Course on

Computer networks

Medium access sublayer

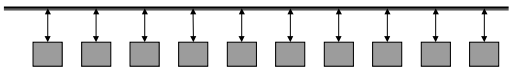
Henk Corporaal

Leiden University
2000

Medium access sublayer

A number of systems are connected to share medium:

- Who gets access?
- Used for LANs
- Addressing
- Broadcast possible



Note: WANs use point-to-point (except for satellite channels)

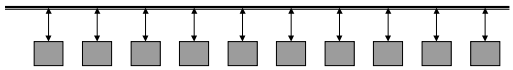
Medium Access Sublayer

Overview

- Channel allocation problem
  - static
  - dynamic
- Multiple access protocols
  - how to divide the line among multiple users
- IEEE standards for LANs and MANs
- Bridges
  - between different LANs
- High-speed LANs
- Satellite networks

Channel Allocation problem

Single channel has to be shared among multiple users



Solutions:

- **static** allocation
  - user gets a channel for the whole 'conversation'
- **dynamic** allocation
  - channel capacity allocated for a user is not fixed; depends on usage

Static channel allocation

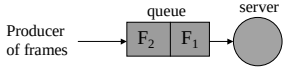
Using FDM or TDM to divide channel capacity among N users

- Inefficient bandwidth use:
  - some users may be idle
- Users may be denied access even when other users do not use there allocated capacity
- Long time service delay T

Note that this is not unrealistic; most traffic has burst nature

Static Allocation

Queueing theory of single server: *illustrating service delay for subdivided channel*



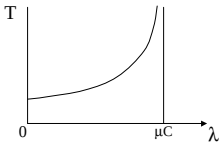
Consider one channel with:

- capacity C bps
- frame rate λ frames/sec
- frame length $L_{average} = 1/\mu$ bits (exponential or normal distribution: $f(L) = \alpha.e^{-\beta(L-\gamma)^2}$, average = γ)
- frame time = $L/C = 1/\mu C \Rightarrow$ max frame rate = μC

Static Allocation

Service time:

$$\Rightarrow T = \frac{1}{\mu C - \lambda}$$



Dividing channel into n subchannels:

$$T_{div} = \frac{1}{\mu \frac{C}{n} - \frac{\lambda}{n}} = nT$$

So service time gets worse !

Dynamic channel Allocation

Assumptions:

1. Station model:
   N independent stations generating frames within interval Δt with probability = $\lambda \Delta t$
   λ is frame rate (frames/second)

2. Single communication channel

3. Collision occurs when 2 frames are transmitted simultaneously

Dynamic Cannel Allocation

4.Time is
   a) continuous, or
   b) slotted (discrete intervals)

5. Distinguish *'carrier sense'* and *'no carrier sense'*

Multiple Access Protocols

Different protocols:

- ALOHA : pure and slotted
- CSMA : carrier sense multiple access protocols
- Collision free protocols
- Limited contention protocols
- Wireless LAN protocols
- Cellular Radio

ALOHA

Station transmits as soon as if it has a frame

Efficiency ?

Assume:

- N frames/sec generated; channels C bps : average frame length L bits
- frame-time T_{frame} = L/C seconds
- S frames / frame-time generated (utilisation of channel)
  $\Rightarrow S = N \times C / L \ (< 1)$
- G frames / frame-time transmitted; **due to collision** G > S
- N, S and G are means of *Poisson distribution*

Poisson distribution

Poisson distribution is derived from the binomial distribution:

- If event happens with prob. p, the prob. that the event happens exactly X times in N trials is:

$$p(X) = \binom{X}{N} p^X (1-p)^{N-X} = \frac{N!}{X!(N-X)!} p^X (1-p)^{N-X}$$

If N large and p not close to 0 or 1 we get a Normal distribution
If N large and p small (rare event) we get a Poisson distribution

