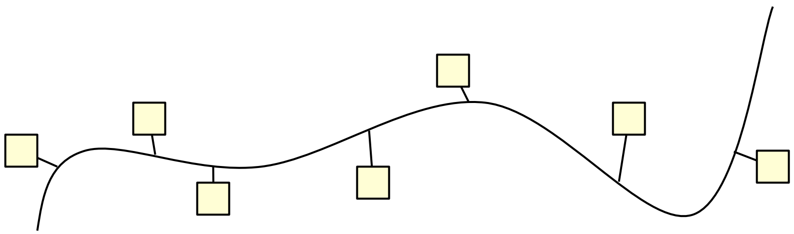
COMP2011 -- **Networks**, Databases and Graphics

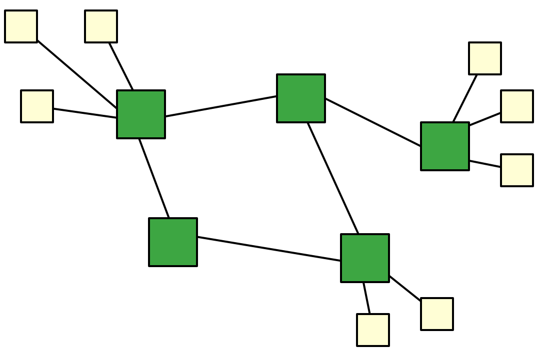
5. Sharing the Media

Dr. Shi Zhou
Department of Computer Science
University College London

Sharing

- Sharing a link
  - TDM, FDM - fixed allocations
    - Sub-channels each used by a separate activity
  - Statistical multiplexing - variable allocations
    - Packets of separate activities mixed in an arbitrary order
- Sharing a network
  - Circuit switching
  - Packet switching
    - Statistical multiplexing
    - Favoured by the bursty nature of computer traffic





What about sharing the medium?

- To eliminate the switches and share the media directly
- Need a way to control medium access
  - Only one computer can be using it at any one time
  - “Medium Access Control” (MAC) algorithms

Approaches

- Bad ideas
  - TDM - stations allocated time slots and take turns to transmit
  - FDM - each station transmits on a different frequency (Rx need to listen on all frequencies)
  - Both waste resource when station not sending
  - Hard to add stations – need more slots or frequencies
- Good ideas ?
  - Contention
    - If one is speaking, everyone else keeps quiet
    - What if two people start talking simultaneously? (Collision)
  - Token
    - A token is passed around in rotation.
    - Only one with the token is allowed to transmit.

Aloha protocol - simple contention

- Packet radio - University of Hawaii - 1970s
- "Contention-based" system
  - Transmitters uncoordinated => collisions
- Collision recovery - wait a random time and try again
  - Need to prevent collisions occurring systematically on retransmissions.
  - There can be considerable (propagation) delay before collisions can be detected.
    - A geostationary satellite sits 35km above the equator. The round-trip delay is 240ms!

Contention systems

- Contention systems allow conflicts between stations in order to solve channel allocation problem.
- Ideally,

– when traffic is light, all transmissions should succeed with minimal delay

– when traffic is heavy, the full capacity should be available and shared fairly between competing stations.

– however, most schemes fall a long way short of this.

- Performance evaluation

– Direct measurement of real systems (not simple)

– Simulation

– Mathematical modelling

Aloha Utilisation

- What proportion of the time is the network doing useful work? Here is a mathematical model:

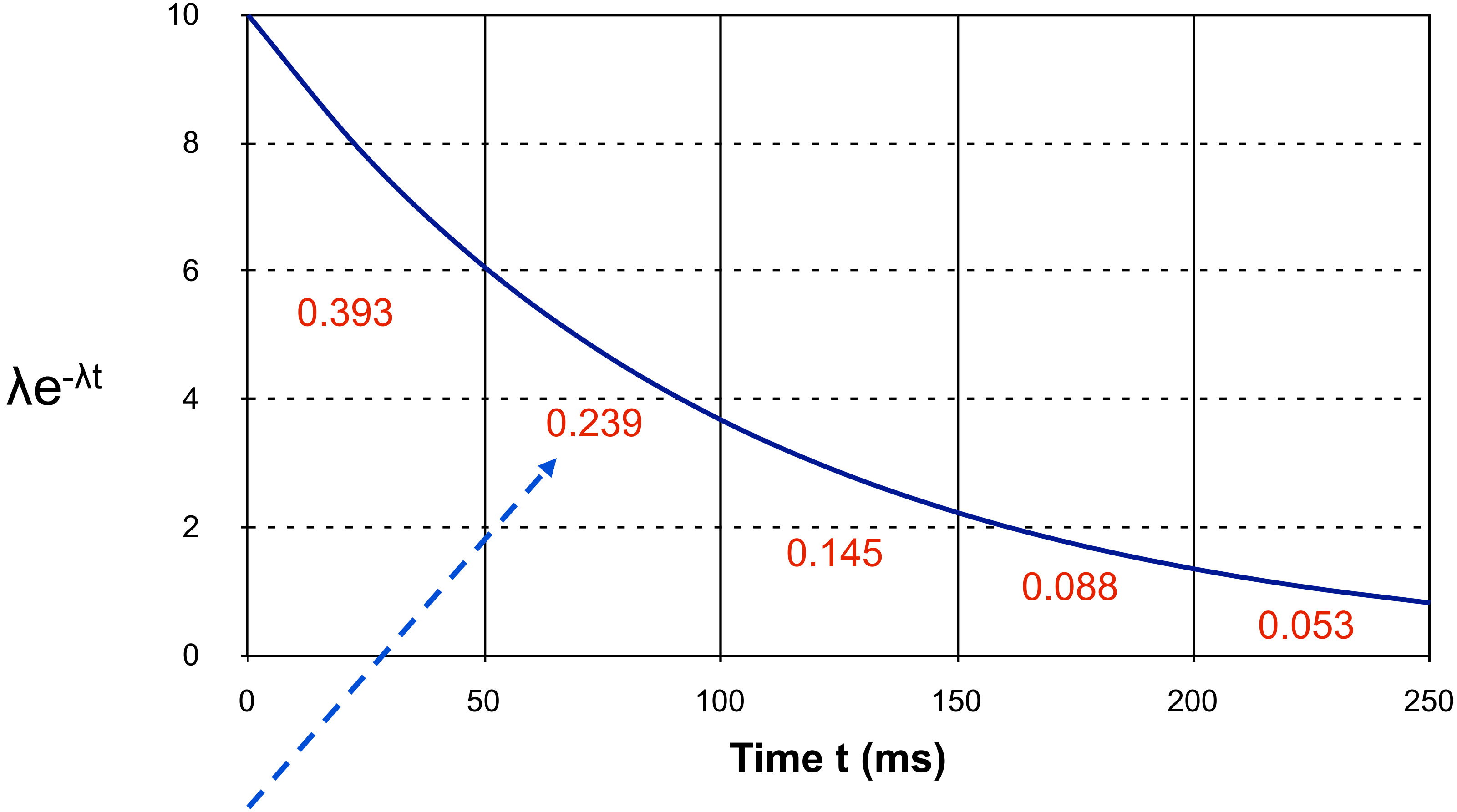
1. Suppose each packet takes T second (sec) to transmit, e.g. T=0.01 sec
   - assume all packets are the same length
   - T as our unit of time, "packet time"
2. Define "**Normalised Load**" G = number of packets generated in T sec, e.g. 20 packets/sec => G=0.2
3. Define "**Normalised Throughput**" S = number of packets delivered in T sec (N.B. S < G, S<1)
4. Assume r collisions, hence r retransmissions in T sec
   - We have S = G – r or r =G-S (definition of r)
   - E.g. if 20% collide, S=(80% of G), and so with G=0.2, S=0.16

