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

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?

Scheduled Access MACs

- Polling: controller polls each node
- Reservation systems
  - Central controller
  - Distributed algorithm, e.g. using reservation bits in frame
- Token ring: token travels around ring and allows nodes to send one packet
  - Distributer version of polling
  - FDDI, …

Outline

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

Pure ALOHA

Poisson Process

Informal: memory less

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 $a(t_{1} + \Delta t) - a(t_{2})$ 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 $\lambda$ 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 = $\lambda' > \lambda$.
- 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 $2x m$ sec.

Pure Aloha: Analysis

- Calculate the “Probability of no collision” two ways:
  1. Probability that there is no arrival in interval $2x m$:
     \[ P(\text{no arrival in } 2x m) = e^{-2N\lambda' m} = e^{-2G} \]
  2. Since all new arrivals eventually get through, we have
     \[ \lambda'/\lambda' = S/G = \text{Fraction of transmissions that are successful} \]
- So, \[ S/G = e^{-2G} \] 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 $2xT$ 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
  - Slotted design is also not very efficient when carrying variable sized packets!
- Still, not bad for an absolutely minimal protocol!
  - Good solution if load is low – used in some sensor networking technologies (cheap, simple)

Outline

- Data link fundamentals
  - And what changes in wireless
- Aloha
- Ethernet
- Wireless-specific challenges
  - Ethernet review
  - How wireless differs
- 802.11 and 802.15 wireless standards
“Regular” Ethernet CSMA/CD

- Multiple Access: multiple hosts are competing for access to the channel
- Carrier-Sense: make sure the channel is idle before sending – “listen before you send”
- Collision Detection: collisions are detected by listening on the medium and comparing the received and transmitted signals
- Collisions results in 1) aborting the colliding transmissions and 2) retransmission of the packets
- Exponential backoff is used to reduce the chance of repeat collisions
  » Also effectively reduces congestion

Carrier Sense Multiple Access/ Collision Detection (CSMA/CD)

Ethernet Backoff Calculation

- Challenge: how do we avoid that two nodes retransmit at the same time collision
- Exponentially increasing random delay
  » Infer “number” senders from # of collisions
  » More senders → increase wait time
- First collision: choose K from {0,1}; delay is K x 512 bit transmission times
- After second collision: choose K from {0,1,2,3}
- After ten or more collisions, choose K from {0,1,2,3,4,...,1023}

How to Handle Transmission When Line is Sensed Busy

- p-persistent scheme:
  » Transmit with probability p once the channel goes idle
  » Delay the transmission by t_{prop} with the probability (1-p)
- 1-persistent scheme: p = 1
  » E.g. Ethernet
- nonpersistent scheme:
  » Reschedule transmission for a later time based on a retransmission delay distribution (e.g. exp backoff)
  » Senses the channel at that time
  » Repeat the process
- When is each solution most appropriate?
Collisions

Dealing with Collisions

- Collisions will happen: nodes can start to transmit “simultaneously”
  » Vulnerability window depends on length of wire
- Recovery requires that both transmitters can detect the collision reliably
  » Clearly a problem as shown on previous slide
- How can we guarantee detection?

Detect Collisions

- Limit length wire

Detect Collisions

- Minimum packet size
So What about Wireless?

- Depends on many factors, but high level:
  - Random access solutions are a good fit for data in the unlicensed spectrum
    » Lower control complexity, especially for contention-based protocols (e.g., Ethernet)
    » There may not always be a centralized controller
    » May need to support multi-hop
    » Also used in many unlicensed bands
  - Cellular uses scheduled access
    » Need to be able to guarantee performance
    » Have control over spectrum – simplifies scheduled access
  » More on this later in the course

Summary

- Wireless uses the same types of protocols as wired networks
  » But it is inherently a multiple access technology
- Some fundamental differences between wired and wireless may result in different design choices
  » Higher error rates
  » Must support variable bit rate communication
  » Signal propagation and radios are different

Outline

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

Wireless Ethernet is a Good Idea, but …

- Attenuation varies with media
  » Also depends strongly on distance, frequency
- Wired media have exponential dependence
  » Received power at d meters proportional to $10^{-kd}$
  » Attenuation in dB = $k d$, where $k$ is dB/meter
- Wireless media has logarithmic dependence
  » Received power at d meters proportional to $d^{-n}$
  » Attenuation in dB = $n \log d$, where $n$ is path loss exponent; $n=2$ in free space
  » Signal level maintained for much longer distances?
- But we are ignoring the constants!
  » Wireless attenuation at 2.4 GHz: 60-100 dB
  » In practice numbers can be much lower for wired
Implications for Wireless Ethernet

- Collision detection is not practical
  - Ratio of transmitted signal power to received power is too high at the transmitter
  - Transmitter cannot detect competing transmitters (is deaf while transmitting)
  - So how do you detect collisions?
- Not all nodes can hear each other
  - Ethernet nodes can hear each other by design
  - "Listen before you talk" often fails
  - Hidden terminals, exposed terminals, Capture effects
- Made worse by fading
  - Changes over time!

Hidden Terminal Problem

- Lack signal between S1 and S2 and cause collision at R1
- Severity of the problem depends on the sensitivity of the carrier sense mechanism
  - Clear Channel Assessment (CCA) threshold

Exposed Terminal Problem

- Carrier sense prevents two senders from sending simultaneously although they do not reach each other’s receiver
- Severity again depends on CCA threshold
  - Higher CCA reduces occurrence of exposed terminals, but can create hidden terminal scenarios

Capture Effect

- Sender S2 will almost always "win" if there is a collision at receiver R.
- Can lead to extreme unfairness and even starvation.
- Solution is power control
  - Very difficult to manage in a non-provisioned environment!
Wireless Packet Networking Problems

- Some nodes suffer from more interference than others
  - Node density
  - Traffic volume sent by neighboring nodes
- Leads to unequal throughput
- Similar to wired network: some flows traverse tight bottleneck while others do not