ALOHA

Probability of k frames generated / frame-time is

$$\Pr[k] = \frac{G^k e^{-G}}{k!}$$ (Poisson distribution)

$\Rightarrow \Pr[0] = e^{-G}$ (prob. **no** frame generated in one frame-time)

No collision if no other frame generated in two frame-times (fig 4-2)

$P_0 = Pr[0]*Pr[0] = e^{-2G}$

$S = G.P_0 = e^{-2G}$ Result in fig 4-3

Slotted Aloha

Divide time in discrete intervals

Users have to agree on slot boundaries

No Collision probability becomes: $P_o = e^{-G}$

Therefore throughput becomes : $S = G\, e^{-G}$

Maximal utilisation of channel (see fig 4-3):

- Pure Aloha : 18.4%
- Slotted Aloha : 36.8% (37% empty slots, 26% collisions)

Carrier Sense Protocols

The number of collisions can be reduced by sensing the carrier before transmission

Variants:

- 1-persistent CSMA (Carrier Sense Multiple Access)
- non-persistent CSMA
- p-persistent CSMA
- CSMA with collision detect (CSMA / CD)

CSMA

1-persistent

- listen to channel and transmit as soon as channel becomes idle (greedy approach)
- if collision, wait random time, and try again

Note: Long propagation delays must be avoided (if not, a greater chance exist that, even after one station starts transmission, another one senses idle, and starts as well)

This approach is too greedy

(*name 1-persistent: if idle transmit with p=1*)

CSMA

Less greedy approaches

Non-persistent:

- Sense channel; if idle, transmit
- If not idle, wait random period
- Better utilisation, however, longer delays

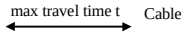
P-persistent:

- Sense channel, if idle, transmit with probability p
- With probability 1-p defer to next slot
- If next slot idle repeat same algorithm

Performance comparison: see fig 4-4

CSMA with Collision Detection

- Widely used on LANs in MAC sublayer (IEEE 802.3; **ethernet**)
- Idea: *stop directly after detecting collision and wait random period*
- How long does it take to detect collision?



- It takes 2t to be sure (contention period, see fig. 4-5) (= 10 micro sec on 1 km cable)

Collision free protocols

- Although CSMA/CD avoids collisions once a station has seized the channel; collisions still occur during the contention period
- Contention period may become large for long (fiber) cables
- Try avoiding all collisions !

Collision free protocols

1. Bit-Map protocol (reserve bandwidth)
   In contention period each station broadcasts if it has a frame to send (see fig 4-6)
2. Binary countdown
   Variant of bit-map with a more efficient (binary) station number encoding (see fig 4-7)

Disadvantage: long delay (especially at low load)

Limited contention protocols

So far:

- Contention protocols: low delay at low load
- Collision-free protocols: high channel utilization at high load

Combine good properties using adaptive algorithms

Limited contention protocols

Consider k stations; each has probability p of transmitting a frame in a certain slot

Success probability (of arbitrary station):

$$\Pr = kp(1-p)^{k-1}$$

$\Rightarrow$ p optimal: 1/k

Optimal success probability becomes (see fig. 4.8):

$$\Pr[optimal] = (\frac{k-1}{k})^{k-1}$$

Limited contention protocols

- Lesson: reduce competition for higher k
  - not all stations should try in the same slot

How to assign stations to slots?

- Dynamic assignment:
  - many stations may try / slot if load low
  - few stations may try / slot if load is high

Limited contention protocols

Adaptive tree walk protocol (see fig. 4-9);

a depth-first traversal of a tree:

- In slot 0 all stations may try
- If collision, all stations under node 2 may try in slot 1
- Three cases
  - one station succeeds: give next frame slot to node 3
  - collision: continue with node 4
  - no one sends: continue with node 6 !!

Wireless LAN protocols

Wireless protocols are *inherently multiple access* !
Can we use ***sensing*** ?

