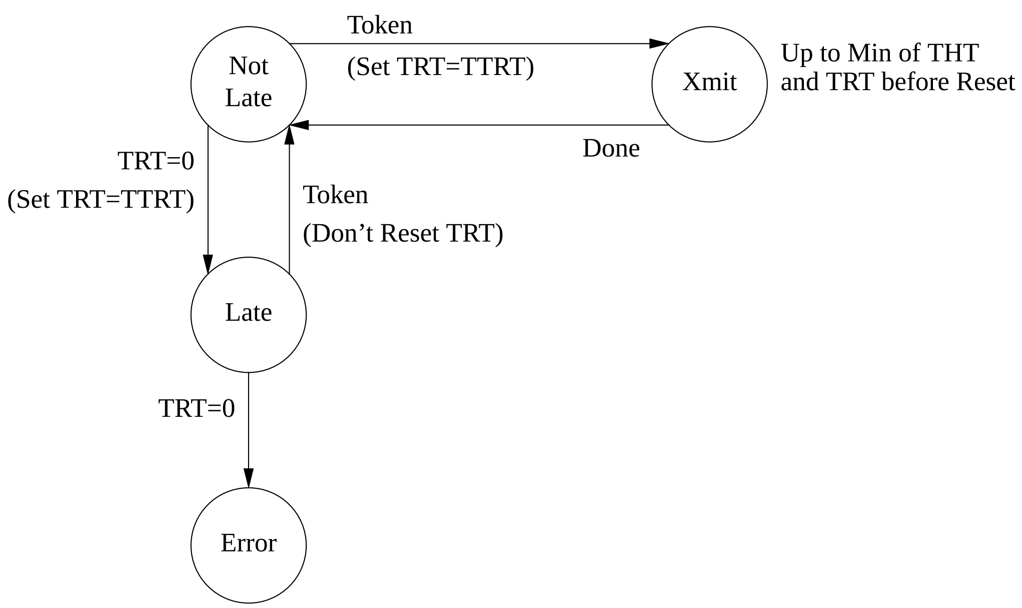
— Whenever the token arrives, xmit up to THT, if data is available.

— If the token is later than 2*TTRT, initiate error recovery procedures.

Protocol for Asynchronous Terminals

TRT decremented periodically

Wait for Token



Rule:

— If the token arrives on time, transmit for THT or time left on TRT, whichever is less

— If token arrives late, forward it and keep TRT

Constraints

High priority traffic

$$\sum_i \text{THT(i)} + \textit{Loop Delay} \le \text{TTRT}$$

α_i = the transmission time for the high priority traffic allocated to node *i* plus propagation time to node i+1 mod m, for nodes 0 to m-1.

$$T = \sum_{i=0}^{m-1} \alpha_i \le TTRT$$

the total high priority traffic in a cycle is less than the TTRT

The amount of low priority traffic is constrained by how early the token arrives.

Show that: A high priority packet is never delayed more than 2*TTRT and that the average token arrival time $\leq TTRT$

Token Arrival times

t_i is the time that the token arrives at node i in the current cycle

t_{i-m} is the time that the token arrives at node i in the previous cycle

Operation:

If $t_i - t_{i-m} < TTRT$, node i can send low priority traffic for $TTRT - (t_i - t_{i-m})$ seconds

If $t_i - t_{i-m} \geq TTRT$, node i only sends high priority traffic

Special case 1:
Only low priority nodes,
There is no constraint on how long a low priority node can hold the token (THT = ∞)
and zero propagation delay

Initially, $t_i - t_{i-m} = 0$ for all $0 \leq i < m$
No nodes are transmitting

All nodes become heavily loaded - xmit as much as they can

1. Node 0 xmits for $TTRT$ at t_0.
2. None of the other nodes can xmit at t_i for $1 \leq i \leq m - 1$
   because $t_i - t_{i-m} = TTRT$
3. Node 0 can't xmit at t_m
   because $TTRT_m - TTRT_0 = TTRT$
4. Node 1 xmits for $TTRT$ at t_{m+1}
   because $t_{m+1} - t_1 = 0$
5. Nodes 2->1 can't transmit at t_{m+2} to t_{2m+1}
   because $t_{m+i} - t_i = TTRT$
6. Node 2 transmits $TTRT$ at t_{2m+2}
   because $t_{2m+2} - t_{m+2} = 0$

All of the low priority nodes get an equal share of the bandwidth in order - Round Robin Polling

Special case 2: Only high priority nodes

$$\sum_{i=0}^{m-1} \alpha_i = T \leq TTRT$$

For each node i: $t_i - t_{i-m} = T$

The token circulates in time $T \leq TTRT$

General Case

The token arrives at node i at $\ldots, t_{i-m}, t_i, t_{i+m}, \cdots$
$t_j \le t_i$ for all $j \le i$

The maximum high priority traffic transmitted by node i is α_i

The maximum low priority traffic transmitted at t_i is

$$x_i = \begin{cases} TTRT - (t_i - t_{i-m}) & \text{if } t_i - t_{i-m} < TTRT \\ 0 & \text{if } t_i - t_{i-m} \ge TTRT \end{cases}$$

The time that the token arrives at node $i + 1$ is

$$t_{i+1} \le \max(t_i + \alpha_i,\ t_i + TTRT - (t_i - t_{i-m}) + \alpha_i)$$

$$\le \max(t_i,\ t_{i-m} + TTRT) + \alpha_i$$

a. Show $t_{i+m} - t_i \le 2 \times TTRT$ for all i

$$t_{i+1} \le \max(t_i,\ t_{i-m} + TTRT) + \alpha_i$$

$$\le \max(t_i,\ t_i + TTRT) + \alpha_i \quad \text{Since } t_{i-m} \le t_i$$

$$\le t_i + TTRT + \alpha_i$$

$$t_{i+2} \le \max(t_{i+1},\ t_{i+1-m} + TTRT) + \alpha_{i+1}$$

$$\le \max(t_i + TTRT + \alpha_i,\ t_i + TTRT) + \alpha_{i+1}$$

$$\le t_i + TTRT + \alpha_i + \alpha_{i+1}$$

By iteration, for $j \le m + 1$, $(t_{i-m+(j-1)} \le t_i)$

$$t_{i+j} \le t_i + TTRT + \sum_{k=i}^{j-1} \alpha_{i+k}$$

For $j = i + m$, $t_{i+m} \le t_i + TTRT + \sum_{k=i}^{i+m-1} \alpha_k$

$$\sum_{k=i}^{i+m-1} \alpha_k = \sum_{k=0}^{m-1} \alpha_k \quad \text{Since } \alpha_i = \alpha_{i \bmod m}$$

$$= T \le TTRT$$

Therefore, $t_{i+m} - t_i \le 2 \times TTRT$

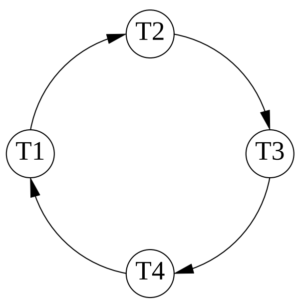
b. Show $\lim_{K \to \infty} \frac{t_{i+Km} - t_i}{K} \le TTRT$

See reference [1] pg 329.

Home work

FDDI

Consider a loop with 4 stations:



The propagation delay between terminals is 0. Ignore the time to transmit a token.

TTRT, TRT, and THT are defined in the class notes. The state machines that specify the operation of the low and high priority sources are also described in the class notes.

The target timed token rotation time at each terminal, TTRT, is 2.5 units.

T2 and T4 are low priority terminals with infinite THT, they transmit as much data as they are allowed.

— At time t=0 a data packet that requires 10.25 units to transmit arrives at T2.

— At time t=4 a data packet that requires 10 units to transmit arrives at T4.

T1 and T3 are high priority terminals with a token hold time, THT, of 1 unit.

— Packets that require 1 unit to transmit arrive at T1 at times 14, 17, 20, 23, 26, 29

— Packets that require 1 unit to transmit arrive at T3 at times 13, 16, 19, 22, 25, 28

A. Starting at time t=0, just after the first data packet arrives at terminal T2, up until time t=19:

  a. Specify the time that the token arrives at each terminal, starting with T2.

  b. The amount of data that the terminal has to send when the token arrives.
  (If the token and data arrive at the same time, the terminal can send the data)

  c. Calculate TRT when the token arrives

    Indicate when the token is late.

    If the token arrives at the same time that TRT expires, don't count it as late,

  d. If the terminal has data to send, calculate THT, the maximum amount of data that the terminal can transmit.

Place your answer in the form of a table:

T	time	D	TRT	THT
2	0	10.25		
3				
. . .				

B. If there are 3 high priority terminals with THT=1, and a packet arrives at each terminal every 3 units, what would happen and why?

2. Token Bus

Reference 1 section 4.5.4

Objective: Construct a virtual token passing loop on a physical bus network.
Techniques:

1. Centralized system
   - Hub Polling
   - A central controller transmits a sequence that invites nodes to Xmit
     — The controller can transmit to each source in sequence, which gives each source access to the same rate.
     — The controller can transmit to some sources more frequently to give them a higher rate than others
     — The sources don't collide with one another
   - Hub polling is used to provide quality of service guarantees in the IEEE 802.11 standard.
     — Polling sequences alternate with periods when the channel uses CSMA/CA, so that both periodic sources and random access sources can share the same channel.
     — Sources can join the polling list by transmitting to the controller during the random access periods.
2. Distributed System
   - One node has a token
     — When the node is done xmitting, it sends the token, an invitation to transmit, to the next node in a list.
   - New nodes are added to the list by regularly sending an invitation to join the list
     — If more than 1 node trys to join, a collision occurs
     — A collision resolution procedure is used to access subsequent invitations, and determines which node joins the token list first
   - Nodes leave the list by notifying the previous node.
     — The previous node forwards the token to the node that the exiting node had been transferring the token
   - Failures occur, the token is lost, if
     — A node leaves the list without notifying the previous node, or
     — There is a transmission error in the token
   - After a failure,
     — All nodes leave the list and transmit a request to rejoin the list
     — Contention resolution is used to form a new list
3. Implicit tokens - (BRAM, MSAP)
   - Instead of transmitting a token with the next address,
     The token is the idle channel following a transmission.
     — Each source knows its position in the list.

- If the next token site does not xmit, an extended idle sequence is the token for subsequent sources to transmit
  - — The time until the idle channel reaches the next token site is $< \beta + \delta$.
    β is the end-to-end propagation delay, and δ is to time to detect a busy or idle channel.
  - — The time until the xmission from an active source to reaches and is detected by all other sources is also $< (\beta + \delta)$.
  - — Therefore, when the channel is idle for $2(\beta + \delta)$, it is assumed that the next source in the token list will not transmit and the idle interval is an implicit token for the second source in the list
- Similarly, when the channel is idle for $2k(\beta + \delta)$, there is an implicit token for the $(k+1)^{th}$ token site in the list
- Tokens are not lost when sources leave the list without notifying the previous station

— Tokens cannot be lost because of transmission errors.

- Periodically, all sources leave the list and contend to rejoin to clean up lists with inactive sources.

3. Linear Topologies

Topologies and Access Protocols that use Fiber Optic Buses

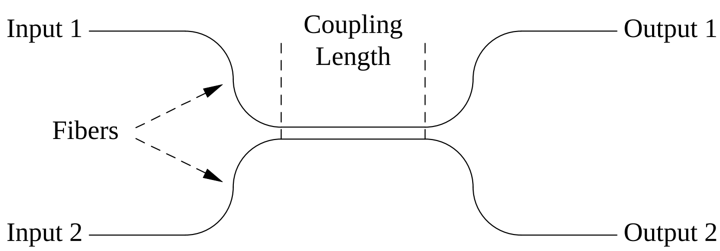
- Signals on fiber buses propagate in one direction, and topologies are constructed that allow all terminals to communicate.
- We will consider single bus topologies, including spiral and "D" networks, and Dual Bus networks.

3.1 Fiber Optic Couplers

Reference 2

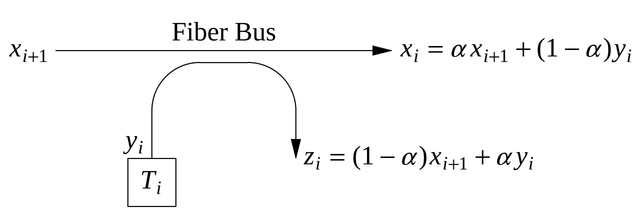
3.1.1 Passive Taps

Directional Couplers



- 2x2 Fused fiber couplers
  - — Operates by coupling energy from one fiber to another
  - — Dependent on proximity of the fibers and the length of the coupling
- **Symmetric:**
  The fraction of the energy that we couple from input 2 to output 1 =
  the fraction of the energy that we couple from input 1 to output 2.
- **Conservation of Energy:** The total energy out = the total energy in
- **Result:** We cannot build a directional coupler that puts all of the energy from two inputs onto one output. The energy on the second output would also equal the sum of the 2 input energies, and the total energy out would be twice the energy in.
- Directional couplers are not perfect, and some energy is lost in each device.
  Typically, the loss is about .2 db per device

Passive taps: Connecting Terminals to a Fiber Bus



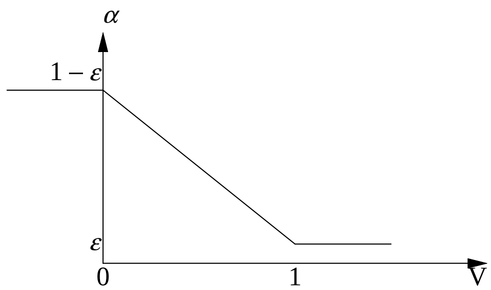
- The more energy that is put onto the bus, the more energy that must be taken off, even when the terminal is not transmitting.

