Mobile and Sensor Systems

Lecture 3: Wireless Sensor Systems and IoT
Prof Cecilia Mascolo

Acks to Dr A. Gaglione (Digital Catapult) for the slides on IoT
In this lecture

• We will describe wireless sensor networks in general and the properties of sensor nodes.
• We will introduce sensor network MAC Layer issues and some solutions.
• We will illustrate IoT MAC protocols
Wireless Sensor Networks?

• In many situations, we want to measure things to develop a better understanding of various phenomena.
• With this insight, we can then design novel or improved systems.
Example: Sensor Network Macroscope

Conventional Manual Monitoring

Long-lived High-density Sensing Networks
Many Applications

- Structural health monitoring
- Environmental monitoring
- Animal behaviour
- Warehouse logistics
Characteristics

• Farmer wants to cover his entire vineyard
  – Large number of sensing devices.

• He wants to keep the cost down
  – Low cost, resource constrained.

• He cannot run wires to these many devices
  – Battery powered, wireless.
An Example of Sensor Network Architecture

- **Sensor Node**
- **Patch Network**
- **Sensor Patch**
- **Gateway**
- **Transit Network (IP or not)**
- **Intranet/Internet (IP)**
- **Client Data Browsing and Processing**
- **Access point**
  - Base station
  - Proxy
- **Data Service**
- **Other information sources**
Sensor Systems
vs Standard or Mobile Systems

• Sensor nodes have limited computational resources and energy.
• Sensor nodes are prone to failures (especially because they are often deployed in challenging conditions).
• The topology of a sensor network might not change frequently:
  – Many deployments involve nodes with fixed locations.
  – Some deployments may have mobile sensors.
Sensor Node

• A typical sensor node is composed of,
  – Sensing device
  – Low power radio
  – Small storage
What Operating System runs on a sensor?

• Operating system useful to simplify programming tasks and to allow more control over operations of the system

• But what can we do with such a constrained device?

• Given the kind of applications needed it is important to support concurrency…[frequent and parallel collection from different sensors]
Energy Management

• Local computation does not consume significant amount of energy.

• **The main source of energy consumption is the radio.**

• Current draw on Telosb,
  – Microcontroller ON, Radio OFF 1.8mA
  – Microcontroller ON, Radio ON 21mA
Energy Management

• In order to save energy, limit the number of radio transmissions.

• Idle listening consumes as much power as transmission.

• Current draw on Telosb,
  – Idle listening 23mA
  – Transmitting 21mA

• Idle listening is wasteful when average data rate is low.

• Switch off the radio when idle.

• Transmissions from other sensor nodes are lost.
Radio Duty Cycling

• Switch off the radio of all sensors at specific intervals:
  – Very precise synchronization.
  – Still probable idle time for sensors which do not communicate.

• More refined strategy:
  – Wave of switch off time depending on topology.
  – Still an overestimate of the communication needs of some sensors (traffic might be varying across the network).
Dynamic duty cycling

- More refined strategies have been proposed which aim to allow sensors with more packets to stay awake longer and others to sleep more.
  - Synchronized (e.g. S-MAC)
  - Asynchronous (e.g. B-MAC, X-MAC)

- Synchronized protocols try to **negotiate a schedule** among neighboring nodes.
- Asynchronous protocols rely on **preamble sampling** to connect a transmitter to receivers.
Sensor-MAC (S-MAC)

- Idea: Switch nodes off, ensure that neighboring nodes turn on simultaneously to allow packet exchange (rendez-vous)
  - Packet exchange occurs only in these **active periods**
  - Need to also exchange wakeup schedule between neighbors
  - When awake, essentially perform RTS/CTS
  - Use SYNCH, RTS, CTS phases
S-MAC

- SYNC phase divided into time slots with CSMA and backoffs to send schedule to neighbours.
- Y chooses a slot and if no signal is received in this slot, it will transmit its schedule to X otherwise it will wait for next wake up of X.
- RTS phase: X listens for RTS packets (CSMA contention).
- CTS phase: X sends one and extends its wake up time.
S-MAC synchronized islands

- Nodes try to pick up schedule synchronization from neighboring nodes.
- If no neighbor found, nodes pick some schedule to start with.
- If additional nodes join, some node might learn about two different schedules from different nodes — “Synchronized islands”.
- To bridge this gap, it has to follow both schemes and use more energy.
Preamble Sampling
(Low Power Listening, LPL)

• So far: Periodic sleeping supported by some means to synchronize wake up of nodes to ensure rendez-vous between sender and receiver.

• Alternative option: Don’t try to explicitly synchronize nodes:
  – Have receiver sleep and only periodically sample the channel.

• Use long preambles to ensure that receiver stays awake to catch actual packet. Example: B-MAC and WiseMAC.

Start transmission:

Long preamble  Actual packet

Check channel  Check channel  Check channel  Check channel

Stay awake!
Problems with this technique

- **Overhearing**
  - All receivers listening to the preamble have to stay awake to find out who is the intended receiver.

- **Energy Consumption**
  - Long preamble causes increased energy consumption at both the transmitter and the receiver.

- **Latency**
  - Long preamble introduces per-hop latency.
X-MAC

• Short preamble
  – Reduce latency and reduce energy consumption

• Target in preamble
  – Minimize overhearing problem.