Transmitter has finite range !
Advantage:
- multiple connections can be established (using cells)

Disadvantage:
- transmitter can not decide if collision occurred ; competition is too far away (see fig 4-11)

Wireless LAN protocols

MACA: ***Multiple Access with Collision Avoidance***
(see fig. 4-12) (used as basis for 802.11 standard)
- Sender: sends RTS (request to send)
- Receiver: sends CTS (clear to send)
- RTS and CTS contain length of message to be sent

Key observations:
- Any station hearing RTS must keep silent during CTS; they may however send concurrently with the coming message
- Any station hearing CTS must keep silent during next message

Digital Cellular Radio: GSM

GSM: Global system for mobile communication

Combination of FDM, TDM and slotted Aloha
(fig. 4-13)
- FDM: divide freq. Spectrum into 124 duplex channels (bands) of 200 kHz each
- TDM: multiplex 8 time slots (frames) per channel
- Together this gives 8*124 = 992 channels
  - many of them are however not available because of conflicts with neighbor cells

Slot: 148 bit sent in 547 microsec (fig. 4-14)

Digital Cellular Radio: GSM

Three Control channels are used:

1. Broadcast channel for:
   - outputting identity of base station
   - each mobile checks its signal strength to see if it moved to other cell
2. Dedicated channel for:
   - location updating,
   - registration,
   - call setup

Digital Cellular Radio: GSM

3. Common Control channel for:
   - paging (of incoming calls); each mobile monitors this channel
   - random access channel, using slotted Aloha to request a slot on the dedicated channel, to set up a new call
   - access grant channel

Setting up a call:
   - mobile requests a dedicated access channel# using random access channel; this is the only channel where we can get collisions; so it is used very shortly
   - base grants channel# on access grant channel
   - setup the call on dedicated channel
   - start the conversation on one of the agreed regular channels

Digital Cellular Radio

- We will now look at a completely different way to allocate multiple users (channels) on a single line

- Compare:
  - time division: TDM
    - each speaker gets its turn
  - frequency division: FDM
    - concurrent speakers separated far enough
  - Now: code division
    - every speaker speaks a different language ***at the same time***

Digital Cellular Radio

CDMA: Code Division Multiple Access (fig. 4-16)

Key idea: each station uses whole frequency band, but speaks in its own language (spread spectrum)

Language: use ***chip sequence*** S, e.g. 1100.1010
(in practice hundreds of bits may be used)

- transmit 1: 1100.1010 (= S)
- transmit 0: 0011.0101 (= not S, or $\overline{S}$)

Digital Cellular Radio

- for the following theory it's easier to assume that we transmit 1 as: 1 1 -1 -1 1 -1 1 -1
  and 0 as: -1 -1 1 1 -1 1 -1 1

Each station has unique and orthogonal chip; for 2 m-vector chips S and T holds:

$$S \bullet T = \frac{1}{m}\sum_{i=1}^{m} S_i T_i = 0$$

Properties:

$$S \bullet S = 1$$
$$S \bullet \overline{T} = 0$$
$$S \bullet \overline{S} = -1$$

Digital Cellular Radio

Practical problems:

- Keys should be known in advance
- Senders should be received with same power
  - send with inverse power of received signal strength
- Synchronization: what if senders are not synchronized:
  - 1. Lock into the 'wanted' channel
  - 2. Other channels will be received as background noise

IEEE standard 802 for LANs

Set of standards for LAN broadcasting networks:

- 802.1 Introduction and Network interface specification
- 802.2 LLC: logic link control protocol (upper part of data link layer); common to all protocols
- 802.3 Ethernet: CSMA/CD (Xerox)
- 802.4 Token bus (GM)
- 802.5 Token ring (IBM)

802.3 Ethernet

Whole family of 1-persistent CSMA/CD protocols

- 1-10 Mbps
- Various media
  - first used on 50 ohm coaxial cable

Started as ALOHA system on Hawaiian Islands

Carrier sensing was added by Xerox

Cabling options: see fig. 4-17

Connecting a station: see fig. 4-18

Wiring a building: see fig. 4-19

802.3 Encoding

Needed: unambiguously determine start, end and middle of each bit without reference to external clock