3.1.2 Active Taps

Lithium Niobate Switches

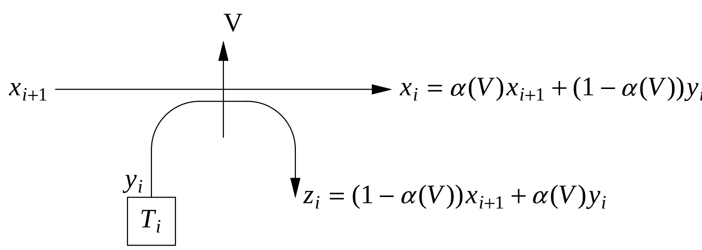
- pico seconds switching times

- applying an electrical signal changes the index of refraction and the light path
- If the characteristic of the switch is



the station applies $V = 0$ when it is not transmitting and $V = 1$ when it is transmitting.

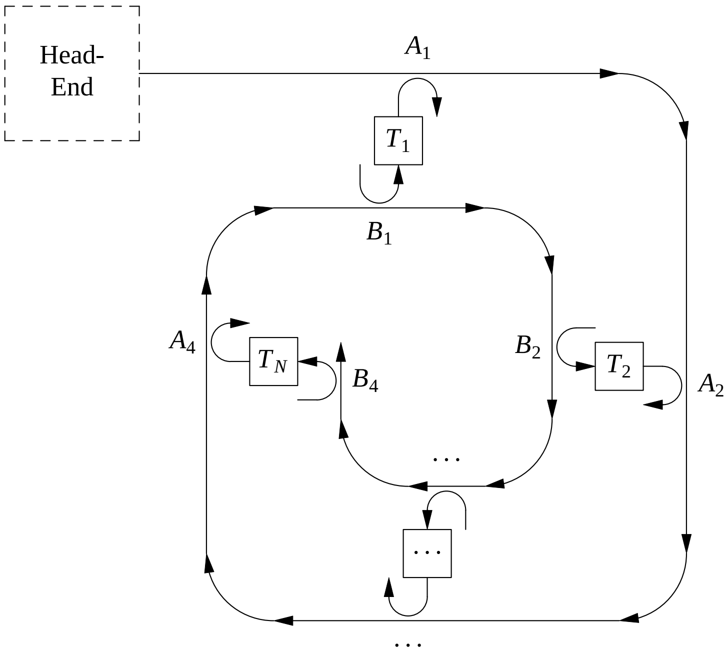
- If there is contention between 2 signals, instead of a collision, the downstream terminal wins.
- Active taps



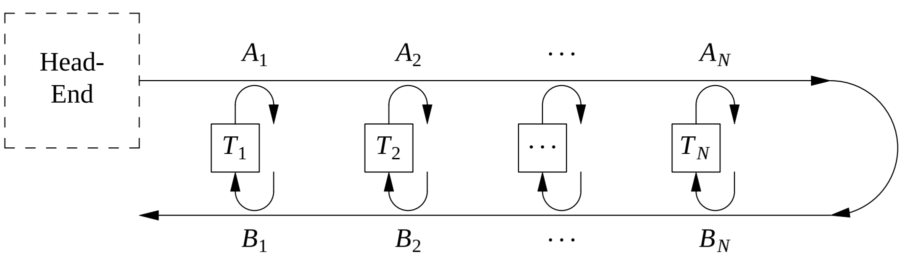
When a terminal wants to allow the signal from upstream to pass through it applies $V = 1$ that allows the signal to pass.

When a terminals wants to transmit, it applies a voltage $V = 0$ that allows more of its own signal to be inserted on the fiber.

3.2 Spiral and D Topologies



Spiral Network Reference [3]



D Network Reference [4]

3.2.1 Operation

- The transmit fiber passes each terminal, $T_1, T_2, \cdots, T_N$
  - — Each terminal, T_i, inserts data onto the fiber at tap A_i.
  - — An optional head end, prior to terminal T_1, may periodically transmit timing for the beginning of a slot, or may start a token for a token passing protocol.
- After every terminal has a chance to insert data on the fiber, the fiber once again passes every terminal.
  - — Each terminal, T_i, reads the data at tap B_i.
  - — In the D network, the receive fiber passes the terminals in the order $T_N, T_{N-1}, \cdots, T_1$.
  - — In the spiral network, the receive fiber passes the terminals in the order $T_1, T_2, \cdots, T_N$
  - — The affect of the receive order was a subject of active debate in the 80's, but had a very minor effect on the operation of the networks.
    - — In a token passing protocol, the spiral network is slightly fairer than the D network because the signal passes the receive tap in the same order as the transmit tap.
    - — For most purposes, the D and Sprial network are identical

3.2.2 Protocols on a D Network

We will consider 3 classes of protocols to share the facilities of a "D" bus network, I) Slotted Protocols, II) Token Passing Protocols, and III) Random Access Protocols

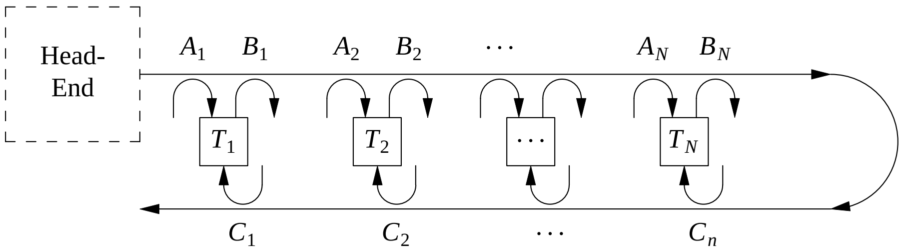
3.2.2.1 Slotted Protocol

- The headend periodically xmits
  - — a start of slot,
  - — followed by a busy bit that is zero,
  - — followed by a fixed size data area
- A terminal with data to xmit
  - — Receives the start of slot,
  - — Xmits busy bit =1
  - — Receives the busy bits transmitted by upstream stations
  - — If received busy bit = 0, it transmits data, otherwise, the slot is already full, and the terminal waits.
  - — The upstream terminals have priority over the downstream terminals.
- In order to implement this protocol, a terminal must be able to:

  a. Receive the signal at it's input to the network, so that it can identify the start of a slot, and

  b. simultaneously transmit its own busy bit, while receiving the busy bit transmitted by an upstream station, so that it does not overwrite a transmitted packet.

- A terminal has 3 taps, 2 consecutive taps on the transmit side of the bus and the a tap on the receive side of the bus.



1. Tap A, is used to read the busy bit from upstream stations
2. Tap B, is used to transmit the terminals busy bit and data.
3. Tap C, follows the bend in the "D" network, and is used to receive the signals transmitted by all of the stations

— The directional property of the taps prevents the signal transmitted at B from interfering with the signal received at A

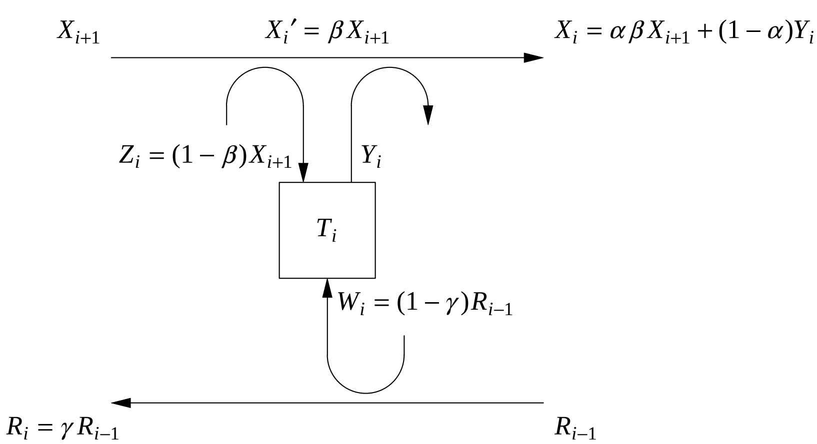
— The additive property of the tap, and the "Or'ed" reception strategy prevents T_i from changing a "1" in the busy bit to a "0".
If an upstream station transmits a busy bit of "1", the bit leaving this station is "1" independent of what T_i transmits.

— There are 2 tap losses on the transmit side of each terminal

— When the electronics is incapable of making a decision in a transmitted bit time:

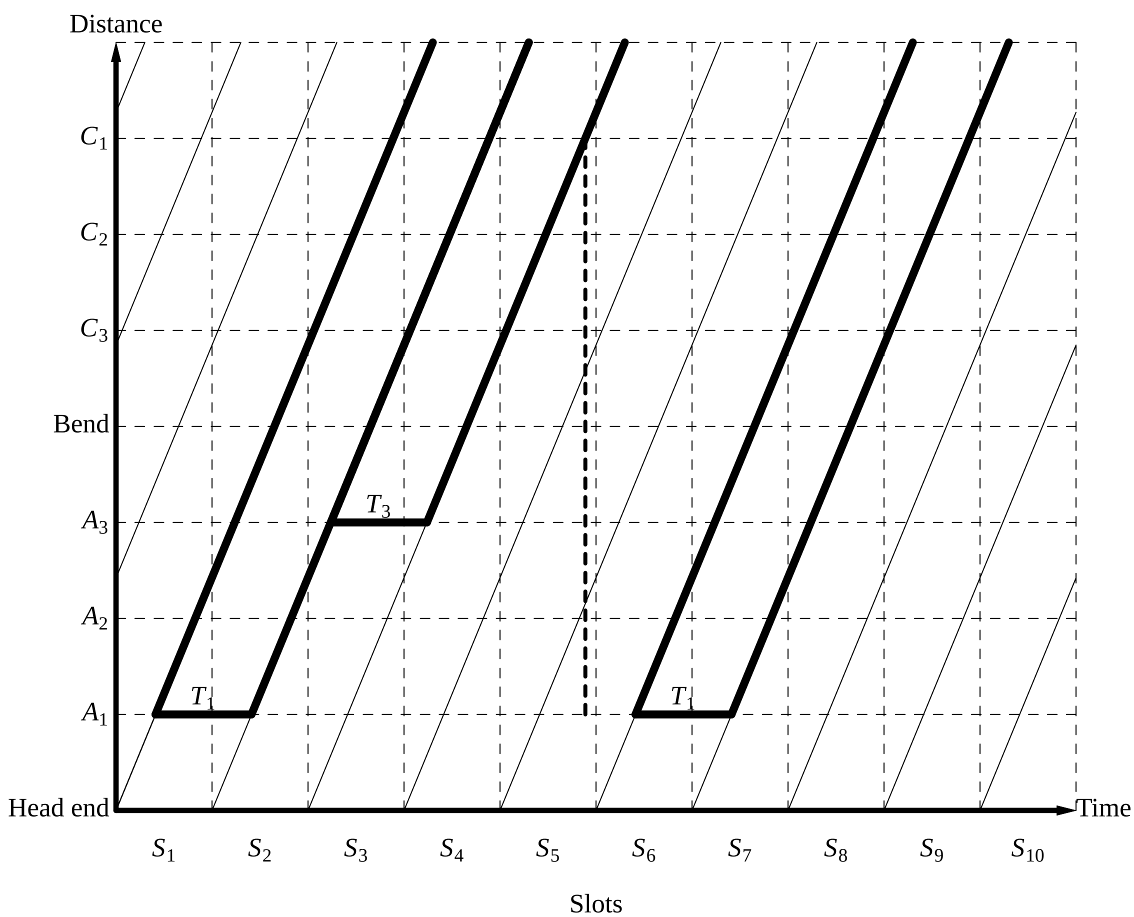
a. A fiber loop is inserted between the two taps on the transmit side to create a delay, or

b. The busy bit is longer than the other bits



- Fairness: After xmitting a packet, T_i waits for an empty slot at C_i, to give all terminals downstream from it a chance to transmit

- Example: At the beginning of slot 1, T_1 has 2 packets to transmit, and T_3 has 1 packet to transmit
  - Note that there are several slots in transit at any particular time



- The throughput of this scheme approaches 1 with passive taps.

— Every slot can be used to transmit data

- There is no reason to use switched taps

— The throughput of "fair" systems can be much less than 1 when a small number of stations transmit, and the propagation delay is much greater than the packet transmission time.

