Chapter 4

The Medium Access Control Sublayer

LLC, MAC } DLL

MAC → Sublayer of DLL

The Channel Allocation Problem

→ Dedicated resource (channel for FDM, time for TDM) for a user

- Static Channel Allocation in LANs and MANs
- Dynamic Channel Allocation in LANs and MANs

→ No dedicated resource. Here, all frames can be assumed to be somehow magically arranged orderly in a big central queue.

Let, mean time delay → T, channel capacity C bps, arrival rate λ frames/s, frame length follows an exponential probability density function with mean $1/\mu$ ~~frames~~ bits/frame. ~~Now,~~ From queuing theory, it can be shown for Poisson arrival and service time, $T_{dynamic} = \frac{1}{\mu C - \lambda}$

Now, $1/\mu$ bits → 1 frame
1 bit → μ "
C " → $C\mu$ " service rate

Ex: $C = 100$ Mbps, $1/\mu = 10{,}000$ bits, $\lambda = 5000$ frames/s

Wrong ⇒ $T = \frac{1}{\mu C} = 100\,\mu sec$ ← holds only if no contention

Right ⇒ $T = \frac{1}{\mu C - \lambda} = 200\,\mu sec$

→ FDM

Let, N independent channels; so, capacity $= C/N$, mean input rate $= \lambda/N$

So, $T_{static} = \frac{1}{\mu(C/N) - (\lambda/N)} = \frac{N}{\mu C - \lambda} = N T_{dynamic}$

So, delay using FDM is N times worse than ~~static~~ dynamic case [similar logic holds for TDM]

Dynamic Channel Allocation in LANs and MANs

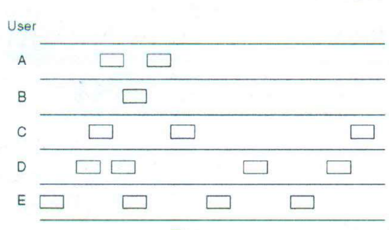
1. Station Model. → N independent stations/terminals
2. Single Channel Assumption. → A single channel is available
3. Collision Assumption. → Simultaneous tx of two frames makes a collission
4. (a) Continuous Time. → Frame tx can begin at any time
   (b) Slotted Time. → Time is divided into discrete intervals (slots). Frame tx begins always at the start of a slot.
5. (a) Carrier Sense. → Stat? assesses the channel for being idle before using it
   (b) No Carrier Sense. → No sensing

Multiple Access Protocols

- ALOHA < Pure, Slotted
- Carrier Sense Multiple Access Protocols < Persistent, Non persistent, CSMA with collision detecⁿ
- Collision-Free Protocols < Bitmap, Binary countdown
- Limited-Contention Protocols – Adaptive treewalk
- Wavelength Division Multiple Access Protocols
- Wireless LAN Protocols < MACA, MACAW

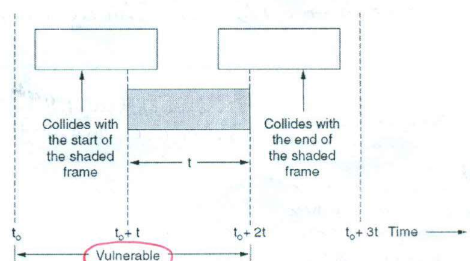
→ Norman Abramson (1970) ⇒ University of Hawaii

Pure ALOHA ⇒ let users transmit whenever they have data to be sent



In pure ALOHA, frames are transmitted at completely arbitrary times.

Pure ALOHA (2)



→ If any ~~po~~ frame gets generated within this time (2 frame time), then there will be a collission

Vulnerable period for the shaded frame.

G → mean # of attempts } per frame time $[G \geq N]$
N → mean # of generations }

N > 1 : Almost always collission;
0 < N < 1 : Reasonable throughput.

Low load: N ≈ 0; so, G ≈ N
High load: G > N

* Throughput = offered load * prob of a tx succeeding = G * P_0
* Prob that k frames are generated during a given frame time is given by the Poisson distribuⁿ: $Pr[k] = \frac{G^k e^{-G}}{k!}$

⇒ So, prob of 0 frame $= P_r[0] = e^{-G}$

Now, vulnerable time period is two frame long; so, here mean # of frames generated is 2G.