Two encodings (fig. 4-20)

- Manchester encoding
  - sending 1: 1-0 transition
  - sending 0: 0-1 transition
  - Every bit has transition in the middle
  - However, twice the bandwidth capacity needed !
- Differential encoding

802.3 uses Manchester, +/- 0.85 volt signalling

802.3 Framing

The frame format is shown in fig. 4-21

- Preamble: 7 x 1010.1010 pattern
  - creates 10 MHz square wave of 5.6 microsec for synchronization
- Start field: 1010.1011 denotes start of frame

802.3 Addressing

- Addressing: 6 bytes/ address

Bit 47	46	45…0	type of address
0	0		ordinary local
0	1		ordinary global
1	0		group local
1	1		group global
1	1	1…..1	broadcast

- Global address: unique address assigned by IEEE
- Group: used for multicast

802.3 Framing

- Length field: 0-1500 bytes
- Data: payload
- Pad: ensure minimum frame length = 64 bytes
  - needed for collision detection (see fig. 4-22)
  - LAN, 2500 meter, 10 Mbps, 4 repeaters: 51.2 microsec minimum frame length => 64 bytes
  - Problem occurs when networks get faster (choose longer min. length or restrict network size)
- Checksum: CRC of 4 bytes

Binary Exponential Backoff

- After collision wait 0 or more slots (of 51.2 μsec)
- First collision: wait 0 or 1 slot (choose randomly)
- Second collision: wait 0, 1, 2 or 3 slots
- N-th collision: wait 0 - 2^N-1 slots
- N max = 10 => wait between 0 - 1023 slots
- Give up after 16 trials and leave recovery to higher layers

Algorithm ensures low latency at low load and fairly resolves collisions

802.3 Performance

- Assume constant transmission probability p
  - (no exponential backoff)
- Probability A of acquiring a slot (by one of k stations):
$$A = kp(1-p)^{k-1}$$
- Prob. that contention interval, W, has precisely j slots: $A(1-A)^{j-1}$
- <#slots per contention> = $\sum_{j=0}^{\infty} jA(1-A)^{j-1} = \frac{1}{A}$
- $\Rightarrow$ W = 2t/A (each slot takes time 2t)

802.3 Performance

p optimal (= 1/k) $\Rightarrow$

- A = $(1\text{-}1/k)^{k-1}$ =1/e (for $k \rightarrow \infty$, heavy load) $\Rightarrow$
- W = 2t.e = 5.4t

Channel utilization = t_{frame} / (t_{frame} + 2t/A)

Thus: longer cable gives lower utilization

Ethernet: 10 Mbps; 2.5 km cable, t = 51.2 microsec
Channel utilization is plotted in fig 4-23

802.3 Performance example

- 10 Mbps
- 64 byte = 512 bit slot time ⇒ t_{frame} = 512/10M = 51.2 μsec
  - k=1 ⇒ A = 1 ⇒ Utilization = 0.5
  - k=16 ⇒ A = 0.38 ⇒ Utilization = 0.275
- 1024 byte frame ⇒ t_{frame} = 819.2 μsec
  - k=1 ⇒ A = 1 ⇒ Utilization = 0.941
  - k=16 ⇒ A = 0.38 ⇒ Utilization = 0.859

Switched 802.3 LANs

When the number of active stations goes up an 802.3 LAN easily saturates

Solution:

- Use 100 Mbps ethernet (or some other network)
  - requires replacing lots of wires and hardware
- Use switch (fig 4-24)
  - switch uses high speed (> 1 Gbps) backplane
  - each card or connector forms separate collision domain

802.4 Token bus

Problems of 802.3:

- no priority support
- not real-time (unbounded delays possible)