- In 150 Mbps MAN’s that transmit ATM cells, the throughput can be reduced to 1%.
- We will look at a dual bus protocol that is both fair and efficient.

3.2.2.2 Token Passing Protocol

- The terminal taps are the same as in the slotted system
- The head-end xmits a token with the busy bit = 0
- When a station with data to send sees a token it xmits a busy bit =1
- If it reads a busy bit = 0, it transmits data,
  Otherwise it waits
- A terminal that transmits data, sends a token with a busy bit = 0 at the end
  The data can be variable size
- The head-end reads the end of the D or Spiral
- When the head-end read a busy bit of 0, it sends a new token with a busy bit = 0
  All terminals with data to send have had a chance.
- Token passing is fair. However, when 1 terminal transmits, it must wait a round trip propagation time between slots.

3.3 Random Access Protocols on a Fiber Optic Bus

Reference [5]

- The propagation time from A_i to C_j, the write and read taps on the "D" network, may be much greater than a packet transmission time.
  - — We cannot directly apply CSMA or CSMA/CD
  - — There may be several packets propagating on the fiber at the same time.
- We look at three basic protocols: 1) Aloha, 2) Local CSMA, and 3) Local CSMA/CD.
  - — For each protocol we consider A) unslotted transmission, and B) slotted transmission. In a slotted transmission systems, a head-end generates slot timing
  - — For each of the six systems we consider using a) passive taps, and b) active (switched) taps
- Local protocols use the signal at the source, rather than the receiver to determine when to transmit, by receiving the signal from the upstream stations on the transmit side of the network.
- We will consider the time that the signals from all terminals arrive at the bend in the "D" network
  - — This simplifies the explanation and analysis by allowing us to ignore the packet transmission time versus the propagation delay.
- We will assume fixed size slots, although the unslotted systems may also use variable size packets
  The transmission time of a slot is normalized to 1 unit of time.

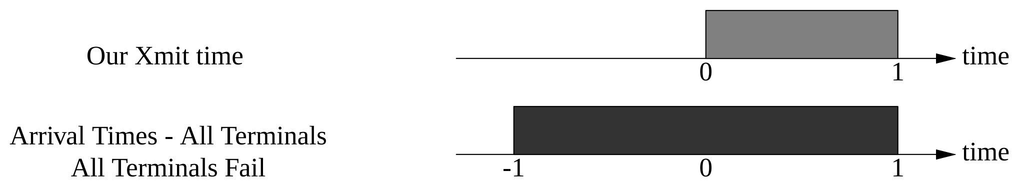
3.3.1 Aloha Protocols

3.3.1.1 Unslotted Transmission

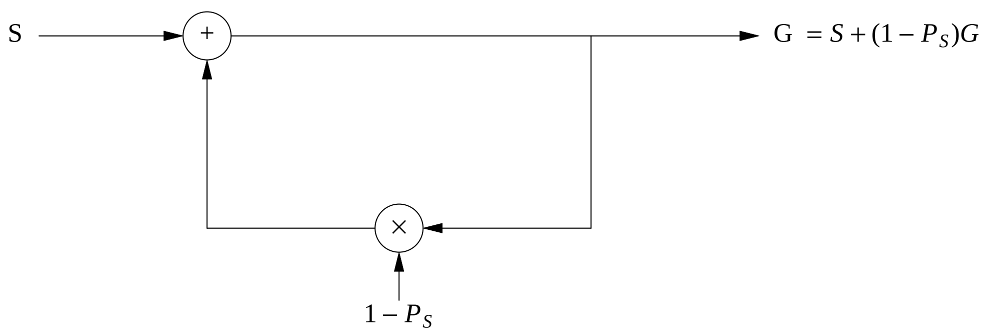
- There is no head-end, and terminals decide when they will transmit

3.3.1.1.1 With Passive Taps

- The transmit tap at a terminal can write on the bus, but does not read the signal from up stream. There is 1 tap on the transmit side of the bus, and it is arranged as in the "D" network.
- A collision occurs if the packet from any other terminal arrives at the bend in the "D" at the same time as our packet



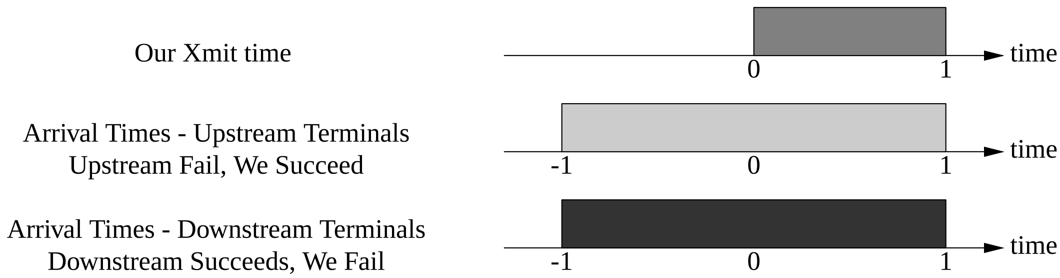
- $P_{success} = e^{-2G}$
- The throughput is $S = Ge^{-2G}$



- The maximum throughput occurs at $\frac{dS}{dG} = 0$
  $G = 1/2$ and $S_{\max} = .5e^{-1} = .184$.

3.3.1.1.2 With Switched Taps (Lithium Niobate)

- The passive tap on the transmit side of the "D" network is replaced with a switched tap.
- When a terminal transmits, it switches its own signal onto the bus, and removes the signal from upstream terminals.
- A downstream terminal wins a collision.



- Similar to perfect capture in the radio domain
- Normalize the linear distance of the input path.
  T_1 is at $x = 0$, and T_N is at $x = 1$
- The probability of success for a terminal at position x is:

$$P_{succ}(x) = e^{-2\int_x^1 G(y)dy}, \quad S(x) = G(x)e^{-2\int_x^1 G(y)dy}.$$

- Assume there is a terminal in each interval Δx, and that the arrivals in each interval Δx is the same, $S(x) = S$.

  The total arrival rate is $S_{tot} = \int_0^1 S(x)dx = S\int_0^1 dx = S$.

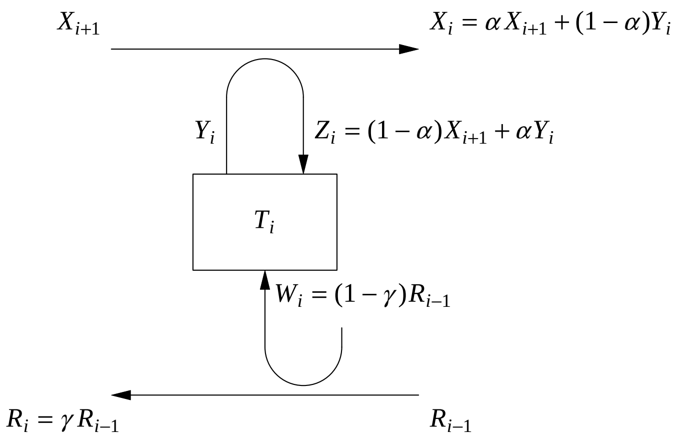
- $$\frac{dS}{dx} = 0 = G'(x)e^{-2\int_x^1 G(y)dy} + G(x)e^{-2\int_x^1 G(y)dy}\frac{d}{dx}\left(-2\int_x^1 G(y)dy\right)$$
- $G'(x) + 2G^2(x) = 0$
- Boundary Condition: $G(1) = S$, since T_N always succeeds on the first try at $x = 1$
- The solution of the differential equation is $G(x) = \dfrac{S}{1 - 2S(1 - x)}$
- As long as $S < 1/2$, $G(x) < \infty$ for all x.
- Therefore, as long as $S < 1/2$ all of the arrivals get through, and the maximum value of $S_{tot} = 1/2$
- If $S(1) \rightarrow 1$, the total throughput goes to one, since T_N always succeeds, but none of the other terminals can get through

3.3.1.2 Slotted Transmission

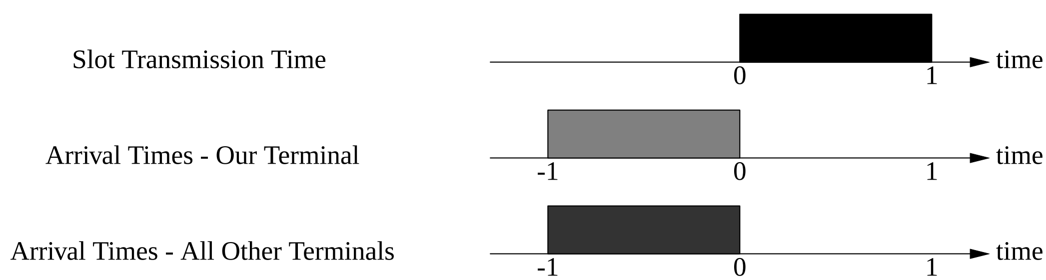
- The head-end on the "D" network transmits the start of slots, and the terminals must read the start of slot on the transmit side to determine when to transmit.

3.3.1.2.1 Passive Taps

- Taps: Unlike slotted and token passing systems, the sources do not simultaneously read and write a bit, and can use a single passive tap on the transmit side of the bus.



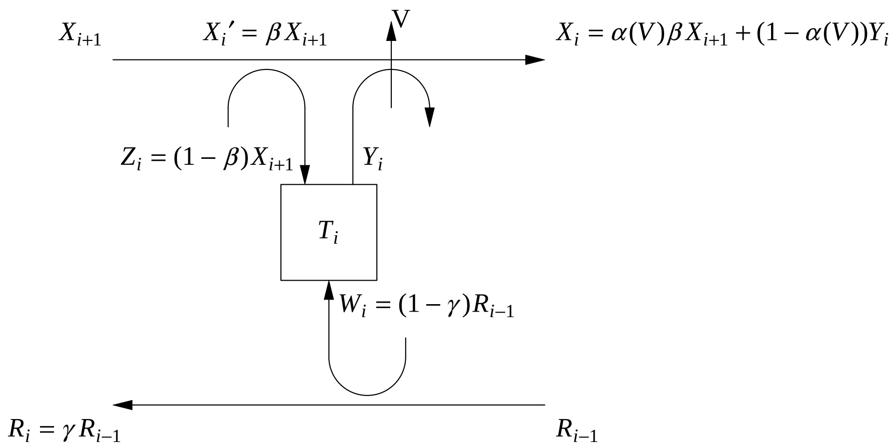
- Collision Diagram



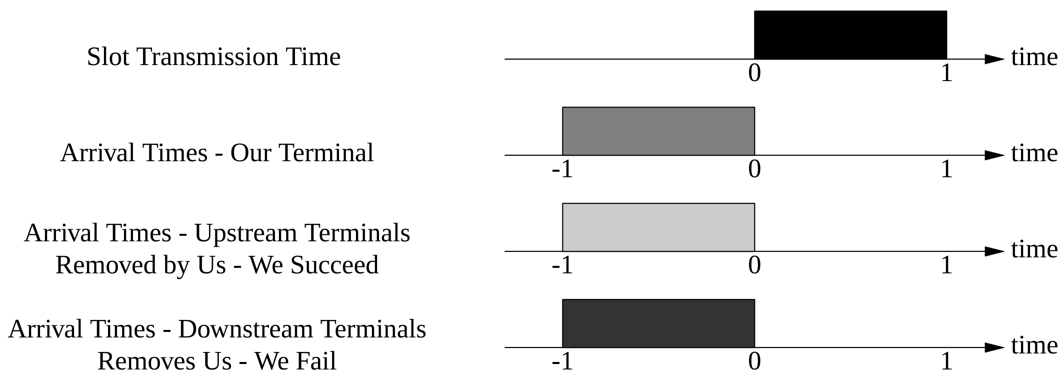
- There is a successful transmission if and only if one terminal transmits in a slot:
  $P_{sucess} = e^{-G}$, $S = Ge^{-G}$, $S_{\max} = e^{-1} = .36$

3.3.1.2.2 Switched Taps

- On the transmit side, a terminal must receive the start of slot from the head end, and must replace the signal on the bus with its own signal when it transmits.
  This may be accomplished with a switched tap that isn't completely switched when a source is not transmitting, but is more easily done with 2 taps on the transmit side, the first always reads the bus, and the second replaces the signal on the bus:



- Collision Diagram



- When there are multiple terminals that transmit in a slot, one terminal always wins.
  The throughput of this system can be 1, when the downstream terminal always has something to send, but what is the throughput when every station gets the same throughput?