1st Result: P(no collision) = S/G

This is our first form of the probability of no collision!

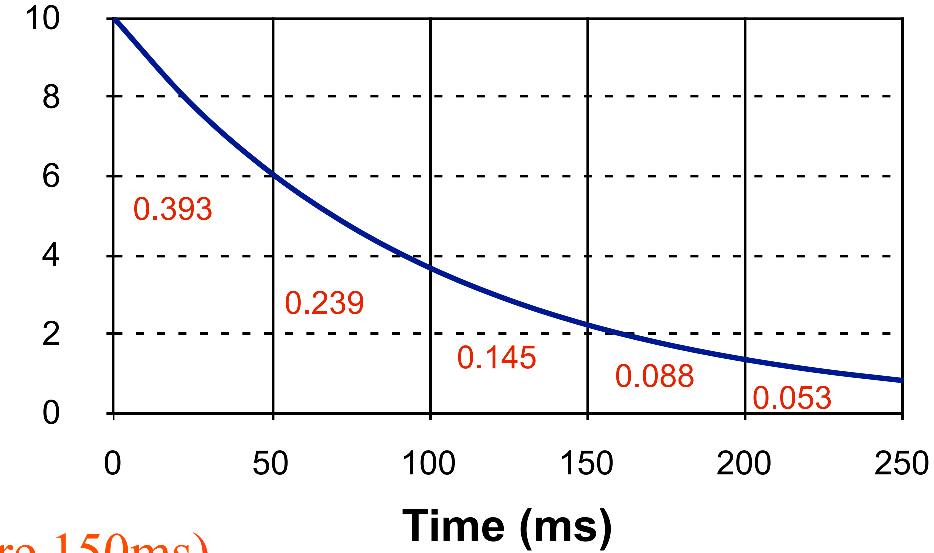
Assumption

We assume that the intervals between transmissions have **negative exponential distribution,** $\lambda e^{-\lambda t}$, where λ (lambda) is the mean arrival rate (10 arrivals per second) and t is time.



The probability that next packet arrives between 50ms and 100ms **from now** is 0.239, which is the area under the curve between 50ms and 100ms.

Negative Exponential Distribution is memory-less



An example:

Suppose there have been no arrivals for 150 ms, i.e. have waited this long.

What is the probability that there will be an arrival during the next 50 ms?

We want a conditional probability

P(arrival before 200ms | no arrival before 150ms)

By Bayes theorem:

$$P(A \mid B) = \frac{P(A \wedge B)}{P(B)}$$

P(arrival before 200ms $\wedge$ no arrival before 150ms) = 0.088 (from graph)