Solution: *give each station a separate slot* => token bus (fig 4-25)

Logically stations form a ring

Stations can be added dynamically

Protocol:

- station which occupies token may send
- thereafter it gives token to next logical station

Very complex protocol; not used much

802.5 Token Ring

Based on physical ring (in contrast to 803.4; see fig 4-28)

Advantage: easy point-to-point hardware

Protocol steps:

- Sender takes token and puts interface in ***'transmit mode'***
- After sending final bit, token is regenerated (1 bit flip in header)
- Sender removes incoming bits from ring; may check if they are not corrupted
- One of the next stations can pick up the token

802.5 Token Ring

Notes:

- ring must be long enough (in bit times) to contain at least a complete token!
  => artificial delays may be inserted (especially when some stations are not connected)
- if one cable or system breaks down ring unusable
  - solution: use wire center (see fig. 4-29)

Comparison 802.x

802.3 compared to 802.4 & 802.5

- widely used
- simple protocol, no centralized control
- low latency when load is low
- stations can be added very easily
- non-deterministic delay ⇒ not real-time
- limited cable length
- no priorities
- analog component

802.4: complex protocol

802.5: point-to-point connections easy, fully digital

802.6 Distributed Queue Dual Bus

MAN, 2 bus, one for upstream and one for downstream traffic:

- 53 byte cells with 44 byte payload are send constantly
  - cell contains 2 protocol bits
  - Request : set when stations wants to make a request
  - Busy : cell is occupied
- Protocol is not greedy; requests are handled in a kind of FIFO mode
- Each host maintains 2 counters:
  - RC : request counter
  - CD : represents position in the waiting queue
  - see fig. 4-32 for the operation

Logical Link Control: 802.2

- Data link layer for 802 protocols consists of two parts (see fig. 4-33):
  - LLC logical link control
  - MAC multiple access control
- LLC implements the functionality as discussed in chapter 3 (if needed), like
  - error control
  - flow control
- It offers three services:
  - unreliable datagram service
  - acknowledged datagram service
  - reliable connection-oriented service

Bridges

Used to connect multiple LANs (fig 4-34)

Operate in Data Link Layer => no knowledge of network layer header => can copy IP, IPX or OSI packets equally well (in contrast to a router)

Why multiple LANs in one company?

- Multiple departments and buildings
- Connect different LANs, eg. 802.3 with 802.5 (fig 4-35)
- LAN distance restriction (max. cable size)
- Reliability
- Recurity (do not forward everything)

Bridges from 802.x to 802.y

Common problems to handle by a bridge (compare different 802 frame formats in fig. 4-36):

- reformatting frame
- recalculate checksum
- bit rate conversion
- frame length may be too large (discard frame?)

Specific problems: see fig. 4-37

Transparent Bridges

Can be inserted anywhere (no hardware/software changes required)

Forwards packets if needed

Backward learning:

- keep table of (destination address + outgoing LAN-link) entries
- update table every time a packet comes in (using the source address)
- remove very old entries (to trace systems going down and other topology changes)

Transparent Bridges

Routing procedure of a bridge:

- discard frame if source and destination LAN are equal
- forward if they are different
- use flooding otherwise (also for broadcast frames)

Redundancy: use multiple bridges between LANs

=> ***loop problem*** (fig. 4-39)

Solution to loop problem: spanning tree bridges (fig. 4-40); build spanning tree using shortest path from root algorithm.

Fast Ethernet

Based on 802.3: called 802.3u

100 Mbps bandwidth

10 Base-T wiring (star twisted pair) needed, using hubs (no T-connectors allowed)

- avoiding the collision detection problem

Three wire types, see fig 4-47

Note: category 3 twisted pair: 4 pairs required

1 down / 3 up ; ternary signalling on 3-up pairs allows 4 bits/signal x 25 MHz => 100 Mbps

Excercises

From chapter four:

- 1 - 4, 8, 13, 14, 18, 21