* So, prob of no frame during the vulnerable period is $P_0 = e^{-2G}$

So, throughput, $S = G \cdot P_0 = G \cdot e^{-2G}$

for max S, $G = 0.5$; $S = .5 * e^{-2 \cdot .5} = .184$

So, best channel utilizaⁿ = 18.4%.

for max S, $\frac{dS}{dG} = 0$; so, $e^{-2G} + G(-2)e^{-2G} = 0$

So, $1 - 2G = 0$; so, $G = 1/2 = 0.5$

Slotted ALOHA ⇒ Time is divided into some ~~discrete~~ intervals. A station is not permitted to send whenever a frame is generated, rather it is required to wait for the beginning of the next slot. So, now the vulnerable period becomes one frame slot.

So, $Pr(0) = \frac{G^0 . e^{-G}}{0!} = e^{-G}$; So, $S = G.P_0 = G.e^{-G}$

For max S, $\frac{dS}{dG} = 0$; So, $e^{-G} + (-1)Ge^{-G} = 0$; So, $1 - G = 0$; So, $G = 1$

So, $S_{max} = 1.e^{-1} = 0.368$; ⇒ So, for slotted ALOHA, the best we can hope for is

37% slots empty, 37% success, and 26% collission.
$P_0 = e^{-G}$ $S = G.e^{-G}$ $100 - 37 - 37 = 26$

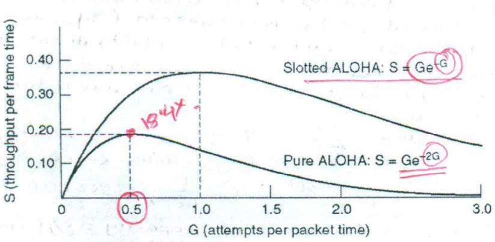
Now, prob of avoiding collision $= e^{-G}$; So, prob. of collision $= 1 - e^{-G}$

Therefore, prob of a tx requires exactly k attempts $= e^{-G}(1 - e^{-G})^{k-1}$

So, expected # of tx attempts ⇒ $E = \sum_{k=1}^{\infty} k P_k = \sum_{k=1}^{\infty} k e^{-G}(1 - e^{-G})^{k-1} = e^{G}$ ⟹ $\frac{1}{1-(1-e^{-G})} = \frac{1}{e^{-G}} = e^{G}$

The exponential dependence indicates that small increase in channel load (G) drastically reduces its performance.

Pure ALOHA (3)



Throughput versus offered traffic for ALOHA systems.

Carrier sense protocols: Stations listen for a carrier and act accordingly.

1-persistent: Senses carrier to find it to be idle. If it is found to be idle, then a sta^n transmits its frame (prob of tx ~~sensing~~ is 1). If a second sta^n senses a channel to be idle where the first station's tx is yet to be received, then there can be a chance of experiencing collision (if the second one transmits).

⇒ in case a channel is found to be busy, a sta^n waits until it gets idle and then it tx.

⇒ problem: sure tx after finishing the ongoing sensed tx. It can result in collision in case of multiple sensing sta^ns.

If a sta^n experiences a collission after sending its packet, then it waits for a random amount of time, and then starts all over again.

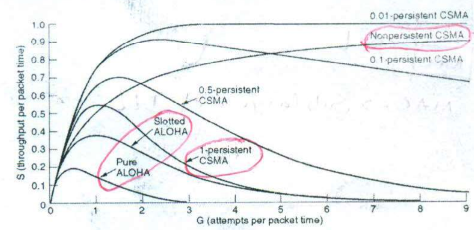
⇒ If the propaga^n delay is larger, then the chance of collission is higher.

Even if the prop^n delay is zero, collission may happen in case of an attempt made at the beginning of a slot.

Non-persistent ⇒ in case of sensing a busy channel, waits for a random time. If the channel is idle, then it starts transmitting.

the middle of third sta^n's tx. when the tx is over, both waiting sta^ns start attempting tx resulting a collision

Persistent and Nonpersistent CSMA



Comparison of the channel utilization versus load for various random access protocols.

p-persistent ⇒ uses slotted channels. If a sta^n becomes ready to send, it senses the channel. If the channel is found to be idle, then the sta^n txs with prob p or defers until beginning of the next slot by prob $q = 1-p$. If next slot is idle, then it either txs or defers by prob p or q..... If the channel is found busy, it waits until the next slot and applies the above algo.

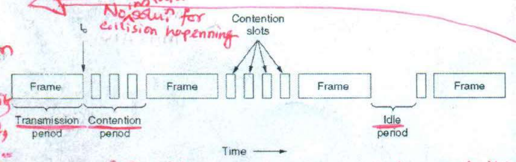
problem solved by using p prob of tx.

CSMA with Collision Detection

CSMA/CD

⇒ tx stops as soon as a collision is detected, then waits for a random time, and try again assuming that no other sta^n started tx in the meantime.

