# Wireless Sensor Networks

Prof. John McLeod

ECE9047/9407, Winter 2021

Wireless sensor networks are one increasingly important application of embedded systems. This lesson serves as an introduction to some concepts related to wireless sensor networks. This material is heavily based on Chapters 1 and 2 of B. Krishnamachari's book, *Networking Wireless Sensors*.

## *Overview*

We previously discussed embedded systems in the context of small, inexpensive devices whose capabilities and functionality was designed and optimized to perform a few well-defined tasks. **Wireless sensor networks** (WSNs) build on this concept to create a network with a large number of embedded systems all acting as sensors to collect data in a spatially-distributed environment.

*Definition:* A **wireless sensor network** is a inter-connected wireless network of embedded systems, all collecting and sharing the same kind of sensing data.

The environments a WSN can be deployed in are wide-ranging, but generally break down into two broad categories: **environmental WSNs** and **industrial WSNs**. Some examples of **environmental WSNs** are:

- *Ecological habitats and protection zones* for tracking wildlife migrations, monitoring the weather, or measuring the impact of tourists.
- *Air pollution monitoring* in urban areas, to identify sources of pollution and provide air-quality updates to residents.
- *Forest fire detection* in remote areas.

Some examples of **industrial WSNs** are:

- *Factories* for monitoring assembly line processes, live updates of warehouse contents.
- *Large buildings* for monitoring temperature, water pressure, and structural integrity.
- *Machine health monitoring* to identify parts that need replacing before they fail and disrupt production.

Full deployment of WSNs is predicted to "... *dwarf previous milestones in the information revolution*," [1] and provide unprecedented ability to observe real-world phenomena with precise spatial and temporal resolution.

[1] This was from a US National Research Council report written all the way back in 2001. Twenty years later it hasn't quite happened yet...

## *The Basic WSN Device*

As you may guess from the name, a WSN consists of embedded systems that typically have some form of sensor input and some form of wireless output.

*Definition:* The embedded systems that comprise the sensing component of the WSN are called **sensor nodes**. They are also sometimes called **motes**.

A **sensor node** will consist of:

- A *microcontroller processor*, typically one with very low power and consequently limited computing power.
- *Memory* (RAM and ROM) for storing device programming and sensor data between wireless transmissions.
- A *wireless transceiver* for transmitting sensor data, and possibly for receiving instructions. The wireless transceiver for a WSN device is usually low-bandwidth and short-range compared to typical wireless transceivers in personal electronics (such as a cellular or WiFi transmitters), this is to minimize power consumption.
- *Sensors* as peripherals to the microcontroller. The exact sensors used are obviously appropriate to the task of the WSN. Again, power consumption constraints often limit the precision and data-collection rate of sensors.

- *Analog-to-digital converters* are often needed for converting the sensor input to a binary value.
- A *GPS locator*, for recording the location of the WSN device. This is unnecessary in systems that employ devices in fixed, planned locations, but many WSNs may involve moving sensors (on vehicles, people, or animals) or involve sensors that are deployed to random locations.
- A *power source* for all of the sensors and electronics.

Some WSN devices can be hardwired to a power grid, effectively eliminating the need for power-efficient components, but most WSN devices are expected to operate independently for a prolonged period of time. The capacity and cost of these independent power sources (i.e. batteries, sometimes augmented with solar cells) is a major limiting factor in both the design of the WSN device and how the topology and deployment of the WSN is conducted.

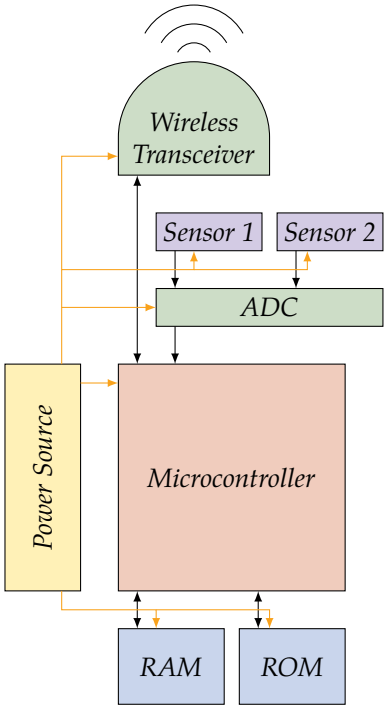


Figure 1: Simplified schematic of a basic sensor node.

