CS450
Network Embedded Sensing Systems
Week 6: Medium Access Control

Andreas Terzis
http://hinrg.cs.jhu.edu/CS450/
Outline

• Background in MAC protocols
  • Role and features of MAC protocols
  • Scheduled access vs. Randomized access
  • Additional WSN requirements
• Two Examples
  • S-MAC
  • BMAC
Characteristics of Sensor Network

- A special wireless ad hoc network
  - Large number of nodes
  - Battery powered
  - Topology and density change
  - Nodes for a common task
  - In-network data processing
- Sensor-net applications
  - Sensor-triggered bursty traffic
  - Can often tolerate some delay
  - Speed of a moving object places a bound on network reaction time
MAC and its Classification

- Medium Access Control (MAC)
  - When and how nodes access the shared channel
- Classification of MAC protocols
  - Scheduled protocols
    - Schedule nodes onto different sub-channels
    - Examples: TDMA, FDMA, CDMA
  - Contention-based protocols
    - Nodes compete in probabilistic coordination
    - Examples: ALOHA (pure & slotted), CSMA
MAC Attributes

- Collision avoidance
  - Basic task of a MAC protocol
- Energy efficiency
- Scalability and adaptivity
  - Network size, node density and topology change
- Channel utilization
- Latency
- Throughput
- Fairness

Primary

Secondary
Energy Efficiency in MAC Design

• Energy is primary concern in sensor networks
• What causes energy waste?
  • Collisions
  • Control packet overhead
  • Overhearing unnecessary traffic
  • Long idle time
    • Bursty traffic in sensornet applications
    • Idle listening consumes 50—100% of the power for receiving
      Dominant factor
  • Wakeup period = sleep period + listen period
  • Duty Cycle = listen period/Wakeup period
Scheduled Protocols

- TDMA
  - Advantages
    - No collisions
    - Energy efficient — easily support low duty cycles
  - Disadvantages
    - Bad scalability and adaptivity
      - Difficult to accommodate node changes
      - Difficult to handle inter-cluster communication
    - Requires strict time synchronization
Scheduled Protocols

- Polling
  - A master plus one or more slaves (star topology)
  - The master node decides which slave can send by polling the corresponding slave
  - Only direct communication between the master and a slave
  - A special TDMA without pre-assigned slots
- Examples
  - IEEE 802.11 infrastructure mode (CFP)
  - Bluetooth *piconets*
Bluetooth

• Target for wireless personal area network (WPAN)
  • Short range, moderate bandwidth, low latency
  • IEEE 802.15.1 (MAC + PHY) is based on Bluetooth
• Nodes are clustered into piconets
  • Each piconet has a master and up to 7 slaves – scalability problem
  • The master polls each slave for transmission
• Frequency-hopping CDMA between clusters
• Multiple connected piconets form a scatternet
  • Different to handle inter-cluster communications
Scheduled Protocols

- Bluetooth (Cont.)
  - How about Bluetooth radio with sensor networks?
  - Scalability is a big problem
  - Lack of multi-hop support
    - No commercial Bluetooth radio supports *scatternet* so far
  - Use two radios – expensive and energy inefficient
  - A node temporarily leave one piconet and joins another – high overhead and long delay
Scheduled Protocols

- Achieving peer-to-peer communications
- Self-Organization — by Sohrabi and Pottie
  - Have a pool of independent channels
    - Frequency band or spreading code
    - Potential interfering links select different channels
  - Talk to neighbors in different time slots
  - Sleep in unscheduled time slots
  - Looks like TDMA, but actually FDMA or CDMA
    - Any pair of two nodes can talk at the same time
  - Low bandwidth utilization
Scheduled Protocols

• LEACH: Low-Energy Adaptive Clustering Hierarchy — by Heinzelman, et al.
  • Similar to Bluetooth
  • CDMA between clusters
  • TDMA within each cluster
    • Static TDMA frame
    • Cluster head rotation
    • Node only talks to cluster head
    • Only cluster head talks to base station (long dist.)
• The same scalability problem
Contention-Based Protocols

- Contention-based protocols
  - CSMA — Carrier Sense Multiple Access
    - Listening before transmitting
    - Not enough for multi-hop networks (collision at receiver)

```
  a  b  c

Hidden terminal: a is hidden from c’s carrier sense
```

- CSMA/CA (CA stands for Collision Avoidance)
  - RTS/CTS handshake before send data
  - Other nodes (e.g. node c) delay transmission
    - Network Allocation Vector NAV aka virtual carrier sense
Contestion-Based Protocols

- Contention-based protocols (contd.)
  - MACA — Multiple Access w/ Collision Avoidance
    - Add duration field in RTS/CTS informing other nodes about their delay time
  - MACAW — improved over MACA
    - RTS/CTS/DATA/ACK
    - Fast error recovery at link layer
  - IEEE 802.11 Distributed Coordination Function (DCF)
    - Largely based on MACAW
Contention-Based Protocols