P(no arrival before 150ms) = 1 - (0.393 + 0.239 + 0.145) = 0.223

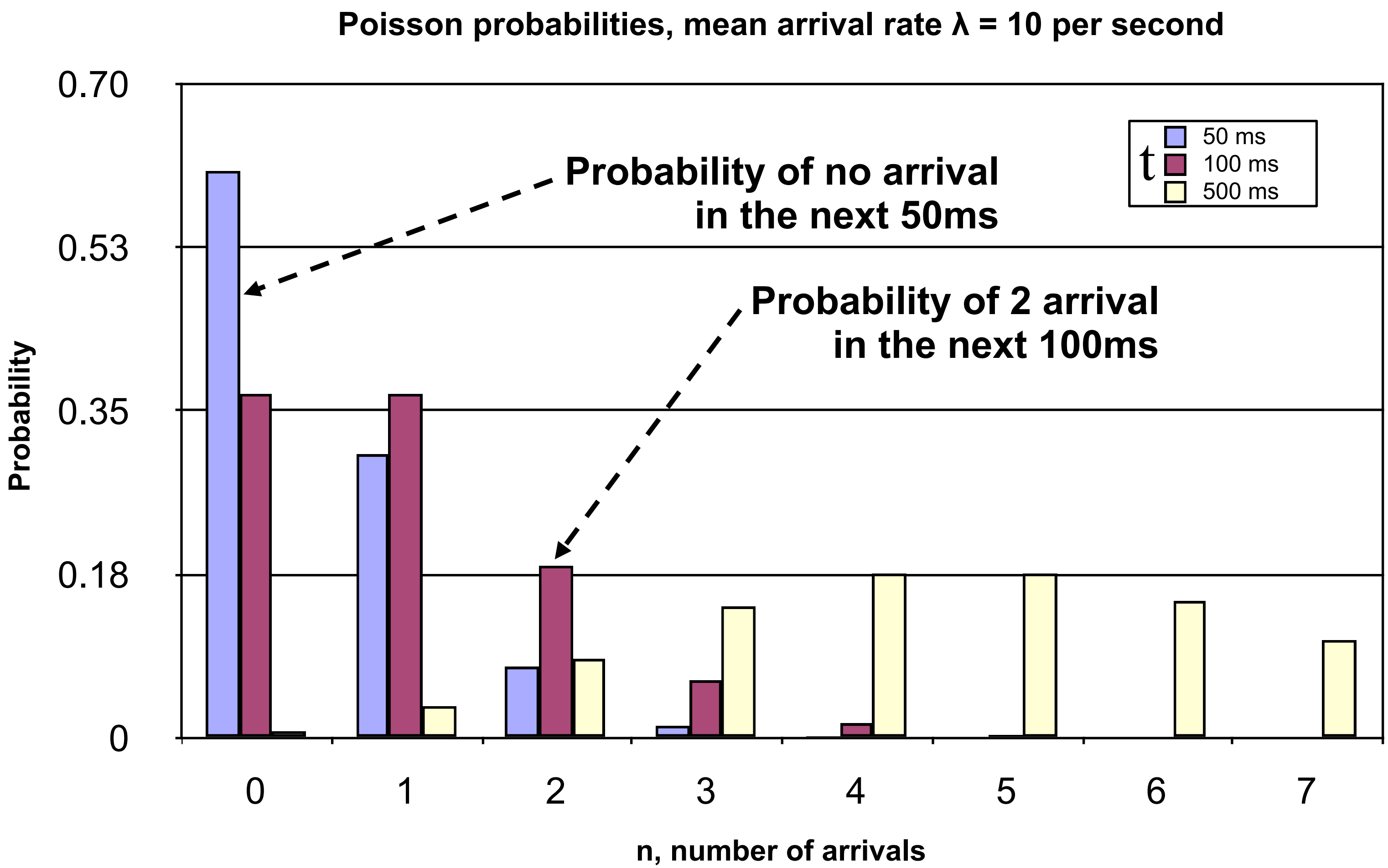
P(arrival before 200ms | no arrival before 150ms) = 0.088 ÷ 0.223

= 0.395 ≈ P(arrival in first 50ms)

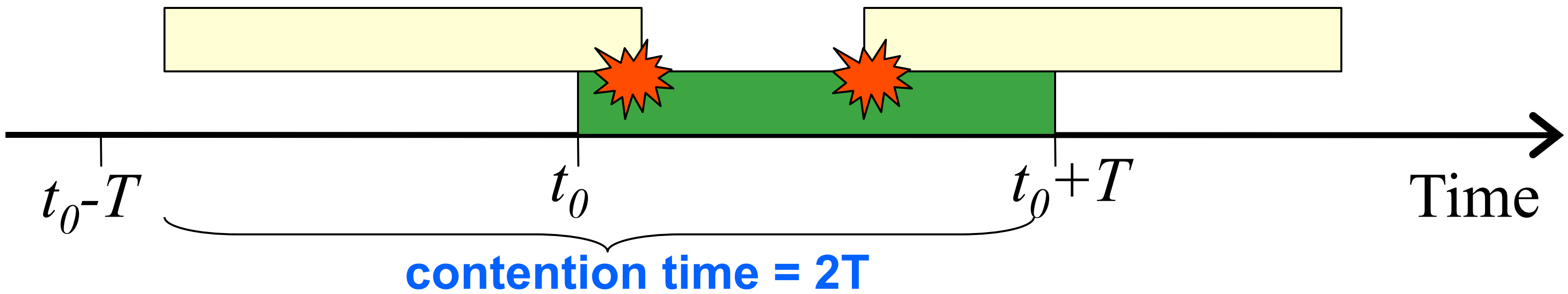
Poisson distribution

If inter-arrival times are Negative Exponential, the probability of n arrivals within time t is given by the Poisson Distribution

$$P_n(t) = \frac{(\lambda t)^n e^{-\lambda t}}{n!}$$



Aloha Collisions: another formula for the probability of a collision



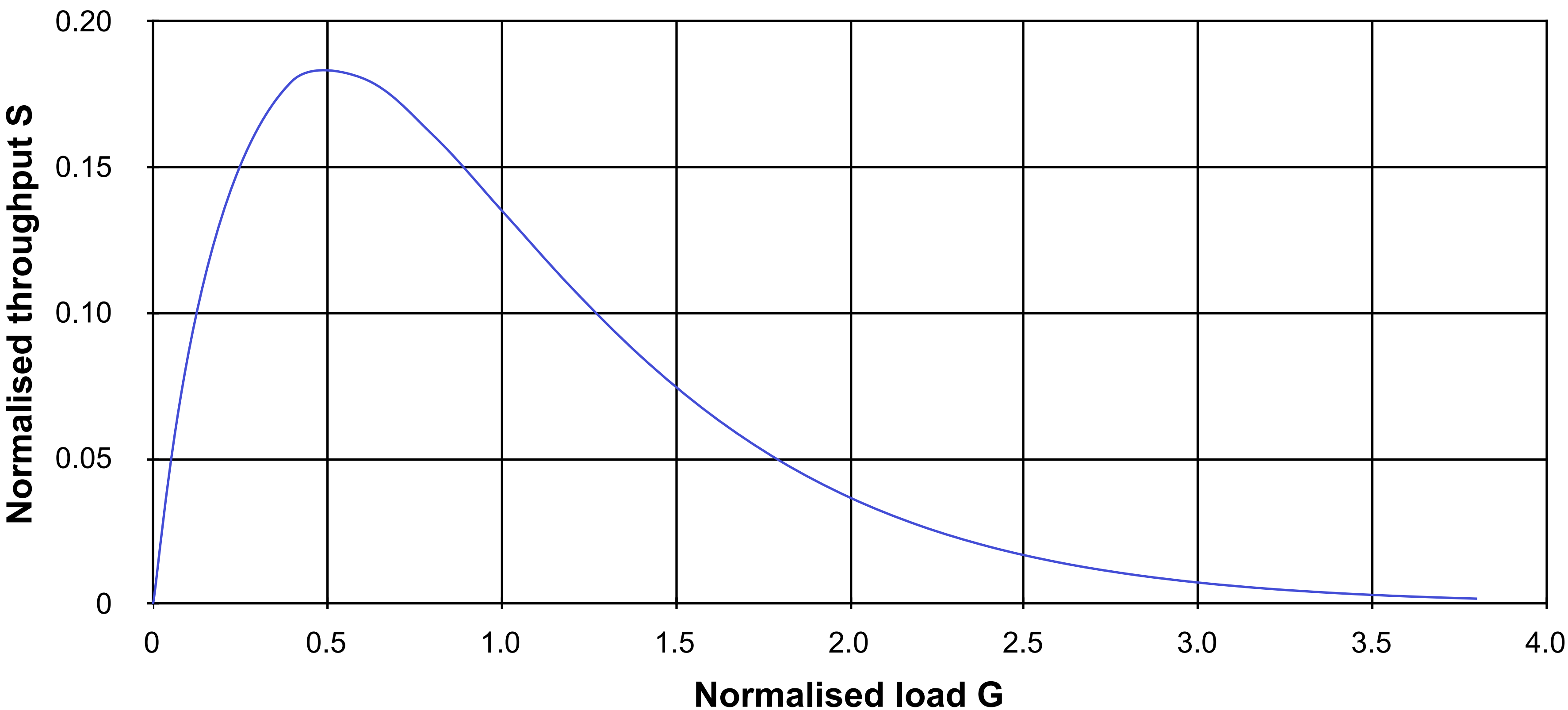
- A packet will collide with any transmission beginning in the interval (t_0-T, t_0+T), i.e. a “contention time” of 2T sec
- What is probability, P_0(2T), of no transmissions in an interval 2T?
- Assume Poisson. G packets in T sec => $\lambda = G/T$ packets/sec

