18-452/18-750
Wireless Networks and Applications
Lecture 8: LAN MAC Protocols
Wireless versus Wired
Peter Steenkiste

Fall Semester 2018
http://www.cs.cmu.edu/~prs/wirelessF18/

Outline

• Data link fundamentals
  » And what changes in wireless
• Aloha
• Ethernet
• Wireless-specific challenges
• 802.11 and 802.15 wireless standards

Datalink Functions

• Framing: encapsulating a packet into a bit stream.
  » Add header, mark and detect frame boundaries, ...
• Logical link control: managing the transfer between the sender and receiver, e.g.
  » Error detection and correction to deal with bit errors
  » Flow control: avoid that the sender outruns the receiver
• Media access: controlling which device gets to send a frame next over a link
  » Easy for point-to-point links; half versus full duplex
  » Harder for multi-access links: who gets to send?

Framing

• Typical structure of a “wired” packet:
  » Preamble: synchronize clocks sender and receiver
  » Header: addresses, type field, length, etc.
  » The data to be send, e.g., an IP packet
  » Trailer: padding, CRC, ..

• How does wireless differ?
  » Different transmit rates for different parts of packet
  » Explicit multi-hop support
  » Control information for physical layer
  » Ensure robustness of the header
Error Control: Error Detection and Error Recovery

- **Detection**: only detect errors
  - Make sure corrupted packets get thrown away, e.g. Ethernet
  - Use of error detection codes, e.g. CRC
- **Recovery**: also try to recover from lost or corrupted packets
  - Option 1: forward error correction (redundancy)
  - Option 2: retransmissions
- **How does wireless differ?**
  - Uses CRC to detect errors, similar to wired
  - Error recovery is much more important because errors are more common and error behavior is very dynamic
  - What approach is used?

Error Recovery in Wireless

- **Use of redundancy**: very common at physical layer – see PHY lectures
- **Use of Automatic Repeat Request (ARQ)**
  - Use time outs to detect loss and retransmit
- **Many variants**:
  - Stop and wait: one packet at a time
    - The most common at the datalink
  - Sliding window: receiver tells sender how much to send
    - Many retransmission strategies: go-back-N, selective repeat, ...
- **When should what variant be used?**
  - Noise versus bursty (strong) interference

Stop and Wait

- **Simplest ARQ protocol**
- Send a packet, stop and wait until acknowledgement arrives
- Will examine ARQ issues later in semester
- **Limitations?**
- **What popular for the datalink?**

Media Access Control

- **How do we transfer packets between two hosts connected to the same network?**
- **Using point-to-point “links” with “switches” -- store-and-forward**
  - Very common in wired networks, at multiple layers
- **Multiple access networks**
  - Multiple hosts are sharing the same transmission medium
  - Need to control access to the medium
  - Taking turn versus contention based protocols
- **What is different in wireless?**
  - Is store and forward used?
  - Is multiple access used?
Datalink Architectures

- Routing and packet forwarding.
- Point-to-Point error and flow control.
- Media access control.
- Scalability.

Switched ethernet, mesh and ad hoc networks

Traditional ethernet, Wifi, Aloha, ...

Multiple Access Networks

- Who gets to send a packet next?
- Scheduled access: explicit coordination ensures that only one node transmits
  - Looks cleaner, more organized, but ...
  - Coordination introduces overhead – requires communication (oops)
- Random access: no explicit coordination
  - Potentially more efficient, but ...
  - How does a node decide whether it can transmit?
  - Collisions are unavoidable – also results in overhead
  - How do you even detect a collision?