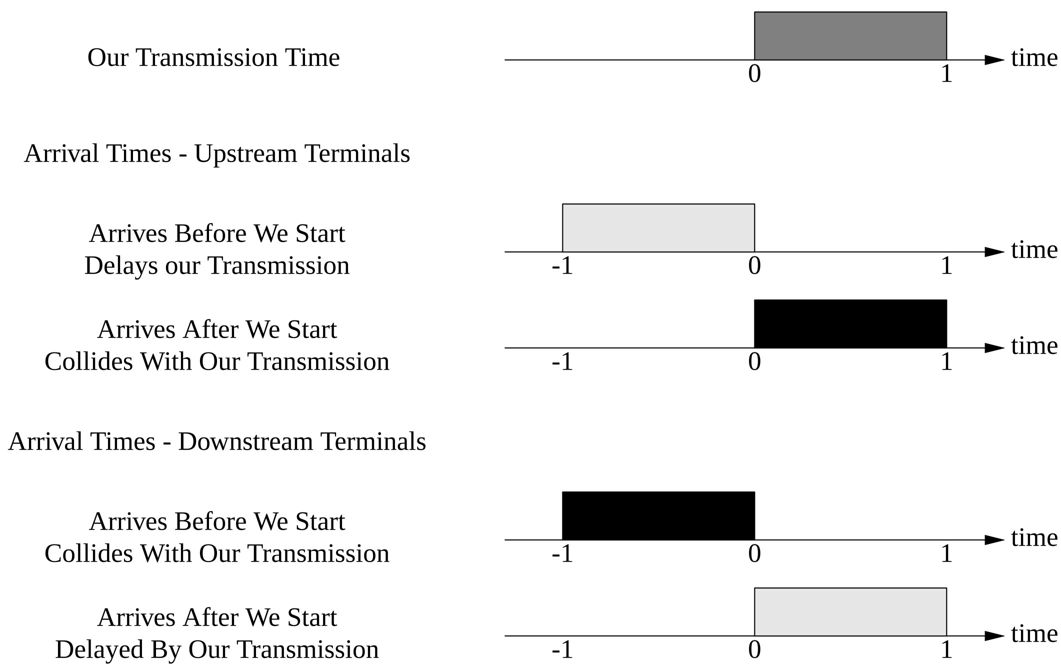
3.3.2 Local CSMA Protocols - LCSMA

- Receive upstream transmission and delay our transmission if there is a transmission in progress.

3.3.2.1 Unslotted Transmission

3.3.2.1.1 With Passive Taps

- We use a two tap configuration with a single passive tap on the transmit side, as in slotted Aloha with passive taps.
- Collision Diagram:



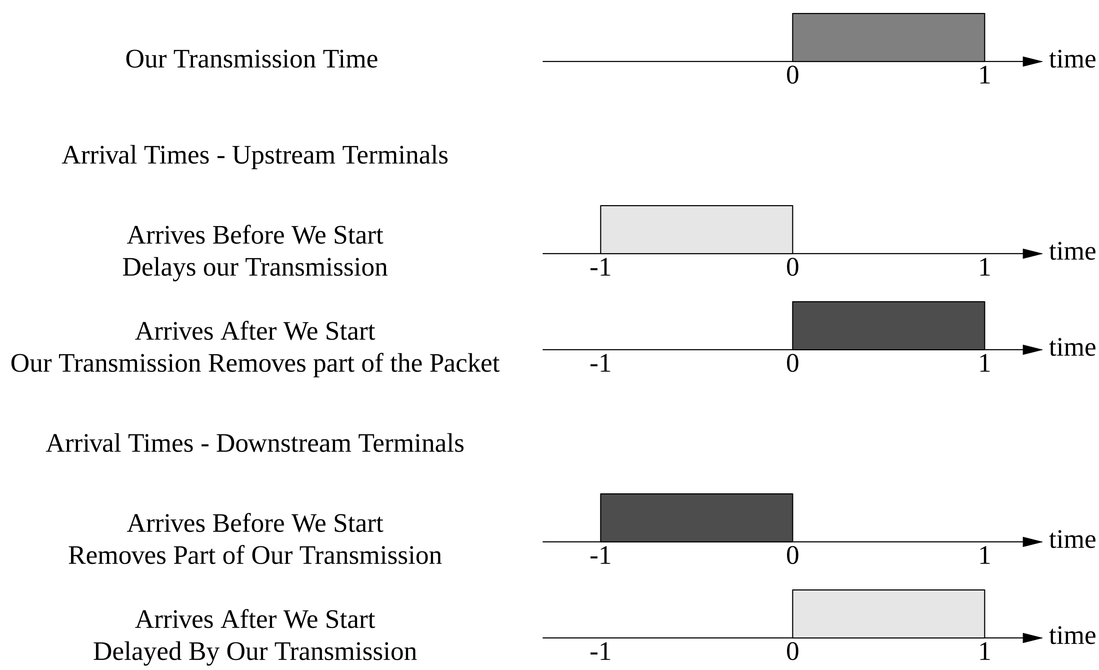
- The analysis of this system is more complicated than the earlier analyses because it must take into account the fact that not all of the packets are transmitted.

  — Packets that are not transmitted do not interfere with downstream terminals.

- The throughput of the system is about 29%, as compared with the 18% obtained with Aloha with passive taps

3.3.2.1.2 Switched Taps

- The tap configuration is the same as in slotted Aloha with switched taps.
- Collision Diagram



- The analyses of these systems must take into account the fact that not all of the packets are transmitted, and that other packets are partially transmitted before they are aborted.
- The throughput of the system with active taps about 57%, as compared with the 50% obtained with simple Aloha with switched taps.

3.3.2.2 Slotted Transmission

In a slotted system there isn't any data being transmitted before we begin transmission in a slot.

Therefore, the operation is the same as slotted Aloha with passive and switched taps.

If the first bit in a slot is the busy bit, and we read and write the busy bit, and defer if the bit is already busy, we obtain the sotted systems on the spiral and "D" network, and can construct fair protocols as we had in those networks,

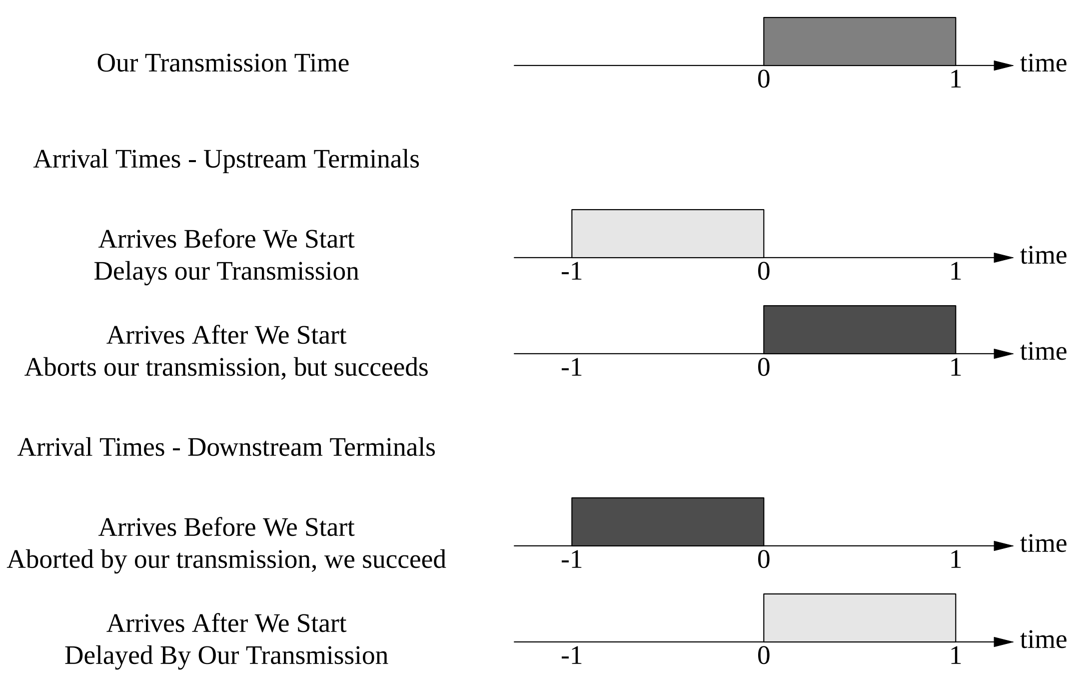
3.3.3 Local CSMA/CD Protocol - LCSMA/CD

- Detect upstream transmission before transmitting
  - — With passive taps we delay our transmission to avoid colliding with the transmission in progress
  - — With active taps we can delay our transmission to give priority to the upstream terminal, or switch the upstream transmission off of the fiber and transmit
    In an unslotted system, switching the transmission off the fiber increases the transmission interval and wastes the time that that transmission has been in progress.
- Continue to read incoming transmission while transmitting our signal.
  - — With passive taps we stop our transmision to avoid a collision.
    The transmission interval is longer than that with a single transmission.
  - — With switched taps, we can switch the incoming signal off the fiber until the end of its transmission, so that downstream terminals do not see an extended transmission interval.

3.3.3.1 Unslotted Transmission

3.3.3.1.1 Passive Taps

- Sources have 2 passive taps on the transmit bus, as in the slotted access protocol so that they can receive the signal on the bus while transmitting.
- Collision Diagram:



- The collision diagram has the same success rate as LCSMA with switched taps, therefore the throughput is 57%.

3.3.3.1.2 Switched Taps

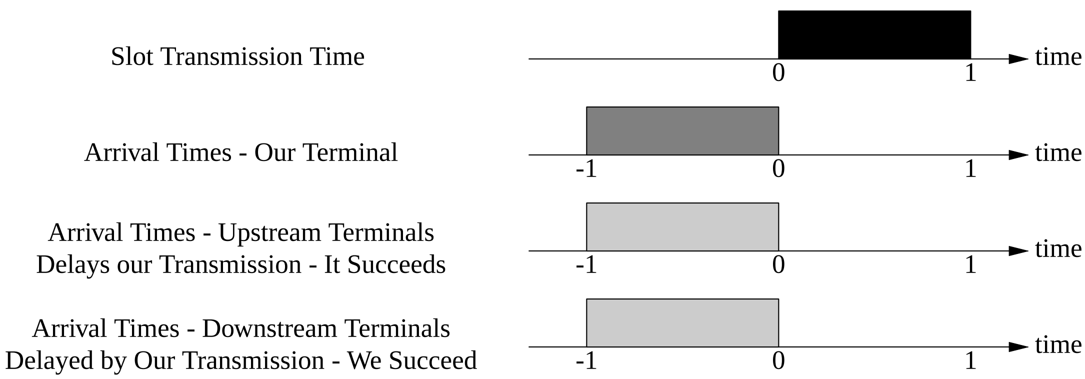
- The terminal configuration is the same as slotted Aloha with active taps.
- If a terminal elects to remove an incoming packet during its own transmission, the operation is the same as LCSMA with switched taps.
- If a terminal elects to abort its own transmission when an incoming packet is received, the operation is the same as LCSMA/CD with passive taps.
- Since the throughput of both of these systems is 57% this is the throughput of this system.

3.3.3.2 Slotted Transmission

- The data packets start with a busy bit, and the operation is the same as in a slotted bus network

3.3.3.2.1 Passive Taps

- The terminal configuration is the same as the slotted bus.
- Collision Diagram:



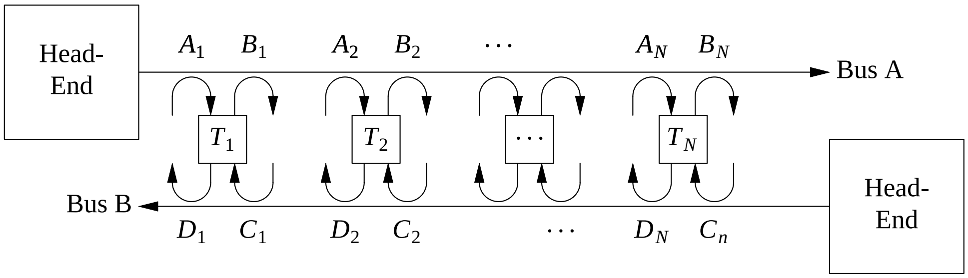
- There are no slots with collisions the throughput $\rightarrow 1$
- We can obtain fair operation with a protocol that waits for an empty receive slot before transmitting its next slot as in slotted "D" networks.

3.3.3.2.2 Switched Taps

- A terminal can elect to have its own packet succeed, or can let an upstream station succeed.
- By replacing an upstream packet with probability P_i that is a function of a terminals position on the bus, we can achieve fair operation without letting idle packets circulate the bus.
- This makes it possible to achieve fair operation at a throughput of 1, independent of the β of the bus.