$$P_n(t) = \frac{(\lambda t)^n e^{-\lambda t}}{n!}, \text{ so } P_0(2T) = \frac{(2G)^0 e^{-2G}}{0!} = e^{-2G}$$

This is our second form of the probability of no collision!

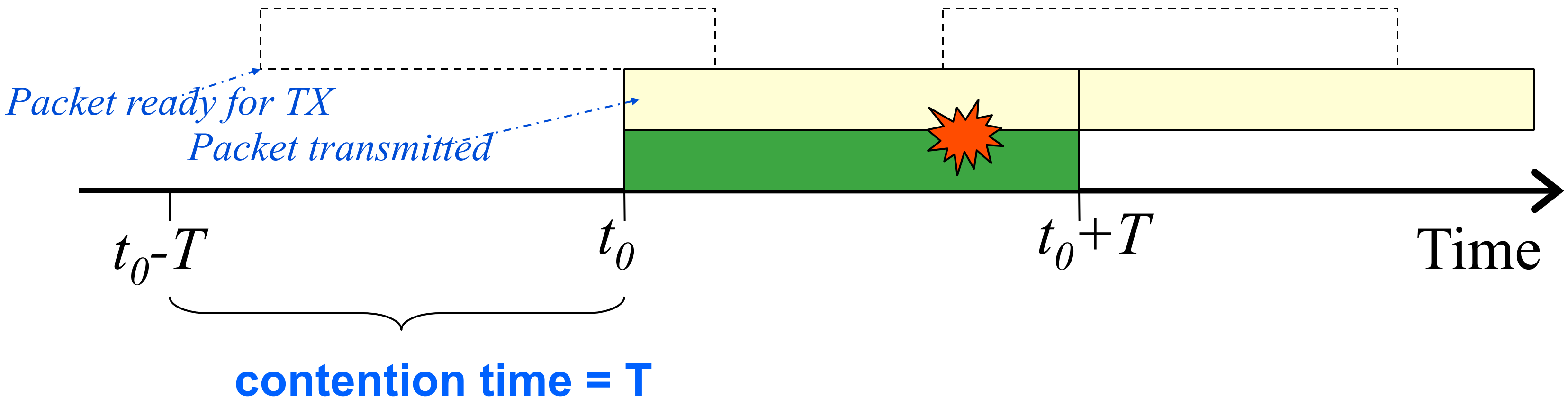
Aloha Performance

- We have from our 1st Result: P(no collision) = S/G
- We have from our 2nd result: P(no collision) = e^{-2G}
- Equating these gives $S = Ge^{-2G}$



Maximum normalised throughput is S_{max} = 18.4% when normalised load G=0.5

Improvement: Slotted Aloha

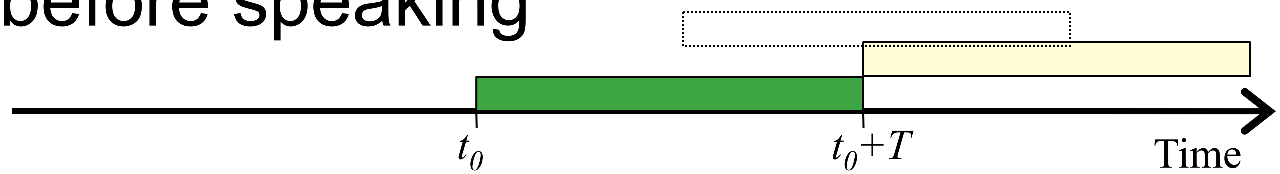


- Packet transmissions are constrained to begin on slot boundaries
  - Must have a reliable time source to identify boundaries