## *Design Considerations*

The basic idea behind WSNs is to distribute a large number of cheap, autonomous **sensor nodes** over an area, and let them do their thing. You will receive all the recorded data, and can then enjoy working with "big data", which apparently is all the rage these days. Unfortunately for wannabe data analysis, several limitations of modern technology present serious constraints on the design of WSN devices.

- *Lifetime*: How long will the WSN devices be deployed before servicing? Generally, the limitations of the power source place an upper limit on the lifetime of the WSN device. Increasing lifetime will require either limiting the power consumption of the device, hard-wiring the device to the power grid, or adding renewable sources of energy.

- *Responsiveness*: To decrease power consumption, WSN devices can switch between active and sleeping modes. However this can limit the response time of the WSN to new events.

- *Robustness*: An ideal WSN covers a large area with a high density of devices. Minimizing the expense of the WSN requires using inexpensive devices, which often have a high failure rate. A WSN should not be sensitive to individual device failures, and the entire WSN should slowly degrade, rather than catastrophically fail, after significant numbers of devices fail.

- *Synergy*: An over-used business-speak word, but in this case

it means developing a WSN where the system-wide performance is superior to the sum of the individual devices. This implies devices must collaborate on computing, communicating, and data storage: for example if a few devices sense a new event, adjacent devices that are not actively sensing can participate in analysing the sensing data.

- *Scalability*: A large-area, high-density WSN is often difficult to deploy all at once. An ideal WSN is scalable, able to handle (at reduced performance) large-area, low-density networks or small-area, high-density networks; and able to smoothly incorporate added (or removed, in the event of failure) devices without downtime.

- *Heterogeneity*: From a manufacturing and servicing perspective, a WSN in which all devices are identical (a homogeneous WSN) is certainly preferred. However to meet the required performance metrics while balancing all the design constraints, some degree of heterogeneity may be required: instead of each device having a full set of sensors, many different devices each with different sensors might be used to minimize cost and power consumption. As another example, most of the computing and communication power can be placed in a central hub (a full-fledged computer, perhaps).

- *Self-Configuration*: A WSN should operate smoothly with minimal human interaction. Ideally, once turned on, the devices in the WSN can automatically incorporate themselves into the WSN, synchronize and calibrate their sensors with the larger

network, and inform neighbouring devices or hubs of their position.

- *Self-Optimization*: The actual operating conditions of WSN is not always carefully controlled (consider, for example, a WSN to monitor an ecological protection zone). The WSN should have some ability to fine-tune by learning from, and improving performance of, its own sensor data.

- *Systematic Design*: Many WSNs have very specific, and sometimes unique, design requirements. Developing such a WSN is a trade-off between developing devices tailored to those requirements, and developing devices with more general applicability. The latter may be important, for example, if environmental factors change during the lifetime of the WSN.

- *Security*: A WSN can collect a wide variety of data, and transmits that data through the network. Hackers may capture a device, or intercept/spoof data transmission. Depending on the purpose of the WSN, information security may be very important.

Clearly, the design of a **sensor node**, and the deployment of the WSN, must be conducted with respect to some or all of these constraints.

## *Deployment: Structured or Random?*

When deploying your WSN, do you scatter the **sensor nodes** to the winds, or do you carefully place each at a predefined location?

*Definition:* A WSN is **structured** when each sensor node is placed at a predefined location, usually defined with respect to the location of other sensor nodes according to a simple rule (i.e. a mesh or a grid).

*Definition:* A WSN is **random** when each sensor node is placed without any systematic pattern and without any consideration of the location of other nodes in the network.

Despite what you may think at first, there are often good mathematical reasons for trying to deploy a fully randomized WSN.

- If the environment is poorly-understood before the network is deployed, it may be difficult to predetermine the best locations for sensors. A random distribution can sometimes offer equivalent performance with fewer devices than a structured distribution.

There are also good practical reasons for trying to deploy a **random** WSN.

- For wide-area ecological, agricultural, or pollution monitoring, dropping WSN devices from an aircraft is a cheap deployment method (assuming the WSN devices can survive the fall).