4. Dual Bus Networks

4.1 Topology



4.2 Operation

- DQDB is a slotted bus network.
- The head-ends generate the start of slot on each bus.
  - — The beginning of each slot contains the start of slot followed by a busy bit and a reservation bit.
  - — Both the busy bit and reservation bit are initially "0"
  - — The data segment of a slot is an ATM cell, 53 bytes.
- The sources determine which bus to transmit on to reach the destination, and acquire a slot on that bus.
  - — A source can receive the busy and reservation bits from upstream stations, while transmitting in that bit, as in the slotted "D" network.
- When a source transmits a slot on Bus A, it transmits a "1" in the busy bit on Bus A, while receiving the busy bit from upstream stations.
  - — If the source receives a "1" in the busy bit, the slot is in use, and the source defers to the upstream source.
- The reservation bit on the opposite bus is used to give the downstream sources equal access to the bus.
  - — Before a source transmits on Bus A, it transmits a "1" in the reserrvation bit on Bus B that is received by all upstream sources on Bus A.
  - — The source receives the reservation bit from upstream sources on Bus B while transmitting its reservation, and retrys in the next reservation bit if the bit is already "1".
  - — Reservations occur at the same rate as data slots, and will not limit the throughput.
- Each source maintains distributed queues of the reservations from downstream sources and its own slots
  - — There are two distributed queues at each source.
  - — There is a distributed queue for the reservations received on Bus B, which reserve slots on Bus A,
  - — And a distributed queue for the reservations received on Bus A, which reserve slots on Bus B.
- Each distributed queue consists of two counters,
  - — C_B, are the reservations from downstream sources that have arrived before the packet that the source will transmit and which have not been serviced.

— C_A, are the reservations that arrive after this source's packet, but before this source transmits its slot. (Note: A source may place the reservation after it transmits the packet)

- A reservation from a downstream source is serviced when an empty slot, busy bit =0, passes the source.

— A downstream source can use this slot to transmit its data.

- When $C_B = 0$, the source transmits its packet in the next available slot, a slot in which the busy bit is "0".

  — After transmitting a packet, a source sets $C_B = C_A$, and $C_A = 0$

- The distributed queues operate with partial information.

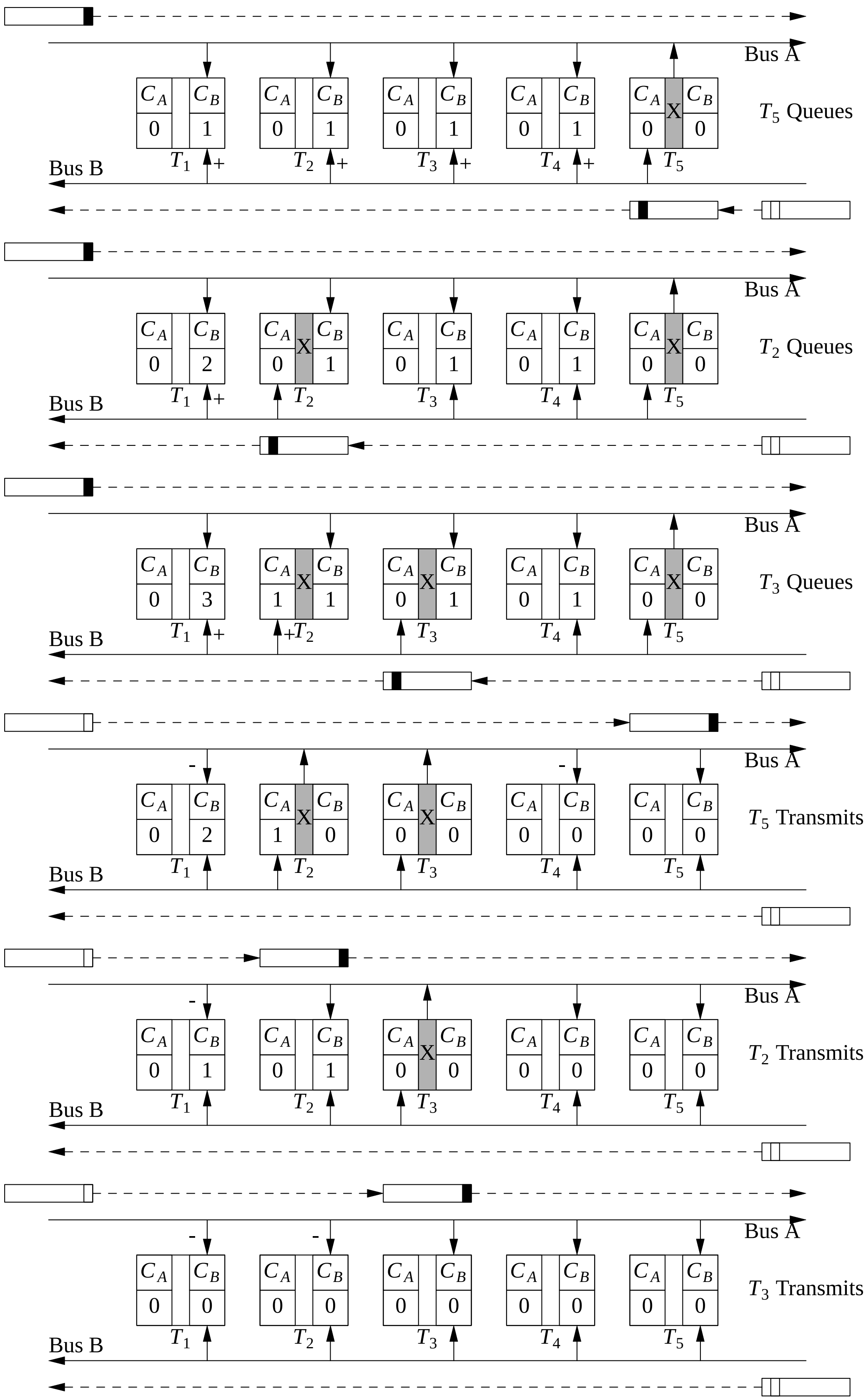
  — A source does not receive the reservations placed by upstream sources, but upstream sources have first access to the slots.

  — If we ignore the propagation delay, the distributed queue is a FIFO queue.
  The sources that place reservations first, transmit their slots first.

- Example:

— Slots arrive at nodes 5, then 2, then 3 to be transmitted on bus A, while Bus A is busy transmitting packets from upstream sources on Bus A.

— Bus A becomes available, and the packets from these three sources are transmitted in the order they arrived.


Bus A
Bus B
T_5 Queues
T_2 Queues
T_3 Queues
T_5 Transmits
T_2 Transmits
T_3 Transmits
T_1
T_2
T_3
T_4
T_5
C_A
C_B

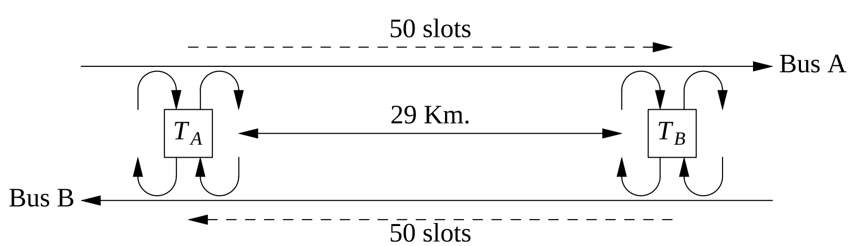
- Sources place 1 reservation at a time
  - — A source places a reservation for a slot, and does not place another reservation until the slot is transmitted.
  - — This prevents a downstream source with a large file transfer from placing a large number of reservations and locking out upstream sources.
  - — When there are simultaneous file transfers from multiple sources, the protocol provides round robin service to each of the sources

DQDB is the first network:

- — to do more with a bidirectional network than treat it as 2 one way networks
- — where the rate and propagation delay caused an unexpected failure.

4.3 Unfairness

- In a large β network, sources that are separated by multiple slots can obtain very different throughputs
  - — DQDB is a metropolitan area network, that spans up to 50 Km., operates at rates up to 155 Mbps, and transmits ATM cells.
  - — If two sources are 29 Km. apart, at 150 Mbps, there are 50 ATM cells in transit between the sources
- The source that starts transmitting first receives a much higher throughput.
- Example:
  - — Sources A and B transmit long files of 53 byte cells on Bus A
  - — The sources are 29 km apart and transmit at 150 Mbps.
    There are 50 slots in transit between A and B



A. Source A is actively transmitting slots, and has at least 50 slots in transit on Bus A when the file at Source B arrives, source A get 99% of the slots

- B xmits a request that gets to A 50 slots later
- A lets one empty slot pass that gets to B 50 slots later,
  100 slots after B transmits the request
- B transmits one slot then sends another request, which reults in another emmpty slot in another 100 slot times.

B. Source B is actively transmitting slots, and has at least 50 reservations in transit on Bus B when the file at Source A arrives.

- Source B gets much more throughput than source A, but the operation is more difficult to explain

4.4 Bandwidth Balancing

References [6, 7].

1. **OBSERVATION:** For DQDB Protocol every source can determine the utilization at the end of the bus.

- For each slot xmitted on bus A, each station either:

  Sees a busy bit on bus A,

  a request on bus B, or

  xmits a slot itself.

- **Therefore,** $\rho_A = \dfrac{\text{Busy bits on bus A + Rqst bits on Bus B + Xmitted Slots on Bus A}}{\text{total slots on Bus A}}$

2. **RATE CONTROL -- FOR DQDB**

- Each Active User

  — Measures Busy Bits + Request Bits

  — Takes Fraction $\alpha < 1$ of Remaining Slots

- THEN

  — System Converges to Fair Operation

  — Convergence is Faster for Smaller α

  — When $\alpha = 1$ Convergence Time Is Infinite
  Original implementation

- Example $\alpha = .9$

NODE A		--	NODE B	
Measure Bsy+Rqst	Take		Measure Bsy+Rqst	Take
0	.9*1=.9		-	-
0	.9*1=.9		.9	.9*.1=.09
.09	.9*.91=.82		.82	.9*.18=.16
.16	.9*.84=.76		.76	.9*.24=.22
.22	.9*.78=.7		.7	.9*.3=.27
.27	.9*.73=.66		.66	.9*.34=.31
.31	.9*.69=.62		.62	.9*.38=.34
. . .	. . .		. . .	. . .
.474	.9*.526=.474		.474	.9*.526=.474

3. A Simple Implementation
   1 counter -- no change in transmission protocol

- To Obtain Rate 8/9, at each source,

  — Count the slots that the source transmits

  — Each time 8 data slots are transmitted,
  Increment C_B by 1

  — Put a dummy request in the request queue to leave an extra empty slot after every 8 of our own slots

- RESULT:
  A source takes *at most* 8/9 of the available slots

a. The steady state throughput at a source is $T_i = \alpha_i \, [1 - \sum_{j \neq i} T_j]$

b. When there are N sources trying to transmit as many slots as they can,

$T_i = T$ for *all* i

$$T \quad = \alpha[1 - (N-1)T] = \frac{\alpha}{1 + \alpha\,(N-1)}$$

c. Example

$\alpha = .8$

N = 3

$$T = \left[\frac{4}{13}, \frac{4}{13}, \frac{4}{13} \right]$$

Unused Bandwidth = $\frac{1}{13}$

d. If the sources use different α_i, they obtain different rates
The sources have different priorities

Homework

DQDB with bandwidth balancing

A. There are 8 users that are trying to transmit as many slots as they can.
4 of the users are high priority and each takes a fraction $\alpha_H = .9$ of the remaining bandwidth.
The high priority users each have a throughput T_H.
What fraction α_L of the remaining bandwidth should each of the 4 low priority users take so that their throughput $T_L = T_H/2$?

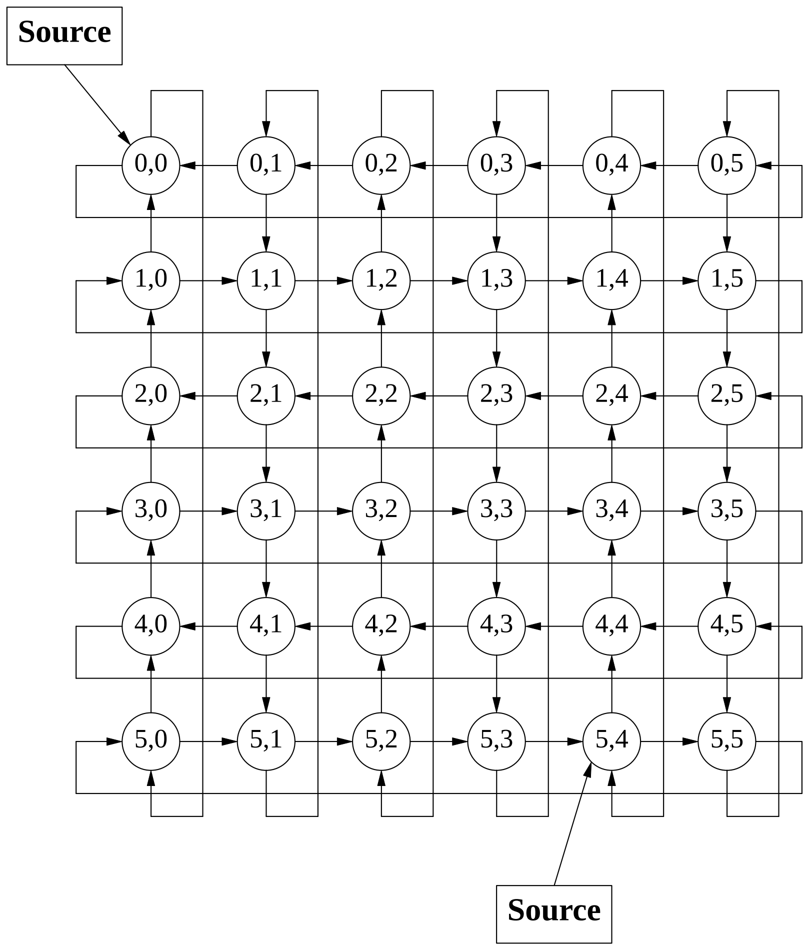
B. What fraction of the bandwidth is unused?