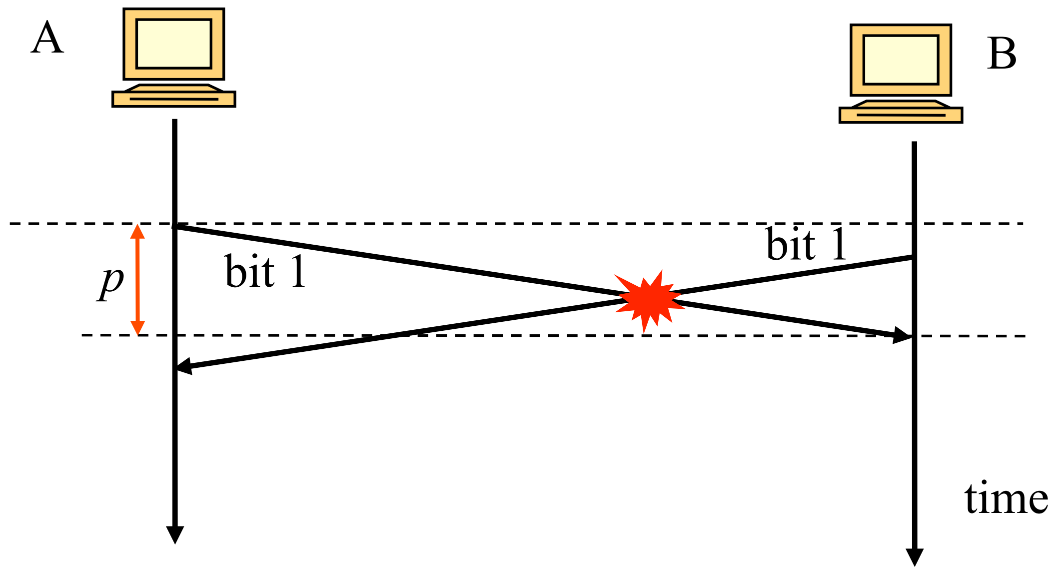
- Packets now vulnerable for a contention time T
- Utilisation is now 36.8%
  - But there is an extra ½T delay on average

Alternative approach to increase throughput: Carrier Sense Multiple Access (CSMA)

- "Listen before speaking"



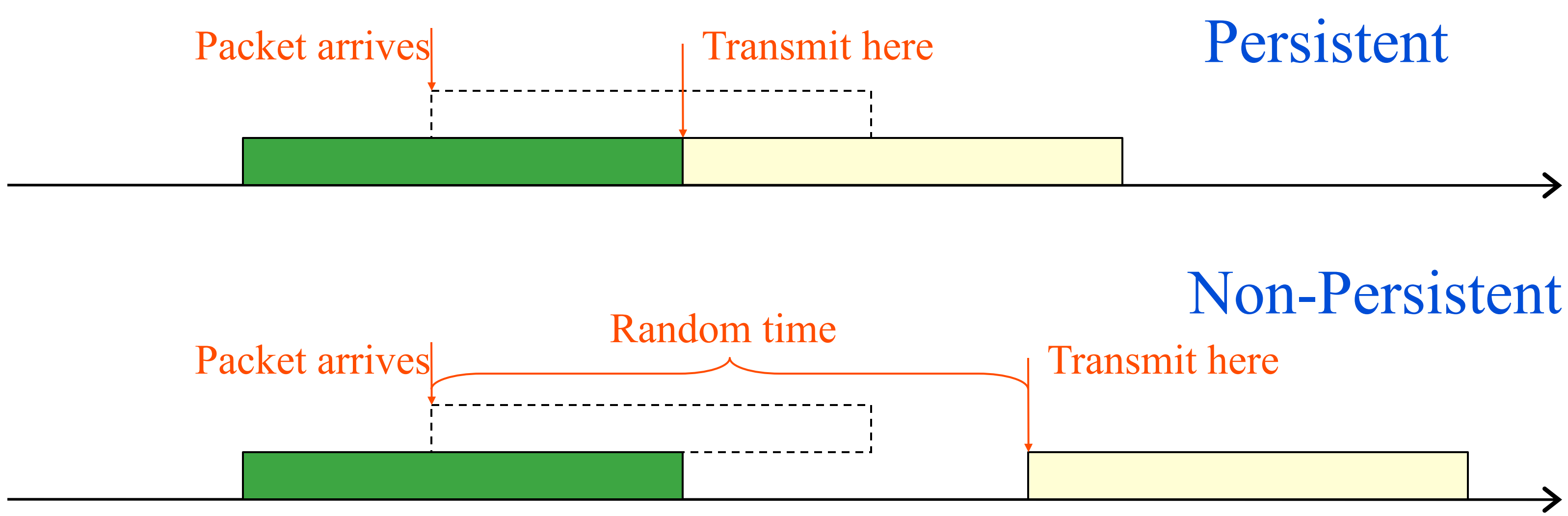
- Avoids collisions?
- No!
  - Contention time is the max bit propagation time p
  - Satellite delay is 240ms !
  - But it helps when propagation delay is low, e.g. for a small LAN



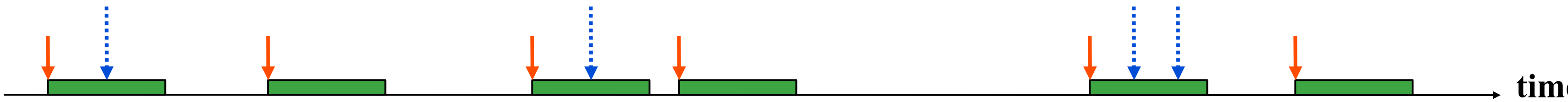
CSMA come in 2 forms:
Persistent and Non-persistent CSMA

Persistent: stations transmit immediately when the medium goes quiet : there will be many collisions (especially when network load is heavy).

Non persistent – “if medium is busy wait a random time and retry”



Non-persistent CSMA throughput

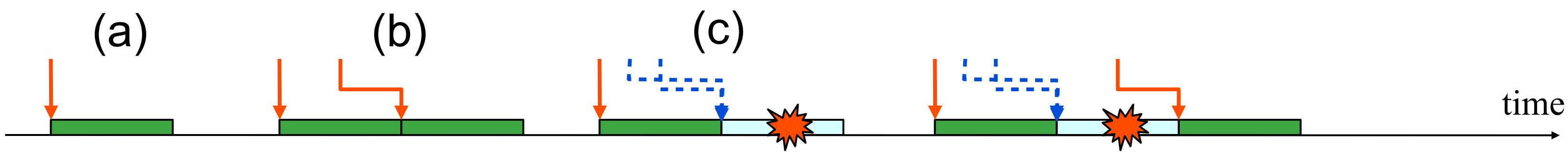


- The arrows show instances of a station listening
  – Short red arrows show "network free" and packet transmission
  – Long blue arrows show "network busy" that result in deferrals
- The load is G packets per unit time
  – Assume fixed length packets. Unite time is time to send a packet.
- The successful delivery rate is S, S<1
- The deferral rate is D, D = G - S