Outline

- Data link fundamentals
  - And what changes in wireless
- Aloha
- Ethernet
- Wireless-specific challenges
- 802.11 and 802.15 wireless standards
**Why ALOHA**

- Developed in University of Hawaii in early 1970’s.
- It does not get much simpler:
  1. A user transmits at will
  2. If two or more messages overlap in time, there is a collision – receiver cannot decode packets
  3. Receive waits for roundtrip time plus a fixed increment – lack of ACK = collision
  4. After a collision, colliding stations retransmit the packet, but they stagger their attempts randomly to reduce the chance of repeat collisions
  5. After several attempts, senders give up
- Although very simple, it is wasteful of bandwidth, attaining an efficiency of at most $1/(2e) = 0.18$

**Poisson Process**

- A Poisson process of “rate” $\lambda > 0$ is a counting process $a(t)$ which satisfies the following conditions:
  1. The process has independent increments in disjoint intervals
     - i.e., $a(t_1+\Delta t)-a(t_1)$ is independent of $a(t_1)-a(t_2)$ if $[t_1, t_1+\Delta t]$ and $[t_2, t_2+\Delta t]$ are disjoint intervals
  2. The increments of the process are stationary.
     - i.e., $a(t_1+\Delta t)-a(t_1)$ does not depend on $t_1$
  3. The probability of exactly one event occurring in an infinitesimal interval $\Delta t$ is $P[a(\Delta t) = 1] \equiv \lambda \Delta t$
  4. The probability that more than one event occurs in any infinitesimal interval $\Delta t$ is $P[a(\Delta t) > 1] \equiv 0$
  5. The probability of zero events occurring in $\Delta t$ is $P[a(\Delta t) = 0] \equiv 1-\lambda \Delta t$

**Poisson Distribution**

- Above definitions lead to: Probability $P(k)$ that there are exactly $k$ events in interval of length $T$ is,
  $$P(k) = \frac{(\lambda T)^k e^{-\lambda T}}{k!}$$
- We call the above probability the “Poisson distribution” for arrival rate $\lambda$
- Its mean and variance are:
  $$E(k) = \lambda T$$
  $$\sigma_k^2 = E(k^2) - E^2(k) = \lambda T$$
- Many nice properties, e.g. sum of a $N$ independent Poisson processes is a Poisson process
Pure ALOHA: Model

- Let there be N stations contending for use of the channel.
- Each station transmits λ packets/sec on average based on a Poisson arrival process.
- All messages transmitted are of the same fixed length, m, in units of time.
- Let new traffic intensity be $S \equiv N \lambda m$.
- Since all new packets eventually get through, ‘S’ is also the network throughput.

Pure ALOHA: Vulnerability

- Simplification: assume the retransmitted messages are independent Poisson process as well.
- The total rate of packets attempting transmission = newly generated packets + retransmitted ones = $\lambda^*$.
- The total traffic intensity (including retransmissions) is, $G = N \lambda^* m$.
- The “vulnerable period” in which a collision can occur for a given packet is $2m$ sec.

Pure ALOHA: Analysis

- Calculate the “Probability of no collision” two ways:

  1. Probability that there is no arrival in interval $2m$ sec:
     $$P(\text{no arrival in } 2m \text{ sec}) = e^{-2N \lambda^* m} = e^{-2G}$$

  2. Since all new arrivals eventually get through, we have
     $$\lambda/\lambda^* = S/G = \text{Fraction of transmissions that are successful}$$

- So,
  $$S/G = e^{2G} \quad \text{Maximum Throughput of Pure Aloha}$$

Analysis Conclusion

- $S$ is maximum at $S = \frac{1}{2e}$ at $G = 0.5$.
**Slotted ALOHA**

- Transmission can only start at the beginning of each slot of length T
- Vulnerable period is reduced to T
  - Instead of $2T$ in Aloha
- Doubles maximum throughput.

**Discussion of ALOHA**

- Maximum throughput of ALOHA is very low $1/(2e) = 18\%$, but
  - Has very low latency under light load
- Slotted Alohas has twice the performance of basic Aloha, but performance is still poor
  - Slightly longer delay than pure Aloha
  - Inefficient for variable sized packets!
  - Must synchronize nodes
- Still, not bad for an absolutely minimal protocol!
  - Good solution if load is low – used in some sensor networking technologies (cheap, simple)

- How do we go faster?