C. If there are I high priority users and J low priority users, and the users use α_H and α_L in part A, what is T_L/T_H?

5. 2-Connected Networks

5.1 The Manhattan Street Network [8, 9, 10]

5.1.1 Topology



Originally proposed as a metropolitan area network

2-connected, point-to-point links - like FDDI

One way streets on surface of a torus
By removing corners and edges, there are fewer hot spots in the network.

Distance between nodes grows as $\sqrt{N}$ rather than linearly.
Fewer resources to transmit a packet → higher throughput.

Sources connected to the nodes
Sources replaced by circuit boards

Regular structure → route based on source and destination address.

In Manhattan if you are on 2^{nd} ave., and 43^{rd} St., you can get to 5^{th} ave. and 72^{nd} St. without asking directions.

With wrap around links, if there are N rows, the distance to the destination row is calculated modulo N. If the distance is > N/2, going in the other direction is shorter.

Why has this become a switch?

— Regular wiring plan for connections between chips, boards and on the back plane

5.1.2 Slotted System

1. Fixed size slots are transmitted in the network
   - At a node the slots from different links are exchanged without using store-and-forward techniques
2. In a network:
   - The clocks have the same (approximate) rate, but different phases at each node
     — The physical slots are longer than necessary to carry the bits, so that slight differences in the clocks do not accumulate
   - The xmission delays are different
     — The slots must be lined up to perform switching
3. In a circuit:
   - There is a common clock
   - All Xmission delays are approximately the same
4. Regulates flow by only allowing new packet to enter when there is an empty slot

5.1.3 Deflection Routing

Uses a slotted system

This routing rule is one of a family of rules known as hot-potatoe routing rules

In general, these rules do not operate on slotted systems

Other members of this family will be described when we discuss routing

Operation:

— If 2 or more packets select the same output, and there are not sufficient output buffers, some of the packets are forced to take another path.

In a slotted system, at least one packet can take the preferred path.

— Operates with or without output buffers

— Output buffers decrease the probability of deflection

— A small number of output buffers are useful when successive slots are uncorrelated, but not when there are large file transfers

Problem with calculating buffer requirements in ATM analysis

The buffers in first generation ATM switches were an order of magnitude too small for the desired discard probability

Found buffers should be proportional to message size, not slot size

Advantage of deflection routing in the MSN

— At many intermediate nodes, the distance to the destination is the same along both output paths

If one arriving slot doesn't care which path it selects, it will not deflect another arriving slot.

— The penalty for deflections is small

When a slot selects the proper path, the distance to the destination is decreased by 1

When a packet is deflected, the distance to the destination is increased by 3
A deflected packet is forced to go around the block

Following a deflection, at the next node either path has the same distance to the destination

— Probabilistic components are introduced into deflection routing in two ways:

1. When more than one output path that is equally "good",

   The paths are equally "good" if they have the same distance to the destination.

   The paths are available if there is sufficient buffering to accomodate all arriving packets that select the path.

   the output paths are selected with equal probability.

2. When some of the packets must be deflected, the packets that are deflected are selected with equal probability.

   In the MSN if both packets select the same output, and there isn't a buffer available, each packet has a probability of 1/2 of being deflected.

— Advantage of probabilistic components

— At any node, for any distribution of flows in the network, all slots have a non-zero probability of progressing toward the destination.

A network in which packets cannot progess toward a destination is live-locked.

— This does not guarantee that the network is stable.

There may be a set of arrivals that result in the average progress toward the destination being negetive.

In the MSN, it has been shown that the network is stable for 2 simultaneous file transfers

— Position dependent deflection rules have been proposed for the MSN, and have been found to result in livelocks

— The position dependent rules that have been tried include rules that give preference to packets that go straight, rather than turning and rules that give preference to packets that are closer the their destination.

— In both of these cases, flows can be found that result in the packets in some of the flows always being deflected and never progressing toward their destination

— If I were to investigate deflection routing again, I would try deflections based on age.

— The average penalty remains the same, since one packet progresses to d-1 and the other changes to d+3, but

— The variance of delay decreases since the probability that old slots become older is decreased.

— When we investigate buffer control mechanisms used by Merlin and Schweitzer in general S&F nets we will see that these mechanisms are dead-lock free.

Home work

In the 36 node MSN, consider a slot from node 0,0 to node 3,3

A. Determine all equal length shortest paths

B. At what fraction of the nodes does either output path have the same distance to the destination?

C. Repeat 1 and 2 assuming that the slot is deflected at the first node at which one output path is preferred over the other.

5.2 Shuffle-Exchange switches and networks

5.2.1 Generalized Perfect Shuffle Switches

- Minimum number of PxP switches to get from P^{i+1} inputs to P^{i+1} outputs
  in most applications, $P = 2$.
- Self routing - At each stage of the switch, the path is determined by the destination address.
- Used in ATM switches and many routers

5.2.1.1 Topology

1. There are P^{i+1} inputs/outputs
   Where P and i are any integers.
2. There are (i+1) stages in the switch, numbered 0 to i
3. Each stage has P^i switches numbered from 0 to $P^i - 1$.
   — Switch K is represented by the i-tuple, $(k_{i-1}, k_{i-2}, \cdots, k_0)$
   - Where $k_j = 0, 1, \cdots, P-1$.
   - And $K = \sum_{j=0}^{i-1} k_j P^j$.

   — We can calculate k_i recusively:
   - $d_0 = K$
   - $k_j = d_j \bmod P$
   - $d_{j+1} = (d_j - k_j)/P$
4. Example: P=4, i=5
   — There are 5 stages of switches, and each stage has $4^5 = 1024$ switches.
   — The switches are numbered $0 \le K \le 4^5 - 1 \rightarrow 0 \le K \le 1023$
   — Switch $K = 727$ is represented by a 5-tuple that is calculated as
   - $d_0 = 727$, $k_0 = 727 \bmod 4 = 3$
   - $d_1 = (727 - 3)/4 = 181$, $k_1 = 181 \bmod 4 = 1$
   - $d_2 = (181 - 1)/4 = 45$, $k_2 = 45 \bmod 4 = 1$
   - $d_3 = (45 - 1)/4 = 11$, $k_3 = 11 \bmod 4 = 3$
   - $d_4 = (11 - 3)/4 = 2$, $k_4 = 2 \bmod 4 = 2$
   - $K = (2, 3, 1, 1, 3) = 2 * 4^4 + 3 * 4^3 + 1 * 4^2 + 1 * 4^1 + 3 * 4^0$
     $= 2 * 256 + 3 * 64 + 1 * 16 + 1 * 4 + 3 = 512 + 192 + 16 + 4 + 3 = 727$
5. Each switch is a PxP switch, and switch K at stage L is connected to the P switches,
   $(P * K) \bmod P^i$, $(P * K + 1) \bmod P^i, \ldots, (P * K + P - 1) \bmod P^i$
   at stage $L + 1$.
   - Since $K = \sum_{j=0}^{i-1} k_j P^j$, $PK \bmod P^i = \left[k_{i-1} P^i + \sum_{j=1}^{i-1} k_{j-1} P^j \right] \bmod P^i = \sum_{j=1}^{i-1} k_{j-1} P^j$
   - Switch $K = (k_{i-1}, k_{i-2}, \cdots, k_0)$ at level L is connected to switches
     $(k_{i-2}, \cdots, k_0, 0), (k_{i-2}, \cdots, k_0, 1), \cdots, (k_{i-2}, \cdots, k_0, P-1)$
     at level $L + 1$

- In the example: Switch $K = 727 = (2, 3, 1, 1, 3)$ at Level 1 is connected to the inputs of switches
  $(3, 1, 1, 3, 0) = 860$, $(3, 1, 1, 3, 1) = 861$, $(3, 1, 1, 3, 2) = 862$, $(3, 1, 1, 3, 3) = 863$
  at Level 2.

6. There are P^{i+1} inputs to the P^i switches at level 0.

— The inputs are labelled $M = (m_i, m_{i-1,} \cdots, m_0)$

- Where m_r are integers base P,
- And $M = m_i P^i + m_{i-1} P^{i-1} + \cdots + m_0$

— Input M is connected to switch M mod P^i

- Input = $(m_i, m_{i-1,} \cdots, m_0)$ is connected to switch $(m_{i-1}, m_{i-2,} \cdots, m_0)$
- *We can connect M to any switch at level 0, as long as P inputs are connected to each input switch, but chose this pattern for convenience.*

7. There are P^{i+1} outputs from the P^i switches at level l.

— The outputs are labelled $N = (n_i, n_{i-1,} \cdots, n_0)$.

- Where $(n_i, n_{i-1,} \cdots, n_1)$ is the switch number.
- And n_0 is the output number of that switch.

— Output N at level l is connected to

- The n_i^{th} input of switch $(n_{i-1}, n_{i-1}, \cdots, n_0)$
- *This convention is required to make the switch self-routing.*

5.2.1.2 Routing through the switch

- The switch is self-routing, the number of the destination is used to select the next PxP switch on the path at each stage.

1. The input is $M = (m_i, m_{i-1,} \cdots, m_0)$
   The destination is $N = (n_i, n_{i-1}, \cdots, n_0)$.
2. At each stage, k, select the switch ouput n_{i-k}

   — The switching procedure operates by shifting the integers that represent M out of the switch position while shifting in the integers that represent N

   A. The input arrives at stage 0 at switch $M = (m_{i-1}, m_{i-2,} \cdots, m_0)$

   B. It is switched to output n_i,
   which is connected to switch $(m_{i-2}, m_{i-3}, \cdots, m_0, n_i)$ at level 1

   C. At level 1 it is switched to output n_{i-1},
   which is connected to switch $(m_{i-3}, \cdots, m_{0,} n_i, n_{i-1})$ at level 2.

   D. At stage i-1, the input is at switch $= (m_0, n_i, n_{i-1}, \cdots, n_2)$.
   It is switched to output n_1,
   Which is connected to switch $(n_i, n_{i-1}, \cdots, n_1)$ at level i

   E. At level i we select output n_0,
   which is conneced to N

3. We get to any P^{i+1} output from any of P^{i+1}inputs in $i + 1 = \log_P P^{i+1}$ stages

   This is the smallest number of stages possible with PxP switches

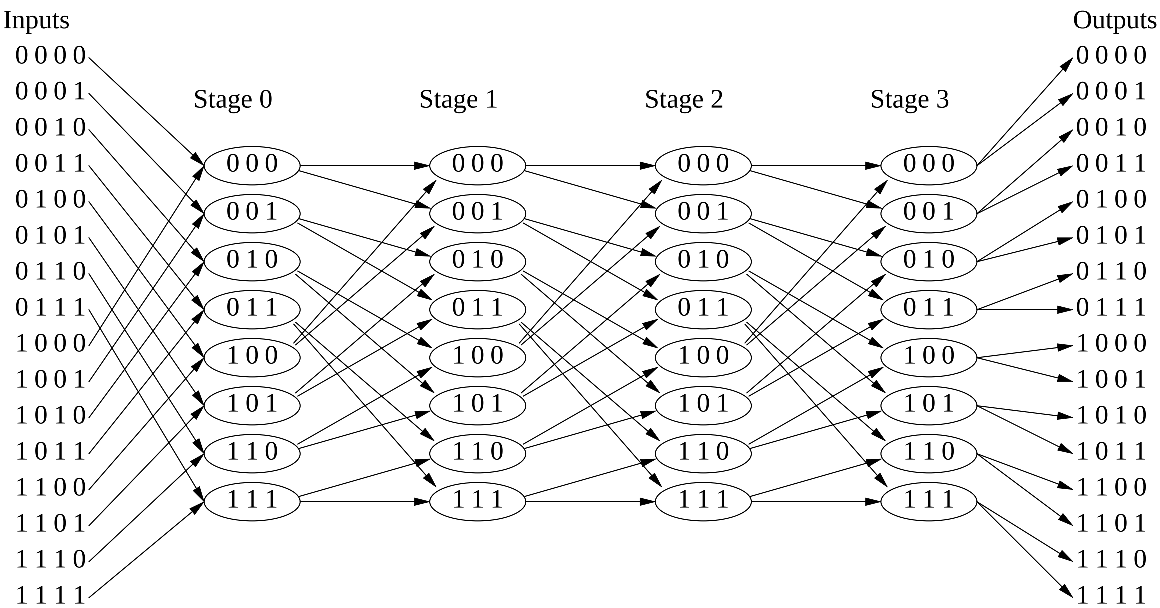
— At each switch we can get to at most P outputs.