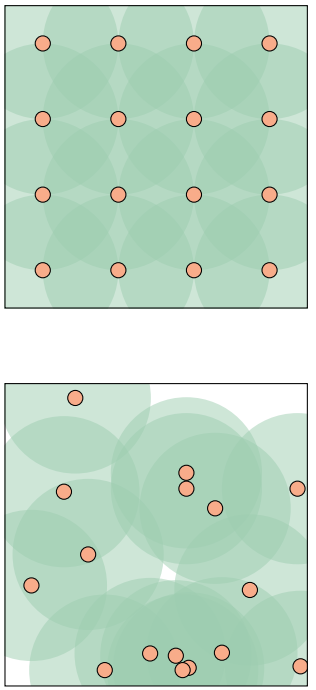

Figure 2: **Structured** WSNs tend to give equal coverage to all areas in the region, while **random** WSNs tend cluster, providing better coverage in some areas and worse coverage in others.

However real-world examples of fully random WSNs are limited. In particular, a **heterogeneous** WSN may use computer servers for hubs or gateways, these obviously have to be carefully placed and often connected to the power grid. [2] Many WSNs also are installed by people, making fully random deployment onerous. One compromise is to use a **pseudo-random** deployment:

- First, place hubs/gateways (if any) at locations suitable for their unique constraints (i.e. grid power, perhaps even indoors).
- Second, sensor devices are densely placed at "priority areas", where it is expected (through experience, analysis, or even just guesswork) that most relevant events will occur.
- Finally, additional sensor devices are placed to cover (or partially cover) any blind spots and to ensure the entire network can be connected.

Of course the deployment method is primarily driven by the nature of the WSN: many industrial WSNs (especially those in factories or warehouses) benefit from being **structured**.

[2] The differences between **heterogeneous** and **homogeneous** WSNs are discussed in more detail below.

## *Deployment: Redundancy or Replacement?*

When designing the WSN it is important to consider how to handle device failure. A WSN with particularly cheap sensor nodes may be designed with considerable **redundancy**, where multiple nodes cover the same sensing area. In this case, it will take multiple device failures to make a region go "dark".

- This type of WSN requires a particularly large initial deployment, but minimizes future support costs.

Alternatively, WSNs may be designed to have failing devices **replaced** as needed. This approach might be used if each device is expensive and/or is expected to have a long operational lifetime.

- In this case, there usually is unavoidable (and unexpected) periods when regions go "dark" after a device fails and before it is replaced.
- This type of WSN typically has lower deployment costs, and larger support costs.

Again, the degree of redundancy and the expected replacement cycle depend largely on the nature of the WSN. WSNs in controlled environments (such as temperature and humidity monitoring in buildings) can typically have sensor nodes replaced with little additional cost over the price of a new device. However WSNs in uncontrolled environments (such as water quality monitoring along a river network) may have prohibitive replacement costs.

## *Deployment: Homogeneous or Heterogeneous?*

Arguably the design ideal for a WSN is a network of identical nodes, each of which participates in all necessary operations (typically sensing, data processing, data storage, communication), [3] but in many cases it is simpler and cheaper to design a WSN with at least some specialist nodes.

[3] This is sometimes referred to as **smart dust**: a collection of sensor nodes that can be randomly deployed in any environment, will automatically construct an ad-hoc network, and will constantly adjust task delegation depending on the events occurring within the environment.

*Definition:* A **node** is a device connected to the WSN. Sensor nodes, previously discussed, are one kind of node.

*Definition:* A **homogeneous** WSN uses identical devices for all required operations. All **nodes** can act as sensor nodes in a **homogeneous** WSN.

*Definition:* A **heterogeneous** WSN uses specialist devices for each task. Obviously sensor nodes are used for sensing, but often other devices are used for long-range communications, data storage, and/or data processing.

*Definition:* A **hub** is a type of node whose primary purpose is facilitating communication between different regions of the network, or between the WSN and the outside world.

When the WSN is **homogeneous** and devices are identical, then deployment is fairly simple.

- There is no intrinsic order to deployment, since each device is the same.

- Deployment can be carried out in parallel by multiple agents with limited information on the other's activities (i.e. "each of you take a walk through the park and attach one of these to a tree every 30 to 50 feet.").

If the WSN is **heterogeneous**, there is usually a required order to deployment. Furthermore, fully random deployment is often not possible.

- Typically **hubs** are placed first at key locations.
- If different types of **sensor nodes** are used, then high-quality sensor devices may then be distributed to priority areas or on a structured grid.
- Finally, low-quality **sensor nodes** (if used), can be placed to cover remaining areas, or even randomly deployed.

Most practical WSNs today are heterogeneous, and the deployment of **sensor nodes** — whether **structured** or **random** — occurs within regions serviced by **hubs**.