⇒ Alternating conten^n and tx periods. Also, there can be idle periods.



⇒ τ → time required to reach the most distant sta^n

⇒ A sta^n cannot be sure that it has seized the channel until it has transmitted for 2τ without hearing a collision. After seizing, tx could go at any rate.

CSMA/CD can be in one of three states: contention, transmission, or idle.

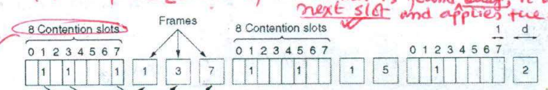
⇒ Collision detec^n → Analog process

⇒ Half-duplex system, as the receiving line is busy for idle listening to detect collision.

⇒ Bad performance for large 2τ; This can happen for long cable, small frame.

⇒ Problem of CSMA/CD → collision can still occur in contention period.

Collision-Free Protocols



⇒ Reserva^n protocol: Broadcasts desire to tx before actual tx

⇒ Each contention period consists of exactly N slots, 1 for each sta^n. A sta^n inserts a 1 to its allocated slot, if it has a frame to tx.

⇒ After the conten^n slots get over, each sta^n has a complete knowledge of which sta^ns wish to tx. They begin tx in numeric order.

The basic bit-map protocol.

⇒ low load: high #ed sta^ns need to wait for 0.5N slots on an avg; low #ed sta^ns " " " " 1.5N " " "

Considering that a sta^n gets ready in the middle of conten^n period, So, sta^ns need to wait for N bits on an avg ← overhead

⇒ High load: All sta^ns mostly have something to send all the time; So, the overhead is only 1 bit per frame

⇒ Efficiency: $\frac{d}{d+N}$ for low load; $\frac{d}{d+1}$ for high load [d → # of data bits]

Collision-Free Protocols (2)

⇒ High priority for higher #ed sta^n

⇒ Broadcasts address as a binary string, with higher #ed bit first

⇒ Equal length for all addresses

⇒ A sta^n gives up broadcasting its intent through broadcasting its binary bit string address, once it sees a higher-order bit pos^n set by someone else

⇒ If the first field of a frame could be made address, then efficiency will be 100%.

⇒ Parallel interface: Successful sta^ns get circularly permuted in order to give higher priorities to sta^ns that have been silent unusually long.

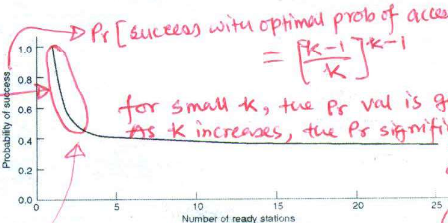
	Bit time 0 1 2 3
0 0 1 0	0 – – –
0 1 0 0	0 – – –
1 0 0 1	1 0 0 –
1 0 1 0	1 0 1 0
Result	1 0 1 0

Stations 0010 and 0100 see this 1 and give up

Station 1001 sees this 1 and gives up

The binary countdown protocol. A dash indicates silence.

Limited-Contention Protocols



Acquisition probability for a symmetric contention channel

Tries to operate here

⇒ Pr [success with optimal prob of access to channel] $= \left[\frac{k-1}{k}\right]^{k-1}$

for small k, the Pr val is good. As k increases, the Pr significantly decreases towards an asymptotic value of 1/e (Best case for Slotted ALOHA)

To do so, sta^ns are divided into (not necessarily disjoint) groups. Then, groups follow collision free protocol among them. On the other hand, sta^ns within a group follow collision free protocol

Slotted ALOHA among sta^ns; contention protocol

Gr1 Gr2 - - -

Bitmap or Binary countdown contention-free protocol

Extreme cases

① 1 sta^n per group: Binary countdown/bitmap

② 1 gr with all sta^ns: Slotted ALOHA

⇒ Performance measure: ① delay for low-load, ② Efficiency for high load

→ low-load: Pure/Slotted ALOHA

→ High-load: Collision-free protocol (as efficiency higher)

⇒ P: Prob of channel acquire/access; K: # of sta^ns contending for channel

Pr [some sta^n successfully acquire the channel] $= kP(1-P)^{k-1}$

For best performance or optimal p, $\frac{dPr}{dp} = 0$; So, $k(1-P)^{k-1} - k(k-1)P(1-P)^{k-2} = 0$

So, $k(1-P) - k(k-1)P = 0$; ∴ $k - kP - k^2P + kP = 0$; ∴ $P = 1/k$

So, Pr [success with optimal P] $= k . \frac{1}{k}(1 - \frac{1}{k})^{k-1} = \left(\frac{k-1}{k}\right)^{k-1}$