- Transmission rate control — by Woo and Culler
  - Based on a special network setup
    - A base station tries to collect data equally from all sensors in the network
  - CSMA + adaptive rate control
  - Promote fair bandwidth allocation to all sensors
    - Nodes close to the base station forward more traffic, and have less chances to send their own data
  - Helps in congestion avoidance
# Scheduled vs. Contention Protocols

<table>
<thead>
<tr>
<th></th>
<th>Scheduled Protocols</th>
<th>Contention Protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collisions</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Energy efficiency</td>
<td>Good</td>
<td>Bad</td>
</tr>
<tr>
<td>Scalability and adaptivity</td>
<td>Bad</td>
<td>Good</td>
</tr>
<tr>
<td>Multi-hop communication</td>
<td>Difficult</td>
<td>Easy</td>
</tr>
<tr>
<td>Time synchronization</td>
<td>Strict CS 450</td>
<td>Loose or not required</td>
</tr>
</tbody>
</table>
Energy Efficiency in Contention Protocols

- Contention-based protocols need to work hard in all directions for energy savings
  - Reduce idle listening – support low duty cycle
  - Reduce control overhead
  - Reduce collisions
  - Avoid unnecessary overhearing
Energy-Efficient MAC Design

- PAMAS: Power Aware Multi-Access with Signaling — by Singh and Raghavendra
  - Improve energy efficiency from MACA
  - Avoid overhearing by putting node into sleep
  - Use separate control and data channels
    - RTS, CTS, busy tone to avoid collision
    - Probe packets to find neighbors transmission time
  - Increased hardware complexity
    - Two channels need to work simultaneously, meaning two radio systems.
Energy-Efficient MAC Design

- Power save (PS) mode in IEEE 802.11 DCF
  - Assumption: all nodes are synchronized and can hear each other (single hop)
  - Nodes in PS mode periodically listen for beacons & ATIMs (ad hoc traffic indication messages)
- Beacon: timing and physical layer parameters
  - All nodes participate in periodic beacon generation
- ATIM: notifies receivers in PS mode to stay awake for Rx
  - ATIM follows a beacon sent/received
    - Unicast ATIM needs acknowledgement
    - Broadcast ATIM wakes up all nodes — no ACK
Energy-Efficient MAC Design

• Unicast example of PS mode in 802.11 DCF
Zigbee (I)

- Industry standard through application profiles running over IEEE 802.15.4 radios
- Target applications are sensors networks, interactive toys, smart badges, remote controls, and home automation
  - Personal Area Networks (PANs)
Zigbee (II)

- Three devices specified
  - Network Coordinator
  - Full Function Device (FFD)
    - Can talk to any device, more computing power
  - Reduced Function Device (RFD)
    - Can only talk to a FFD, simple for energy conservation
- Operation modes
  - Random access
    - CSMA/CA with optional ACKs on data packets
  - Scheduled access
    - Optional guaranteed time slots (GTS), which supports contention-free access
ZigBee (III)

- FFD acting as PAN coordinator periodically transmits frame beacon
  - Indicates start and configuration of **superframe**
    - Active and Inactive periods
    - Active period contains 16 slots
      - Contention Access Period (CAP) and Guaranteed Time Slots (GTSs)
Case Study: S-MAC

• S-MAC — by Ye, Heidemann and Estrin
• Tradeoffs
  - Latency
  - Fairness
  - Energy

• Major components in S-MAC
  - Periodic listen and sleep
  - Collision avoidance
  - Overhearing avoidance
  - Massage passing
Coordinated Sleeping

- Problem: Idle listening consumes significant energy
- Solution: Periodic listen and sleep

- Turn off radio when sleeping
- Reduce duty cycle to ~ 10% (120ms on/1.2s off)

Latency 🙁 ➔ 🙆 Energy
Coordinated Sleeping

- Schedules can differ

Node 1: 
- Listen
- Sleep
- Listen
- Sleep

Node 2: 
- Listen
- Sleep
- Listen
- Sleep

- Prefer neighboring nodes have same schedule — easy broadcast & low control overhead

Schedule 1

Schedule 2

Border nodes: two schedules
Coordinated Sleeping

- Schedule Synchronization
  - New node tries to follow an existing schedule
  - Remember neighbors’ schedules
    - to know when to send to them
  - Each node broadcasts its schedule every few periods of sleeping and listening
  - Re-sync when receiving a schedule update
- Periodic neighbor discovery
  - Keep awake in a full sync interval over long periods
Coordinated Sleeping

- Adaptive listening
  - Reduce multi-hop latency due to periodic sleep
  - Wake up for a short period of time at end of each transmission