## *Deployment: Network Topology*

Power efficiency is usually prioritized in WSN **sensor nodes**. Wireless communication requires significant power consumption, and this consumption increases with increasing communication radius. Consequently the network topology of a WSN is an important consideration.

*Definition:* **Network topology** refers to which nodes can communication with which, and the communication path(s) between any two nodes in the network.

Ideally, the WSN would have every node able to communicate with every other node. Because of the need for power efficiency, this is rarely feasible. As previously mentioned, I use the term **hub** to refer to a node that prioritizes communication power over sensing. [4]

- In a **heterogeneous** WSN, **hubs** are often specially designed for this role. In fact, often computers connected to the power grid (if available) are used rather than embedded systems.
- A **homogeneous** WSN may still have **hubs** if the WSN has the *self-configuration* and *self-optimization* ability to identify nodes which are less needed for sensing (either due to redundancy in the area or historical lack of sensing events) and can instead prioritize power usage for communication.
- A truly advanced WSN may be capable of dynamically promoting **sensor nodes** to **hubs** (and vice-versa) based on real-

[4] Other texts call these **gateways** or **cluster head nodes**, and sometimes a **gateway** is a the **hub** that also communicates with the outside world.

time information such as regional sensing event activity and particularly nodes running low on power.

One of the most important descriptions of WSN network topology is the number of **hops** in the network.

*Definition:* A **hop** is each set of direct node-to-node communication that must be traversed before data reaches a **hub**.

Some aspects of network topology, such as the degree of connectivity and other performance metrics, will be discussed next week. For the moment, we simply discuss some broad categories of network topology.

*Definition:* A **star-connected single-hop** network has each sensor node communicate directly with the hub.

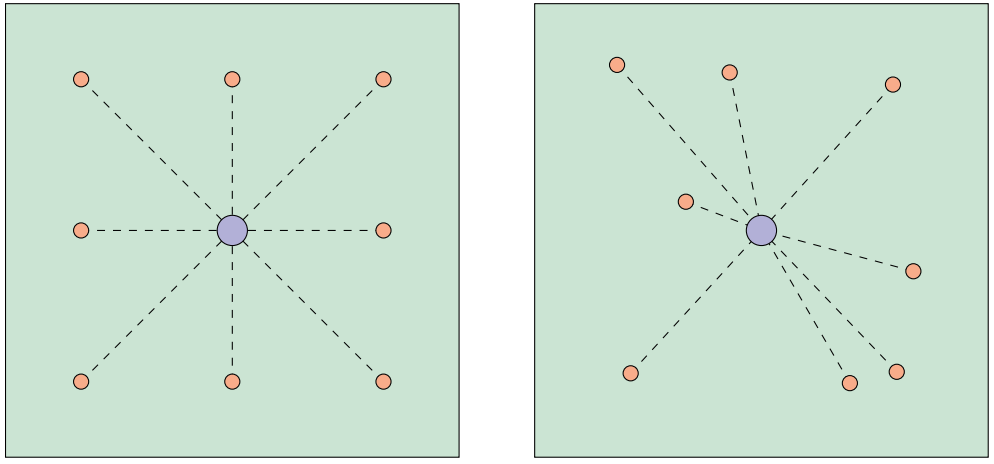

Figure 3: Star connected single-hop networks can be structured or random.

*Definition:* A **multi-hop grid** or a **multi-hop mesh** network requires some sensor nodes to pass data on to other sensor nodes before reaching the hub. A *grid* is typically a **structured** WSN, a *mesh* is a less-structured, **random**, or **pseudo-random** WSN.