— Therefore, in $i+1$ stages we can get to at most P^{i+1} different outputs from any starting position.

— There are no alternate paths through the switch.
If there is more than one way we can reach an output, then we cannot reach the maximum number of outputs.

5.2.1.3 SX in ATM networks

In ATM networks we have always used 2x2 switches, $P = 2$
Example of 16X16 ATM switch.



5.2.1.4 Buffering in Slotted System

— 2 inputs in a switch may be directed to the same output stages 0, 1, , $\cdots$, $i-1$ by the self routing rule, even if the inputs are headed for different destinations.

— Both inputs cannot be forwarded in the slot time

— We can:

1. Discard one of the inputs.
   It is lost.

2. Insert buffers at the contention points and forward the outputs in a later slot.

   — This reduces the loss rate, but the buffers may overflow.

3. Operate the switch fabric at a higher rate than the slot arrivals.

   — If we operate the switch fabric at twice the line rate, and 2 slots that are directed to different outputs contend at stage 2, switch 010, both may be forwarded to their outputs and transmitted in the same transmission slot time.

4. Two input slots that are destined to the same output cannot be transmitted at the same time, even if they make it through the fabric

   — The slots may be stored at the output of the switch, or blocked and stored at the input.

— Switches have been designed with both input and output buffers.

— Generally, fewer output buffers are required for the same loss probability

— Some simple designs have been constructed without intermediate buffers.

— We switch fixed size cells.

- This allows us to switch different inputs onto the output lines without storing them.
- Variable size packets cannot be switched onto the output line without storage, to account for the different start and stop times of the packets.

— Shuffle-exchange switches are considered space division switches because several packets can go through the fabric simultaneously.

— In the mid 90’s, after the introduction of Web browsers, the Internet started growing rapidly.

- It was thought that the store-and-forward switching techniques in routers would not keep up with the demand, and would be replaced by space division ATM switches.
- This didn’t happen. Most current routers divide variable size packets into fixed size cells, and use space division switches.

Home work

Non-binary shuffle-exchange switch

A. Draw a 27x27 switch with P=3

B. Find the path from input M=17 to output N=13

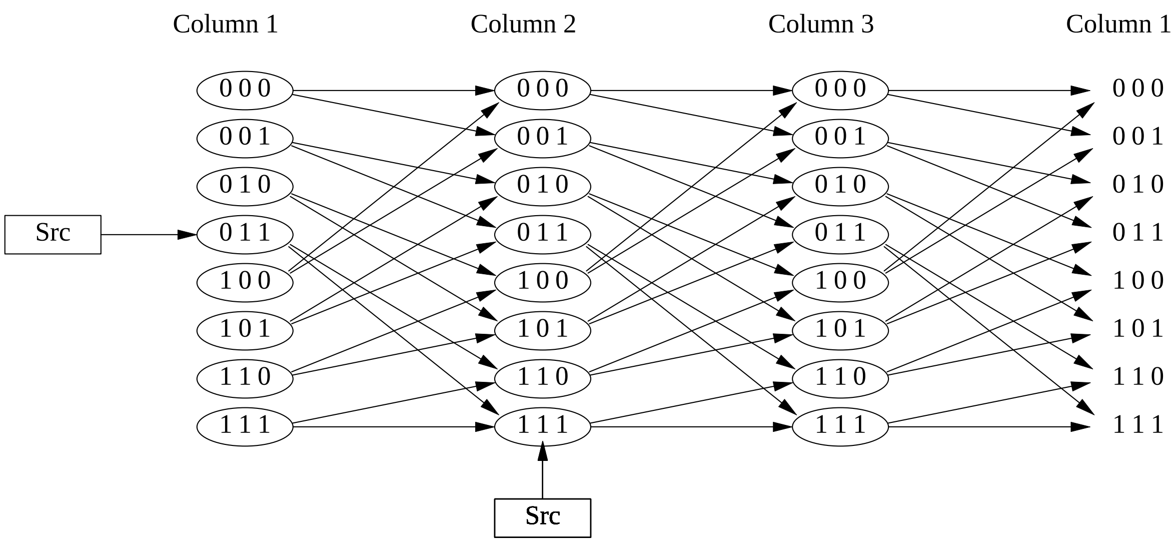
C. Draw a tree from input 7 to all outputs.
Note that this is the largest number of outputs that can be reached from this input in 3 steps.

5.2.2 Shuffle-Exchange Networks

How do we turn the shuffle exchange switch into a network?

1. Connect sources at each of the 2x2 switching elements, as in the MSN
   Use the same access rule as in the MSN
2. Connect the outputs of the switch to the inputs to make a cyclotomic structure

References [9, 11, 12]



This network has 24 nodes/sources.

The 7 nodes in the same column are reached in 3 steps.

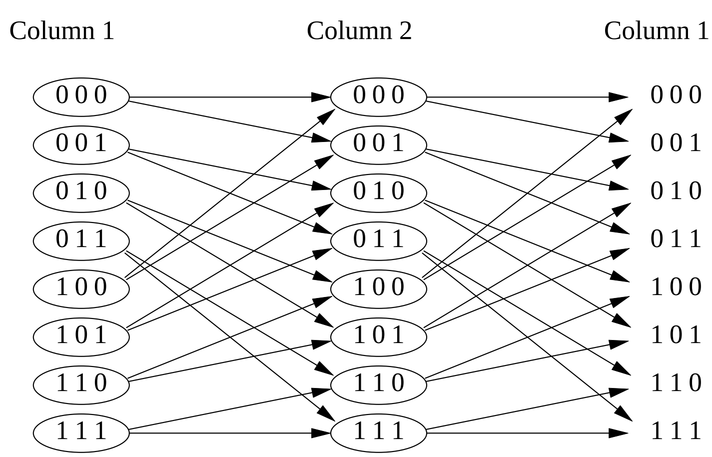
2 of the 8 nodes in the next column are reached in 1 step, the other 6 are reached in 4 steps

4 of the 8 nodes that are 2 columns away are reached in 2 steps, the other 4 are reached in 5 steps.

The average distance between nodes is $\frac{7*3+2*1+6*4+4*2+4*5}{23} = 3.26$

In A SX networks, there must be 2^i nodes in a column to preserve the switching rule, but there doesn't need to be i columns, as in the switch.

For instance, we can construct a 16 node network with 8 rows and 2 columns.



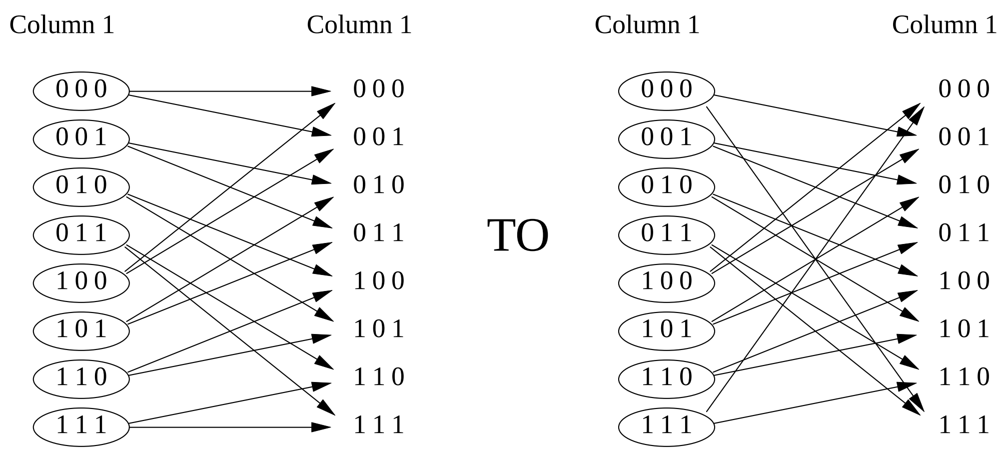
We could also construct a 32 node network with 8 rows and 4 columns.

When we reduce the number of columns to 1, node 0 and node $2^i - 1$ are connected to themselves.

These links are never used.

To avoid wasting the links, link $0 \rightarrow 0$ is replaced with $0 \rightarrow 2^i - 1$, and link $2^i - 1 \rightarrow 2^i - 1$, is replaced with $2^i - 1 \rightarrow 0$

The network is still 2-connected.



In general, the distance between nodes in a S-X network grows as $\log_2 N$, while the distance for the MSN grows as $\sqrt{N}$.
Therefore, the average distance for the S-X is smaller for large numbers of nodes.

In general, when we reduce the average distance, we reduce the number of alternative paths, and increase the distance when there are deflections.

We cannot modify the S-X structure to add single nodes, and extensive re-wiring is needed to add new rows or columns

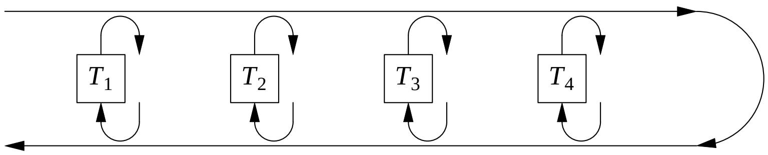
Home work

Connect 18 nodes as 2 columns of 9 nodes in a shuffle net with P=3, and determine the characteristics

6. Mapping Logical Topology onto a Physical Topologies

In a multiple wavelength network the objective is to construct a forwading topology that can forward messages between all nodes without transmitting and receiving on every wavelength at every node

Consider the physical topology



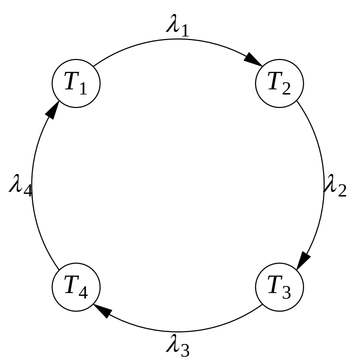
T_1 transmits on λ_1 and receives on λ_4

T_2 transmits on λ_2 and receives on λ_1

T_3 transmits on λ_3 and receives on λ_2

T_4 transmits on λ_4 and receives on λ_3

The logical topology is



- The average number of times that a packet is transmitted between a source and destination when all connections are equally likely is $\frac{1 \times 1 + 1 \times 2 + 1 \times 3}{3} = 2$
- Therefore, if an average of λ messages per second arrives at each source, an average of 2λ messages per second is transmitted on each wavelenght, as opposed to 4λ messages per second in a single wavelenght system
- In addition, there is no contention between sources, since each source transmits on a separate wavelenght, while in the single wavelength system all 4 stations contend with one another.

Now consider a network with 2 wavelengths, instead of 4:

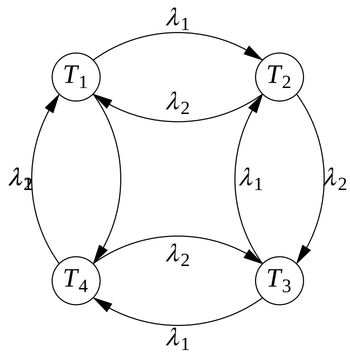
T_1 transmits on λ_1 and receives on λ_2

T_2 transmits on λ_2 and receives on λ_1

T_3 transmits on λ_1 and receives on λ_2

T_4 transmits on λ_2 and receives on λ_1

The logical topology is



- The average number of times that a packet is transmitted between a source and destination when all connections are equally likely is $\frac{2 \times 1 + 1 \times 2}{3} = 4/3$
- Therefore, if an average of λ messages per second arrives at each source, and average of $8/3\lambda$ messages per second is transmitted on each wavelenght, as opposed to 4λ messages per second in a single wavelenght system
- 2 sources contend on each wavelength, while in the single wavelength system all 4 stations contend with one another.

There is only 1 way to construct the logical network when there is one source transmitting on each wavelenght. A loop network.
However, when there are 2 or more sources transmitting on each wavelenght there are many ways to select the 2 sources. The different ways result in different logical networks.

We should select a logical network that:

A. Is completely connected.
Each source can reach each destination

B. Has an equal number of transmissions on each wavelength
So that the contention on each wavelength is equal

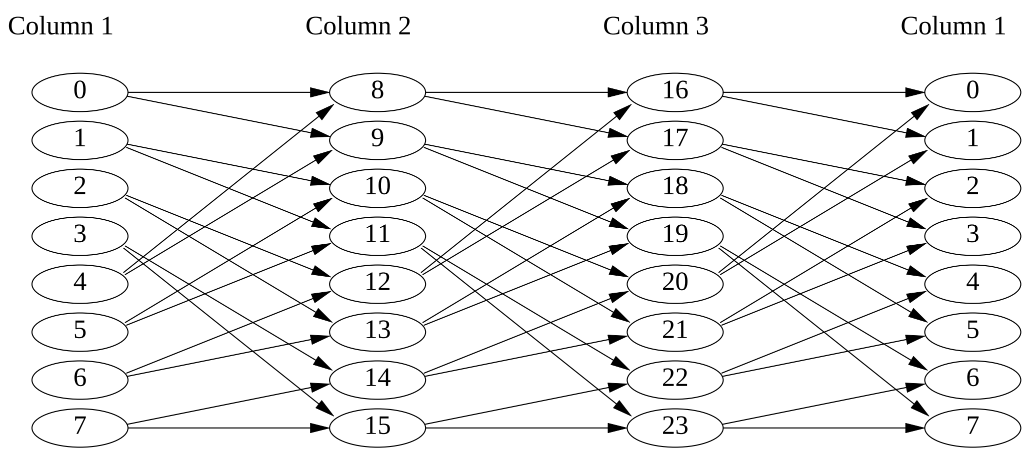
C. Has the minimum average distance between sources
So that the number of times each message is forwarded, and the number of transmissions per wavelength, is minimized

The shuffle-exchange networks have a small average distance between sources.
They can be arranged so that there are the same number of sources per wavelength.
For uniform traffic the load on each wavelength is the same.