• Adding wait time between preambles
  – Reduces latency for the case where destination is awake before preamble completes.
X-MAC

Target address in data header

Long preamble

Send DATA

extended wait time

Recv DATA

R wakes up

Listen for queued packets

Short preambles with target address information

Receive early ACK

Send DATA

ack

Recv DATA

Time & energy saved at S & R

R wakes up

Send early ACK

X-MAC

Sender (S)

LPL

Sender (S)

LPL

Receiver (R)

X-MAC

Sender (S)

X-MAC

Receiver (R)
Low-Energy Adaptive Clustering Hierarchy (LEACH)
Low-Energy Adaptive Clustering Hierarchy (LEACH)

- Assumption: dense network of nodes, reporting to a central sink, each node can reach sink directly.
- Idea: Group nodes into “clusters”, controlled by clusterhead:
  - Setup phase; details: later.
  - About 5% of nodes become clusterhead (depends on scenario).
  - Role of clusterhead is rotated to share the burden.
  - Clusterheads advertise themselves, ordinary nodes join CH with strongest signal.
  - Clusterheads organize: CDMA code for all member transmission. TDMA schedule to be used within a cluster
- In steady state operation:
  - CHs collect & aggregate data from all cluster members.
  - Report aggregated data to sink using CSMA.
Low-Energy Adaptive Clustering Hierarchy (LEACH)
LEACH rounds

Setup phase

Steady-state phase

Advertisement phase

Cluster setup phase

Broadcast schedule

Self-election of clusterheads

Clusterheads compete with CSMA

Members compete with CSMA

Fixed-length round

Time slot 1

Time slot 2

......

Time slot n

Time slot 1

......
Internet of Things
What is the IoT?

“A global infrastructure for the information society, enabling advanced services by interconnecting things based on existing and evolving interoperable information and communication technologies” [1].

“A system of physical objects that can be discovered, monitored, controlled, or interacted with by electronic devices that communicate over various networking interfaces and eventually can be connected to the wider internet” [2].

IoT landscape

**Devices** (tags, sensor nodes, mobile and wearable devices)

**Machines** (home appliances, security systems, vehicles, etc.)

**Environments** (smart homes, buildings, cities)
Typical IoT system architecture

**IoT Cloud**
- Service hosting
- Visualisations
- Advanced analytics
- Data storage

**IoT Gateway**
- Edge analytics
- Local storage

**IoT Device**
- Basic processing
- Short/wide comms
- Sensing/actuation
Classification of communication technologies (range)

- **Near field (< 10 cm)**: RFID, NFC, QR codes
- **Personal area network (PAN)**: Bluetooth / BLE, 802.15.4, Zigbee
- **Local area network (LAN)**: WiFi (802.11x)
- **Wide area network (WAN)**: GSM (2G) / 3G / 4G / 5G, LPWAN (LoRaWAN, Sigfox, NB-IoT, ...)

Ranges:
- Near field: < 10 cm
- Personal area network (PAN): 1 m – 50 m
- Local area network (LAN): 50 m – 1 km
- Wide area network (WAN): 1 km – 50 km
Classification of communication technologies (range vs bandwidth)

- **Bandwidth**
  - High
  - Medium
  - Low

- **Range**
  - Short
  - Medium
  - Long

- Technologies:
  - WiFi (802.11x)
  - Bluetooth BLE
  - 802.15.4, Zigbee
  - 2G
  - 3G
  - 4G
  - 5G
  - RFID, NFC
  - LPWAN
Main characteristics

• Long range communication links

• Low bandwidth, low power

• Deep indoor penetration

• Very cheap radio modules
Typical LPWAN applications

- Smart metering
- Air quality monitoring
- Smart lighting
- Asset tracking (not real-time)
- Tank monitoring
- Waste management

These applications are typically delay tolerant and require the transmission of only a few packets per hours

LoRaWAN
What is LoRaWAN?

- LoRaWAN is the MAC/network layer which uses LoRa as the underlying radio modulation.

3 classes of devices:

A. Battery powered devices, bi-directional comm.
B. Battery powered devices, bi-directional comm., extra receive windows
C. Main powered devices
End-device classes

- **Battery powered – Class A**
  - Bidirectional communications
  - End-device initiates communication (uplink)
  - Server communicates with end-device (downlink) during predetermined response windows
  - For every uplink, there are two possible downlink slots
  - Low power consumption, high latency
End-device classes

• **Low latency – Class B**
  – Bidirectional with scheduled receive slots
  – There are pre-programmed downlink slots. Downlink is possible at these times
  – Periodic beacon from gateway to synchronize downlink slots
  – High power consumption, low latency
End-device classes

• No latency – Class C
  – Bidirectional communications
  – Server can initiate transmission at any time
  – End-device is constantly receiving
  – Main-powered devices, no latency
Summary

• We have described the characteristics of sensor systems and their challenges
• We have introduced MAC protocols for sensor systems
• We have illustrated the IoT protocols

Acks to Dr A. Gaglione (Digital Catapult) for the slides on IoT
Suggested Readings

• TinyOS tutorial: http://www.tinyos.net/tinyos-1.x/doc/tutorial/


Suggested readings