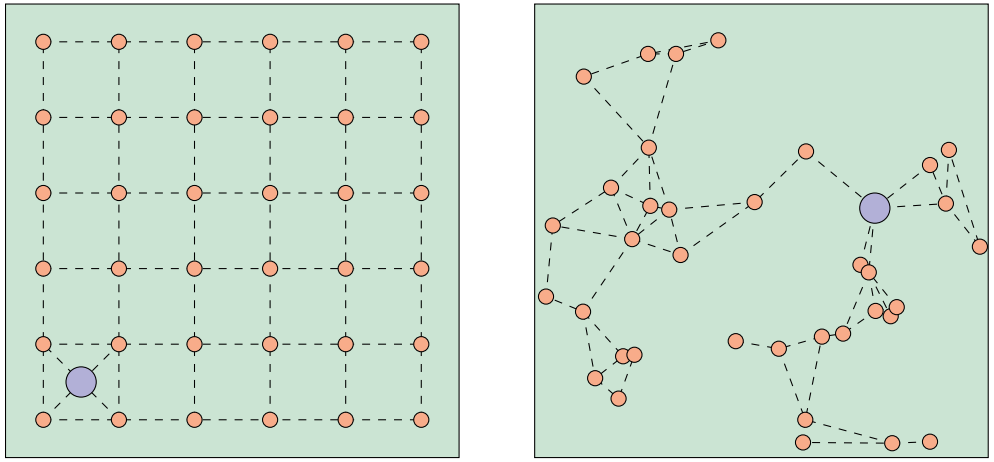

Figure 4: Multi-hop grid and mesh networks. As with all random or pseudo-random networks, care must be taken to ensure all parts of the network are connected together.

From a device design perspective, *single-hop* networks are preferred compared to a *multi-hop* network.

- In a *single-hop* network each **sensor node** only needs to dedicate resources to sensing and relaying that data to the **hub**.
- In a *multi-hop* network, each **sensor node** also sometimes needs to act as a **relay** for data from other **sensor nodes**.

The trade-off, however, is that for a given **communication distance** and a given environment area, a *single-hop* topology may require more **hubs** than a *multi-hop* topology. For larger networks a layered approach may be necessary in the topology.

*Definition:* A **two-tier hierarchical cluster** network has two "classes" of **hub**. A **local hub** communicates with a smaller *cluster*, or region, of **sensor nodes**. Each **local hub** can also communicate with other **local hubs**, and with a **master hub**.

The basic idea with a *hierarchical cluster* network is to connect smaller, relatively independent WSNs together into a larger network that covers the entire environment area.

- This allows **sensor nodes** with low-power, short-range communication to cover a wide area while still being part of what is essentially a *single-hop* network.
- This also allows any data storage or data processing operations of the WSN to be conducted in these smaller sub-networks.

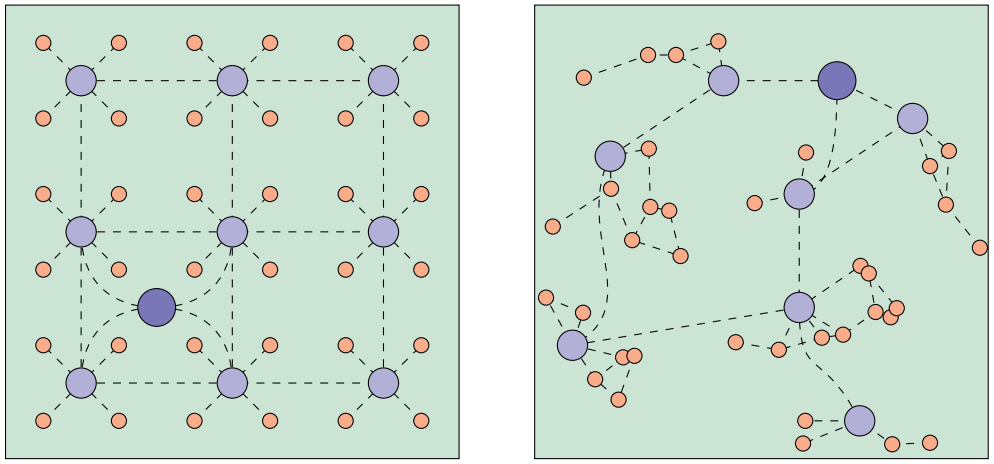

Figure 5: Two-tier hierarchical cluster networks, either structured or random. As with all random or pseudo-random networks, care must be taken to ensure all parts of the network are connected together.

Of course the trade-off here is that even more **hubs** need to be deployed, but often that is a necessary trade-off for a large WSN. It is

possible to have *three-* or even higher tiered networks if additional "classes" of **hubs** are introduced, but you get the idea.

- The clusters within a **two-tier hierarchical cluster** network may be any mixture of structured or random, single- or multi-hop networks.
- A **homogeneous** WSN can become a **two-tier hierarchical cluster** again if the WSN has the *self-configuration* and *self-optimization* ability; and the nodes within the network themselves use some process of self-selection to determine which will act as **local hubs**, which will become the **master hub**, and which will remain **sensor nodes**.

In buildings or urban environments it is often possible to connect the **local hubs** and **master hub** in a wired network, which has advantages in terms of communication security and reliability. In that case, only the distributed **sensor nodes** are really "wireless".