Topic 14
Networking Issues in Wireless Sensor Networks

Bhaskaran Raman,
Department of CSE, IIT Kanpur

http://www.cse.iitk.ac.in/users/braman/courses/wless-spring2007/
Outline

- MAC protocols: S-MAC, B-MAC
- Routing protocol approaches
- Transport protocol: PSFQ
- Time synchronization: FTSP
- Overview of other issues: localization, data aggregation, topology/power control
Outline

- MAC protocols: S-MAC, B-MAC
- Routing protocol approaches
- Transport protocol: PSFQ
- Time synchronization: FTSP
- Overview of other issues: localization, data aggregation, topology/power control
MAC Protocol

• Classical wireless protocol: **CSMA/CA w/ RTS/CTS**
  - Carrier-Sense: listen before transmit
  - Collision Avoidance: backoff before transmit, and on collision
  - Request-to-Send, Clear-to-Send to address hidden node

• Challenge in embedded sensor platforms:
  - Power consumption during **listen** is significant
S-MAC (Sensor MAC)


- S-MAC ideas:
  - Periodic listen/sleep cycle
  - In listen phase, sleep on overhearing RTS/CTS
  - Virtual clusters
    - Neighbours (only) have to synchronize
    - Listen time has to account for clock drift also
    - Initial setup: synchronizer and follower
    - At border of two overlapping clusters: nodes have to wake-up on two different cycles
S-MAC: Delay vs. Energy Savings Trade-off

B-MAC (Berkeley MAC)


- **B-MAC ideas:**
  - Long preambles (> sleep time) while transmitting
  - Listen time further reduced, no synchronization needed

- **B-MAC exports interface:**
  - For application specific adaptation
  - Enable/disable CCA
  - Enable/disable ACK
  - Low-power listening: preamble length & check interval
Check Interval & Energy Consumed


Figure 4: Contour of node lifetime (in years) based on LPL check time and network density. If both parameters are known, their intersection is the expected lifetime using the optimal B-MAC parameters.
S-MAC/B-MAC Applicability

- For which applications is S-MAC/B-MAC applicable?

  ✔ Habitat monitoring
  ✗ Industrial motor monitoring (large sleep period, large data)
  ✗ Bridge monitoring (large sleep period, large data)
  ✗ Volcano monitoring (no sleep period, large data)
Routing Protocol

• Why is wireless different from wired?
  – Lack of link abstraction
    • Packet errors
    • Interference from neighbouring links
    • Self-interference (within a path)
    • Broadcast medium

• Challenge in embedded sensor platforms: low power
  – But blown out of proportion, in my opinion
  – Quick proof: no evaluation of any (non-trivial) routing protocol using any real application parameters
Some Routing Approaches

- **Data centric:**
  - **SPIN** (Sensor Protocols for Information via Negotiation)
    - ADV, REQ, DATA
    - Better than flooding/gossiping
  - **Directed diffusion:**
    - Flood query (specify value range, area of interest, etc.)
    - Response “diffuses” toward sink

- **Hierarchical:**
  - **LEACH** (Low-Energy Adaptive Clustering Hierarchy)
    - Cluster head chosen randomly
    - Nodes choose which cluster to belong to
    - Cluster head rotates
Routing Approaches: Applicability

- For which applications are the above routing approaches applicable?
  - ✗ Habitat monitoring
  - ✗ Industrial motor monitoring
  - ✗ Bridge monitoring
  - ✗ Volcano monitoring
  - ✔ No application has even considered using any of these
    - ✔ Too complex and abstract
    - ✔ No concrete application given
    - ✔ And I cannot think of any either
Routing Metrics

• Minimum-hop can cause problems
• Multi-hop LQI
  – $1/LQI$ is the metric
  – Assumes LQI to be stable over time
    • Assumption may not hold
    • Stability of routing?
• Used in the Redwood deployment
Transport Protocol

• Some applications require **reliable data transfer**
  – Examples: bridge monitoring, volcano monitoring, industrial motor monitoring

• TCP is not really applicable
  – Wireless errors
  – Broadcast medium
  – Congestion control is not an issue
  – Do not always have to deal with competing flows
PSFQ


  - An example protocol designed for sensor networks
  - For reliable transfer of data
  - Specifically designed for *one-to-many* data transfer, works for one-to-one transfer too
  - Example usage: *code update* from base to all other nodes
PSFQ: How it Works

- **Main idea:**
  - Pump-Slowly: refers to data going in forward dirn.
  - Fetch-Quickly: refers to error recovery in reverse dirn.
  - Cycle repeats until data transfer is done successfully

- **Protocol details:**
  - **Timers:** Pump timer & fetch timer are used
  - Fetch can be signal strength based (who is parent in tree)
  - **Proactive fetch:** when nothing received for some time
  - **Report bit:** used by sender to request for ACK
PSFQ Performance

• When can you expect PSFQ to perform well?
  – When effect of pipelining is seen
  – That is, multiple simultaneous hops being used simultaneously

• Crucial parameters: timers
  – May not be that easy to determine optimally
  – Industrial monitoring paper reports poor performance
PSFQ Applicability

• For which applications is PSFQ applicable?
  ✗ Habitat monitoring (reliability not needed)
  ✗ Industrial motor monitoring (they have used it, but reported poor performance, small networks anyway)
  ✗ Bridge monitoring (PSFQ is an overkill)
  ✗ Volcano monitoring (PSFQ is an overkill)
Time Synchronization

• Required for some applications
• Useful for other protocols (e.g. MAC)
• Challenges:
  – Different clocks
  – Clocks drift
  – Clock drift rate may change (with temperature, for e.g.)
  – Multi-hop
FTSP

• FTSP: Flooding Time Synchronization Protocol

• Goal: achieve micro-second granularity synchronization for networks of 100s of nodes
FTSP: How it Works

• Message time-stamping to synchronize clocks
• Multiple such messages to estimate clock drift
  – Using linear regression
• Such synchronization messages can be sent by root, or any synchronized node
FTSP Applicability

• Which applications find use for FTSP?

✔ Volcano monitoring (really needed? or was it used because the software was available?)
✗ Bridge monitoring (FTSP is an overkill)
✗ Industrial motor monitoring (no need for micro-s synch.)
✗ Habitat monitoring (no need for micro-s synch.)
Other Issues in Literature

- **Data aggregation**
  - Scenario description lacks depth thus far

- **Localization**
  - Requirement description lacks depth thus far

- **Topology, power control**
  - Feasibility in question: RSSI variability
  - Usefulness in question: power consumption does not increase that much with tx-power in practice

- **Security**
  - Depth justified only for military applications (if at all), which is taboo for these lectures
Summary

• Several protocols designed in literature, books have been written
  – MAC, Routing, Transport, Synchronization
  – Data aggregation, Localization, Topology/power control, Security

• Field is rich for paper generation (lots of abstract constraints)

• But real applications thus far have used only simple protocols