- Reduce latency by at least half
Collision Avoidance

- S-MAC is based on contention
- Similar to IEEE 802.11 ad hoc mode (DCF)
  - Physical and virtual carrier sense
  - Randomized backoff time
  - RTS/CTS for hidden terminal problem
  - RTS/CTS/DATA/ACK sequence
Overhearing Avoidance

- Problem: Receive packets destined to others
- Solution: Sleep when neighbors talk
  - Basic idea from PAMAS (Singh, Raghavendra 1998)
  - But we only use in-channel signaling
- Who should sleep?
  - All immediate neighbors of sender and receiver
- How long to sleep?
  - The duration field in each packet informs other nodes the sleep interval
Message Passing

- **Problem:** Sensor net in-network processing requires *entire* message
- **Solution:** Don’t interleave different messages
  - Long message is fragmented & sent in burst
  - RTS/CTS reserve medium for entire message
  - Fragment-level error recovery — ACK
    — extend Tx time and re-transmit immediately
- Other nodes sleep for whole message time

Fairness 🙁 → Energy 🙌
Msg-level latency
Implementation on Testbed Nodes

- Platform
  - Mica Motes (UC Berkeley)
    - 20Kbps radio at 433MHz
- Configurable S-MAC options
  - Low duty cycle with adaptive listen
  - Low duty cycle without adaptive listen
  - Fully active mode (no periodic sleeping)
Experiments: two-hop network

- Topology and measured energy consumption on source nodes
- Source 1 ➔ Sink 1
- Source 2 ➔ Sink 2

- S-MAC consumes much less energy than 802.11-like protocol w/o sleeping
- At heavy load, overhearing avoidance is the major factor in energy savings
- At light load, periodic sleeping plays the key role
Energy Consumption over Multi-Hops

- Ten-hop linear network at different traffic load
- 3 configurations of S-MAC
- At light traffic load, periodic sleeping has significant energy savings over fully active mode
- Adaptive listen saves more at heavy load by reducing latency
Latency as Hops Increase

- Adaptive listen significantly reduces latency caused by periodic sleeping.

![Graphs showing latency increase with number of hops for two different traffic loads with and without adaptive listen.](image)
Throughput as Hops Increase

- Adaptive listen significantly increases throughput

- Using less time to pass the same amount of data
Combined Energy and Throughput

- Periodic sleeping provides excellent performance at light traffic load
- With adaptive listening, S-MAC achieves about the same performance as no-sleep mode at heavy load
Adaptive Listen Slots

- In S-MAC all nodes have listen slots of the same duration
  - Different nodes might have different tx/rx patterns
  - Idle listening wastes power
  - Idea: adaptively change the idle listen slot

B-MAC

• What is BMAC?
  • A configurable MAC protocol for WSNs
  • Small core
    • Factors out higher-level functionality
  • Energy efficient

• Goals
  • Low Power operation
  • Effective collision avoidance
  • Simple and predictable
  • Small code size and RAM usage
  • Tolerable to changing RF/networking conditions
  • Scalable to large numbers of nodes
Clear Channel Assessment

- MAC must accurately determine if channel is clear
  - Need to tell what is noise and what is a signal
  - Ambient noise is prone to environmental changes
- BMAC solution: ‘software automatic gain control’
  - Signal strength samples taken when channel is assumed to be free
  - Samples go in a FIFO queue (sliding window)
  - Median added to an EWMA filter
  - Once noise floor is established, a TX requests starts monitoring RSSI from the radio
CCA: single-sample thresholding vs. outlier detection

• Common approach: take single sample, compare to noise floor
  • Large number of false negatives
• BMAC: search for outliers in RSSI
  • If a sample has significantly lower energy than the noise floor during the sampling period, then channel is clear
CCA results

- 0=busy, 1=clear
- Packet arrives between 22 and 54 ms
- Single-sample thresholding produces several false ‘busy’ signals
Low Power Listening

- Goal: minimize listen cost
- Principles
  - Node periodically wakes up, turns radio on and checks channel
    - Wakeup time fixed
    - “Check time” variable
  - If energy is detected, node powers up in order to receive the packet
  - Node goes back to sleep
    - If a packet is received
    - After a timeout
  - Preamble length matches channel checking period
    - No explicit synchronization required
  - Noise floor estimation used to detect channel activity during LPL
Radio power up sequence of operations

- Goals
  - Minimize time radio is on
  - Minimize number of times radio gets started
  - Minimize sampling time (stage e)
LPL check interval

- Single-hop application doing periodic data sampling
- Sampling rate (traffic pattern) defines optimal check interval
- Check interval
  - Too small: energy wasted on idle listening
  - Too large: energy wasted on transmissions (long preambles)
- In general, it’s better to have larger preambles than to check more often!