- In a unit time we deliver S packets on average
  - $\therefore$ Network is busy for a fraction S of the time, so $D = GS$

$$GS = G - S \Rightarrow S(G+1) = G \Rightarrow S = G/_{G+1}$$

Persistent CSMA throughput



- If 0 or 1 packets arrive during a transmission (a or b) there will be no collision
- If n>1 packets arrive during a transmission (c) there will be a collision
  - ∴ P(>1 packet in a unit time) = 1 - P(0) - P(1)
  - ∴ Assuming Poisson $P_n(t) = \dfrac{(\lambda t)^n e^{-\lambda t}}{n!}$, P(>1) = 1 - e^{-G} -Ge^{-G}

- In unit time we deliver S packets and suffer C collisions on average
  - Both of these last one packet time
    - ∴ Network is busy for a fraction $S+C$ of the time
- Fraction of transmissions which generate a collision is $\dfrac{C}{S+C}$

$$\frac{C}{S+C} = 1 - e^{-G} - Ge^{-G} \Rightarrow C = (S+C)(1 - e^{-G} - Ge^{-G}) \qquad [1]$$

$$\text{So, } C(e^{-G} + Ge^{-G}) = S(1 - e^{-G} - Ge^{-G}) \qquad [1a]$$

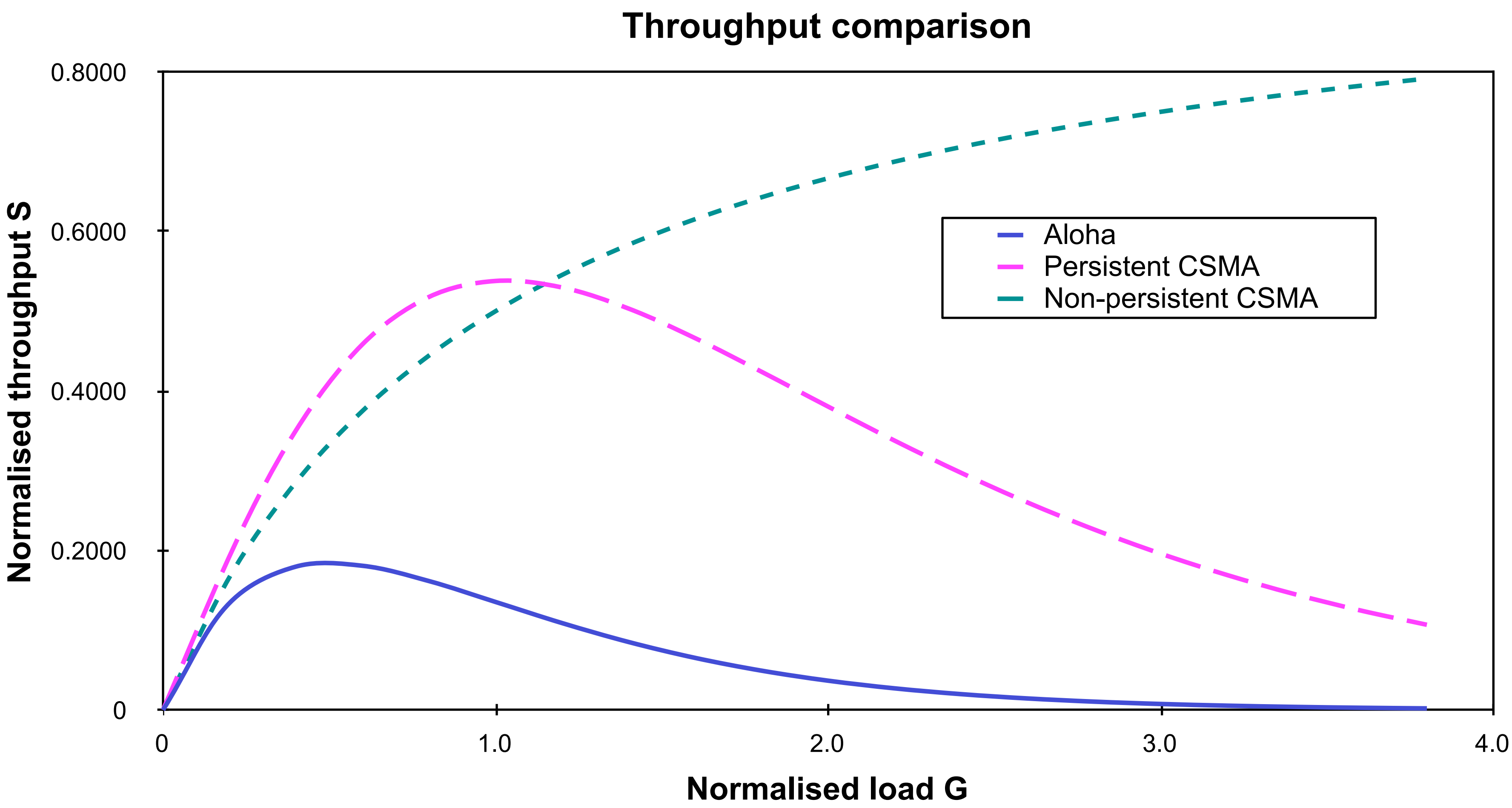
Persistent CSMA Throughput [2]

- What fraction of **arrivals** is lost in collisions?
  - This is P(net is busy & 1 or more **further** arrival occurs during the transmission) = $(S+C)(1-e^{-G})$
- All arrivals not lost are delivered

  $\therefore S = G - G(S+C)(1-e^{-G})$ [2]
- Combining with [1a] gives:

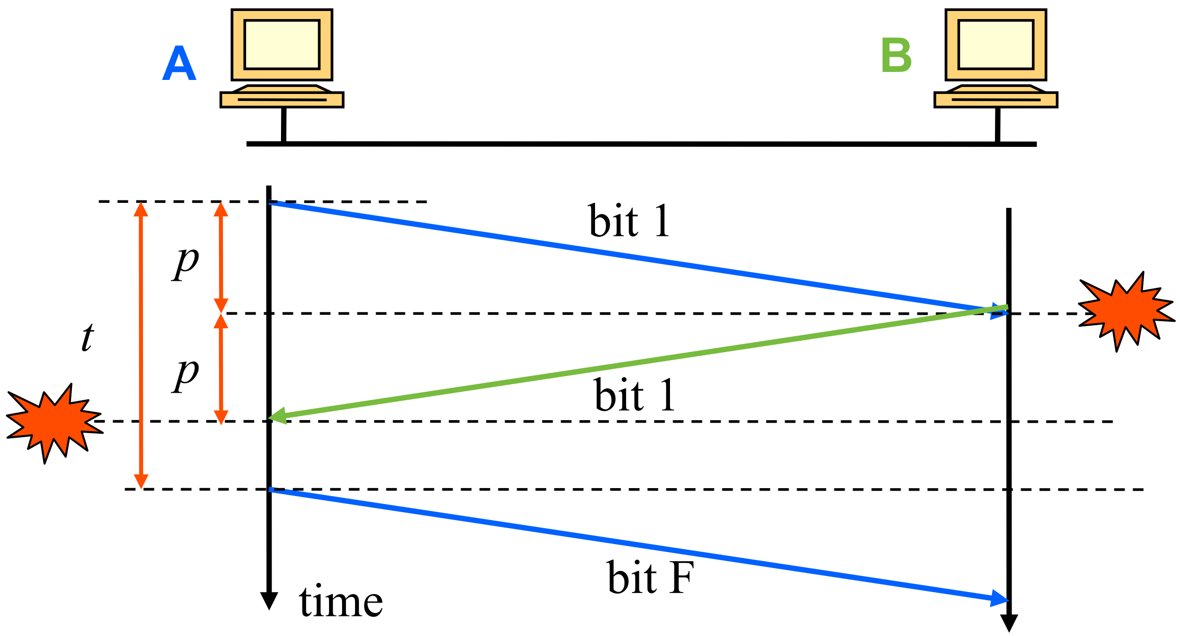
$$S = \frac{G(e^{-G} + Ge^{-G})}{G + e^{-G}}$$

Throughput Comparison



CSMA with Collision Detection (CSMA-CD)

- In CSMA, collisions last one packet time
  - Why not stop once a collision is detected?
  - How long does it take to detect a collision?



- Answer: 2p, where p is the maximum propagation delay

Example: 10Mbps Ethernet with 1500 byte packets:

t_{max}=1.2ms, but p=10μs (with distance of 2Km)

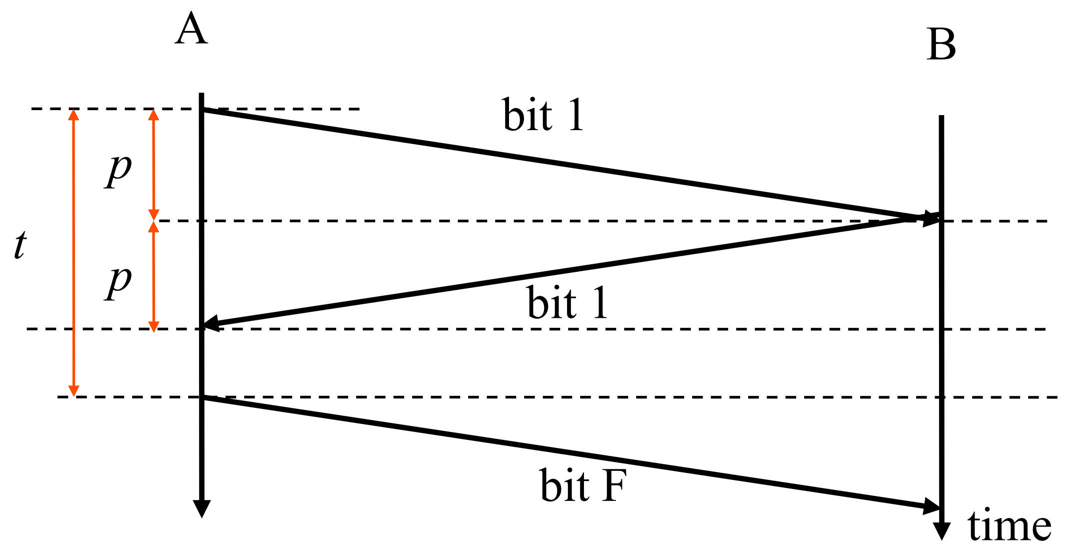
CSMA-CD Performance

- CSMA-CD is a persistent strategy with back-off on a collision
- When load is low,
  - Negligible collisions
  - Performs like persistent CSMA (Tx transmit as soon as channel not-busy)
- When load is high, many collisions
  - Collisions handled by “binary exponential back-off”
  - Effectively a deferral
  - Performs like non-persistent CSMA
- Best of both worlds!
- But …
  - There is a cost (see next slide)
  - CD is difficult in wireless networks

CSMA-CD

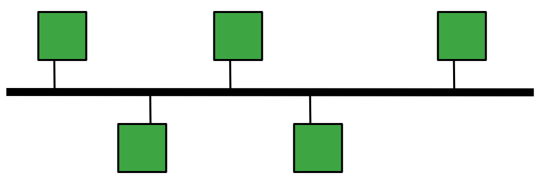
- Protocol used on the Ethernet

- For a given p there is a minimum frame-size
- Must have $t > 2p$ or collisions may be missed
- 10Mbps Ethernet, min. frame is 64 bytes

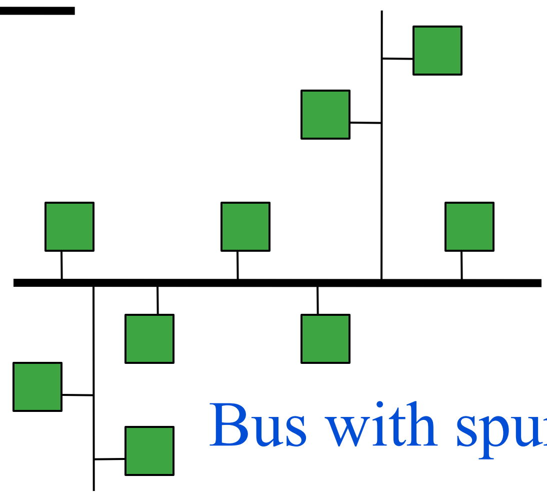


Ethernet

- Originally on coax cable, now mainly on “twisted pair”
- Faster versions often use optical fibre
- Binary exponential back-off