Example: Logical network with 24 nodes arranged in 3 columns of 8 nodes, connected as a shuffle exchange network:

2 nodes transmit on each wavelenght.
0 and 4 on λ_1, 1 and 5 on λ_2, 2 and 6 on λ_3, 3 and 7 on λ_4,
8 and 12 on λ_5, 9 and 13 on λ_6, 10 and 14 on λ_7, 11 and 15 on λ_8,
16 and 20 on λ_9, 17 and 21 on λ_{10}, 18 and 22 on λ_{11}, 19 and 23 on λ_{12},

In order to obtain this logical topology the following nodes must receive on wavelengths:
0 and 1 on λ_9, 2 and 3 on λ_{10}, 4 and 5 on λ_{11}, 6 and 7 on λ_{12},
8 and 9 on λ_1, 10 and 11 on λ_2, 12 and 13 on λ_3, 14 and 15 on λ_4,
16 and 17 on λ_5, 18 and 19 on λ_6, 20 and 21 on λ_7, 22 and 23 on λ_8,



- The average number of times that a packet is transmitted between a source and destination when all connections are equally likely is $\frac{2 \times 1 + 4 \times 2 + 7 \times 3 + 6 * 4 + 4 * 5}{23} = 3.26$
- Therefore, if an average of λ messages per second arrives at each source, and average of 6.52λ messages per second is transmitted on each wavelenght, as opposed to 24λ messages per second in a single wavelenght system

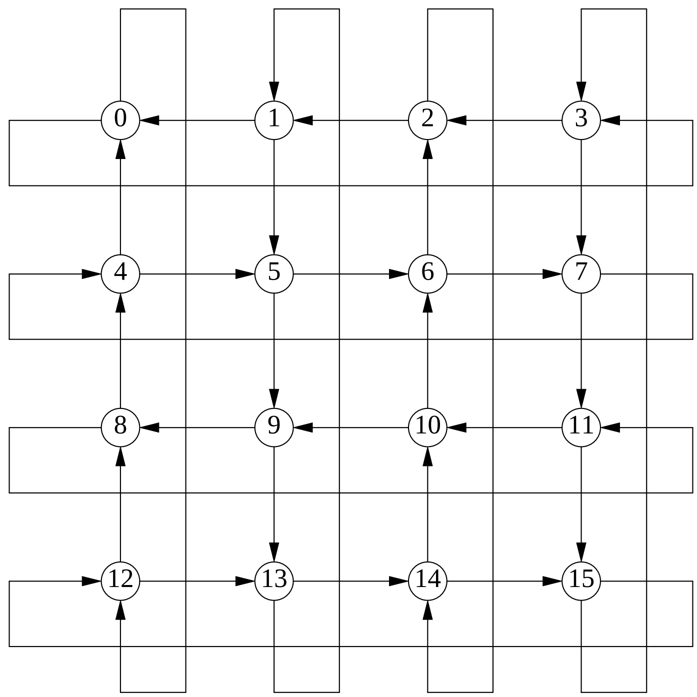
We can reduce the number of wavelengths by using a P connected shuffle-exchange network.

Home work

Mapping a Logical Topology onto a Physical Topology

The physical topology is a "D" network with 16 nodes, each node transmits on one wavelength and receives on one wavelength.

A. Pick the wavelenghts to construct a multihop 4x4 Manhattan Street Network (MSN), with two sources interfering on each wavelength.
(List the wavelengths each node transmits and reveives on)



B. If the "D" network has a head-end that sends slot timing on each of the wavelengths. How can we operate the MSN as a contention free slotted network?

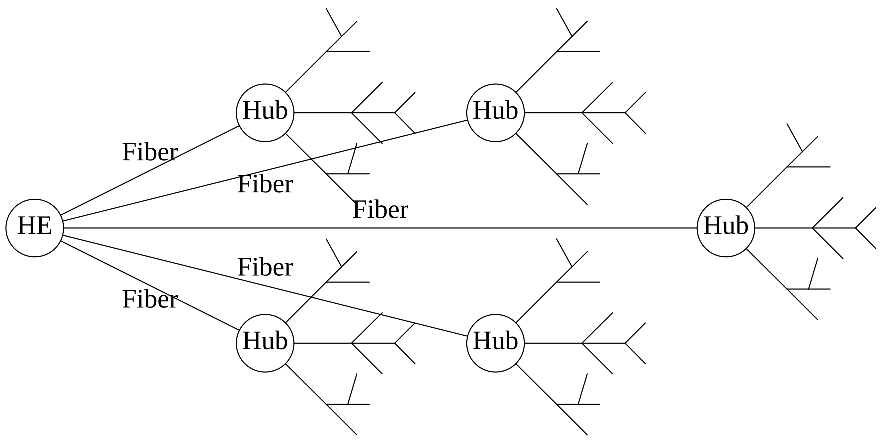
C. If we only have 4 wavelengths can we construct a 4x4 contention free MSN? How?

7. Tree Networks

CATV - Homenet [13]
Power distribution networks are being used to collect meter readings

Topology and Transmission

- CATV networks were originally designed to transmit TV programming from a single head-end to a large number of homes.
  The original networks were designed as trees rooted at the head-end.
- CATV networks are evolving into a hub and spoke topology.



Hub and Spoke Topology

— Fibers distribute the signals to the hub.

— Small trees are rooted at the hubs that deliver the signal to the homes using coaxial cables.

— The hub and spoke topology has a higher signal to noise ratio and is easier to maintain.

- The bandwidth in the CATV cable is divided into frequency bands

— Most of the bands are used to transmit signals downstream from the head end to the homes.

— A small number of bands are reserved to transmit signals upstream, from the homes to the head end.

- CATV taps resemble the passive taps in a fiber optic network,
  - — but typically remove a much smaller fraction of the signal energy.
  - — this is possible because electronic amplifiers are much less expensive than optical amplifiers
- Noise
  - — Because of the directional taps, noise inserted by poorly terminated connectors in a home flows upstream, and is not received at the other homes.
    The noise inserted at a home does not affect the S/N ratio of the TV signal delivered to the other homes.
  - — The signal to noise ratio on the upstream channel is not high enough to supply TV quality pictures, but is adequate to transmit data and digital voice.

- Power distribution networks have a tree topology.
  - — Instead of hubs, transformers reduce transmission capacity at some nodes.
  - — Interest in communications on a smart grid is renewing interest in the tree topology.

Data and Voice Transmission

- Data transmission uses a 2 channel variant of CSMA/CD

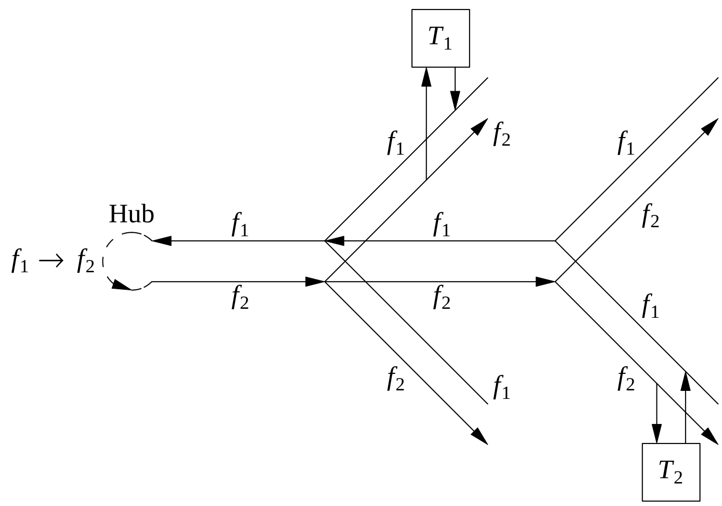
— Terminals transmit on f_1 toward the head-end or hub

— At the hub receive f_1 and re-transmit the signal on f_2

— The terminals receive the signal from the other terminals and perform CSMA/CD on f_2

— The maximum propagation delay in the network is the 2-way propagation delay from a terminal to the hub and back

— CSMA/CD requires that the β of the network is $\ll 1$



2 Channel CSMA/CD

— Two channel CSMA/CD uses standard Ethernet chips

— A 2 frequency network can also be used in cellular radio where there are "hidden nodes" and reflections prevent CD
The base station retransmits the signal [14]

- At present analog voice is placed on the CATV network by subdividing a 6 Mhz TV channel into 3 khz voice channels

— An early experimental system inserted voice in the data channel using MSTDM

- Problems are encountered when using CSMA/CD on CATV networks

1. There are fewer upstream channels than downstream channels in a CATV network.
2. The number of data users on a tree may require more bandwidth than is available.
3. The round trip delays from the end-points of the tree make CSMA/CD inefficient.

- Hubs can be routers

— Each tree that terminates at a Hub is a separate Ethernet and uses the same downstream frequency f_1

— The Hub receives data on the fiber that is destined for the terminals and uses the CSMA/CD protocol to contend with the signals from the tree to send data to the terminals.

— The number of data users is increased by my increasing the number of hubs, and moving the hubs closer to the end-users

- Homenets: A frequency reuse plan was used in CATV networks that consisted of a single tree.
  Homenets partitioned the tree into smaller trees, with their own local headends, dedicated lines from the local head-ends to the network head-end, and required terminals to switch between different receive frequencies.
  Homenets aren't needed in the current CATV topologies.

REFERENCES

[1] D. Bertsekas, R. G. Gallager, **Data Networks**, Prentice-Hall Inc, 1992.

[2] R. Ramaswami, K. N. Sivarajan, "Optical Networks: A Practical Perspective," Morgan-Kaufman Publishers, 2002, 2nd edition.

[3] L. Fratta, F. Borgonovo, F. A. Tobagi, "The Express-Net: A local area communication network integrating voice and data," Proc. Int. Conf. Perf. Data Commun. Syst., Paris, Sept. 1981, pp. 77-88.

[4] J. O. Limb, C. Flores, "Description of Fasnet - A unidirectional local area communications network," BSTJ, vol. 61, no. 7, Sept. 1982, pp. 1413-1440.

[5] N. F. Maxemchuk, "Twelve Random Access Strategies for Fiber-Optic Networks," IEEE Trans. on Commun., vol. com-36, Aug. 1988, pp. 942-950.

[6] E. L. Hahne, A. K. Choudhury, N. F. Maxemchuk, "Improving the Fairness of Distributed-Queue Dual-Bus Networks," INFOCOM ’90, San Francisco, June 5-7, 1990, pp. 175-184.

[7] E. L. Hahne, A. K. Choudhury, N. F. Maxemchuk, "DQDB Networks With and Without Bandwidth Balancing," IEEE Transactions on Communications, Vol. 40, No. 7, July 1992, pp 1192-1204.

[8] N. F. Maxemchuk, "Routing in the Manhattan Street Network," IEEE Trans. on Commun., May 1987, vol. COM-35, no. 5, pp. 503-512.

[9] N. F. Maxemchuk, "Regular Mesh Topologies in Local and Metropolitan Area Networks," AT&T Technical Journal, Sept. 85, vol. 64, no. 7, pp. 1659-1686.

[10] N. F. Maxemchuk, "The Manhattan Street Network," Proc. of Globcom ’85, pp. 255-261, Dec. 2-5, 1986, New Orleans, LA.

[11] M. Hluchyj, M. Karol, "ShuffleNet: an application of generalized perfect shuffles to multihop lightwave networks," INFOCOM ’88, , 27-31 March 1988, pp. 379 -390.

[12] N. F. Maxemchuk, "Comparison of Deflection and Store-and-Forward Techniques in the Manhattan Street and Shuffle-Exchange Networks," IEEE INFOCOM ’89, Ottawa, Ont. Canada, April 25-27, 1989, pp. 800-809.

[13] N. F. Maxemchuk, A. N. Netravali, "Voice and Data on a CATV Network," IEEE J. on Sel. Areas in Commun., vol SAC-3, No. 2, Mar. 1985 pp. 300-311.

[14] N. F. Maxemchuk, "Applying MSTDM to Packet Voice and Data in Cellular Radio Systems," ICNP ’98, Austin Tx, Oct 14-16, 1998, pp. 309-318.