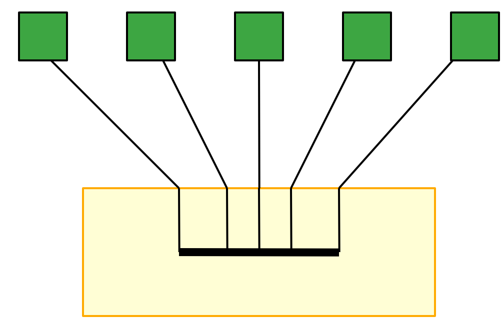
Bus



Bus with spurs

Original 10Mbps single coax cable

Modern 10/100/1000Mbs

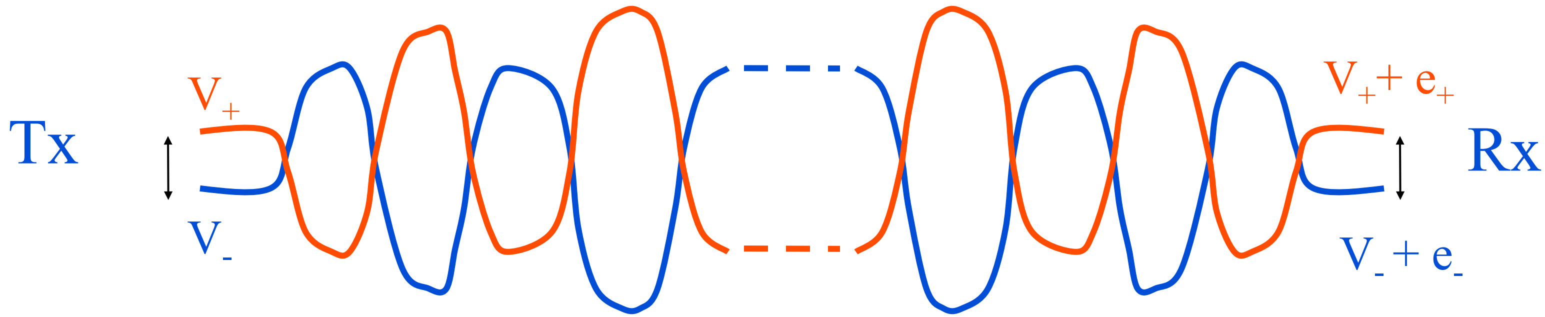


Hub

Ethernet

	Coax cable	Twisted pair	Optical fibre/ Twisted pair
Frequency, f	10Mbps	100Mbps	1000Mbps
Speed of light, c	2×10^8m/s	2×10^8m/s	2×10^8m/s
Span, s	2500m	200m	200m
Bit Tx time, p (=s/c)	12.6μs	1.01μs	1.01μs
Num of Pkt in 2p (2pf/8)	31.5 bytes	25.3 bytes	253 bytes
Min frame size (>2pf/8)	64 bytes	64 bytes	512 bytes

Twisted Pair Cables - two wires twisted around each other



Two wires driven to opposite voltages at Tx, e.g. +0.6V, -0.6V
Direction of voltage difference carries binary value.
Thus, Tx sets wires to V_+ and V_-

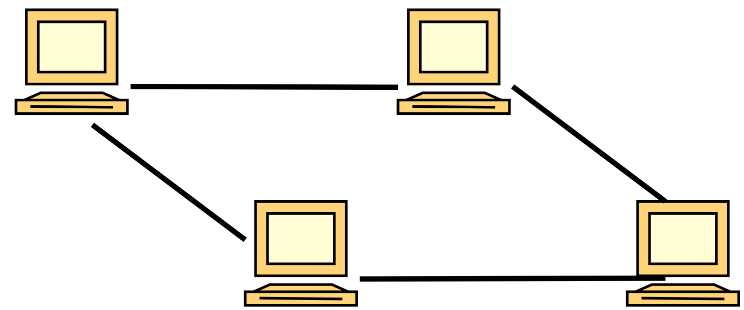
Rx measures voltage difference received.
Rx receives $V_+ + e_+$ and $V_- + e_-$, where e_+, e_- are noise signals

Voltage difference at Rx = $(V_+ - V_-) + (e_+ - e_-)$ = applied voltage

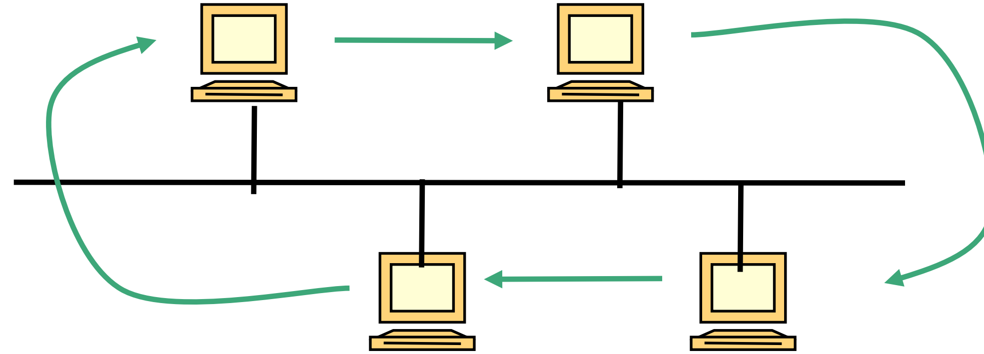
For twisted pair, $e_+ == e_-$ to first order so they cancel on subtraction

Token passing

- Token:
  - is passed around a ring
  - token holder transmits a frame
  - token is then released
- Topology:
  - physical ring
  - logical ring
- Mechanisms are needed to cope with lost token, priority etc.



Token ring (physical ring)



Token bus (logical ring)

Token Passing - Performance

- At light loads, CSMA-CD tends to be more efficient
  - The time for the token to circulate back to a transmitter becomes significant
- At heavy loads, token passing wins
  - If all stations are trying to transmit, the system behaves like TDM
  - High utilisation without collision
- It is possible to determine the maximum delay before a transmission succeeds on a token ring
  - It is when each station has a maximum-size frame to transmit
  - Important in some time-critical applications such as process control.

The End